

MBS MacCocoa 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 MacCocoa Plugin

0.2 Content

- 1 List of all topics 3
- 2 List of all classes 123
- 3 List of all controls 127
- 4 List of all global methods 129
- 5 All items in this plugin 131
- 26 List of Questions in the FAQ 1165
- 27 The FAQ 1175

Chapter 1

List of Topics

• 6 Addressbook	133
– 6.1.1 class ABAccountMBS	133
* 6.1.3 Constructor	134
* 6.1.5 BaseURL as String	134
* 6.1.6 Handle as Integer	134
* 6.1.7 Identifier as String	134
* 6.1.8 isMainAccount as Boolean	134
* 6.1.9 Name as String	135
– 6.2.1 class ABAddressBookMBS	136
* 6.2.3 ABAddressBookErrorDomain as string	136
* 6.2.4 ABMultiValueIdentifiersErrorKey as string	136
* 6.2.5 accountWithIdentifier(Identifier as string) as ABAccountMBS	137
* 6.2.6 addRecord(record as ABRecordMBS) as boolean	137
* 6.2.7 addRecord(record as ABRecordMBS, Account as ABAccountMBS, byref error as NSErrorMBS) as boolean	138
* 6.2.8 addRecord(record as ABRecordMBS, byref error as NSErrorMBS) as boolean	138
* 6.2.9 addressBook as ABAddressBookMBS	138
* 6.2.10 allAccounts as ABAccountMBS()	138
* 6.2.11 Constructor	139
* 6.2.12 enabledAccounts as ABAccountMBS()	139
* 6.2.13 EnableEvent	139
* 6.2.14 formattedAddressFromDictionary(address as Dictionary) as NSAttributedStringMBS	139
* 6.2.15 GotSharedAddressbook as boolean	140
* 6.2.16 groupForName(name as string) as ABGroupMBS	140
* 6.2.17 groupForUniqueId(uniqueid as string) as ABGroupMBS	140

* 6.2.18 groupForUniqueId(uniqueid as string, account as ABAccountMBS) as ABGroupMBS	141
* 6.2.19 groups as ABGroupMBS()	141
* 6.2.20 groupsForAccount(account as ABAccountMBS) as ABGroupMBS()	142
* 6.2.21 kABAddressCityKey as string	142
* 6.2.22 kABAddressCountryCodeKey as string	142
* 6.2.23 kABAddressCountryKey as string	144
* 6.2.24 kABAddressHomeLabel as string	144
* 6.2.25 kABAddressProperty as string	144
* 6.2.26 kABAddressStateKey as string	144
* 6.9.3 kABAddressStreetKey as string	220
* 6.2.28 kABAddressWorkLabel as string	145
* 6.2.29 kABAddressZIPKey as string	145
* 6.2.30 kABAIMHomeLabel as string	145
* 6.2.31 kABAIMInstantProperty as string	145
* 6.2.32 kABAIMMobileMeLabel as string	145
* 6.2.33 kABAIMWorkLabel as string	146
* 6.2.34 kABAlternateBirthdayComponentsProperty as string	146
* 6.2.35 kABAnniversaryLabel as string	146
* 6.2.36 kABAssistantLabel as string	146
* 6.2.37 kABBirthdayComponentsProperty as string	146
* 6.2.38 kABBirthdayProperty as string	147
* 6.2.39 kABBrotherLabel as string	147
* 6.2.40 kABCalendarURIsProperty as string	147
* 6.2.41 kABChildLabel as string	147
* 6.2.42 kABCreationDateProperty as string	147
* 6.2.43 kABDatabaseChangedExternallyNotification as string	147
* 6.2.44 kABDatabaseChangedNotification as string	148
* 6.2.45 kABDeletedRecords as string	148
* 6.2.46 kABDepartmentProperty as string	148
* 6.2.47 kABEmailHomeLabel as string	149
* 6.2.48 kABEmailMobileMeLabel as string	149
* 6.2.49 kABEmailProperty as string	149
* 6.2.50 kABEmailWorkLabel as string	149
* 6.2.51 kABFatherLabel as string	149
* 6.2.52 kABFirstNamePhoneticProperty as string	149
* 6.2.53 kABFirstNameProperty as string	150
* 6.2.54 kABFriendLabel as string	150
* 6.2.55 kABGroupNameProperty as string	150
* 6.2.56 kABHomeLabel as string	151
* 6.2.57 kABHomePageLabel as string	151
* 6.2.58 kABHomePageProperty as string	151

* 6.2.59 kABICQHomeLabel as string	151
* 6.2.60 kABICQInstantProperty as string	151
* 6.2.61 kABICQWorkLabel as string	151
* 6.2.62 kABInsertedRecords as string	152
* 6.2.63 kABInstantMessageProperty as string	152
* 6.2.64 kABInstantMessageServiceAIM as string	152
* 6.2.65 kABInstantMessageServiceFacebook as string	152
* 6.2.66 kABInstantMessageServiceGaduGadu as string	152
* 6.2.67 kABInstantMessageServiceGoogleTalk as string	153
* 6.2.68 kABInstantMessageServiceICQ as string	153
* 6.2.69 kABInstantMessageServiceJabber as string	153
* 6.2.70 kABInstantMessageServiceKey as string	153
* 6.2.71 kABInstantMessageServiceMSN as string	153
* 6.2.72 kABInstantMessageServiceQQ as string	154
* 6.2.73 kABInstantMessageServiceSkype as string	154
* 6.2.74 kABInstantMessageServiceYahoo as string	154
* 6.2.75 kABInstantMessageUsernameKey as string	154
* 6.2.76 kABJabberHomeLabel as string	154
* 6.2.77 kABJabberInstantProperty as string	155
* 6.2.78 kABJabberWorkLabel as string	155
* 6.2.79 kABJobTitleProperty as string	155
* 6.2.80 kABLastNamePhoneticProperty as string	155
* 6.2.81 kABLastNameProperty as string	155
* 6.2.82 kABMaidenNameProperty as string	156
* 6.2.83 kABManagerLabel as string	156
* 6.2.84 kABMiddleNamePhoneticProperty as string	156
* 6.2.85 kABMiddleNameProperty as string	156
* 6.2.86 kABMobileMeLabel as string	157
* 6.2.87 kABModificationDateProperty as string	157
* 6.2.88 kABMotherLabel as string	157
* 6.2.89 kABMSNHomeLabel as string	157
* 6.2.90 kABMSNInstantProperty as string	157
* 6.2.91 kABMSNWorkLabel as string	157
* 6.2.92 kABNicknameProperty as string	158
* 6.2.93 kABNoteProperty as string	158
* 6.2.94 kABOrganizationProperty as string	158
* 6.2.95 kABOtherDateComponentsProperty as string	159
* 6.2.96 kABOtherDatesProperty as string	159
* 6.2.97 kABOtherLabel as string	159
* 6.2.98 kABParentLabel as string	159
* 6.2.99 kABPartnerLabel as string	159
* 6.2.100 kABPersonFlags as string	160

* 6.2.101 kABPhoneHomeFAXLabel as string	161
* 6.2.102 kABPhoneHomeLabel as string	161
* 6.2.103 kABPhoneiPhoneLabel as string	163
* 6.2.104 kABPhoneMainLabel as string	163
* 6.2.105 kABPhoneMobileLabel as string	163
* 6.2.106 kABPhonePagerLabel as string	163
* 6.2.107 kABPhoneProperty as string	163
* 6.2.108 kABPhoneWorkFAXLabel as string	163
* 6.2.109 kABPhoneWorkLabel as string	164
* 6.2.110 kABRelatedNamesProperty as string	164
* 6.2.111 kABSisterLabel as string	164
* 6.2.112 kABSocialProfileProperty as string	164
* 6.2.113 kABSocialProfileServiceFacebook as string	164
* 6.2.114 kABSocialProfileServiceFlickr as string	165
* 6.2.115 kABSocialProfileServiceKey as string	165
* 6.2.116 kABSocialProfileServiceLinkedIn as string	165
* 6.2.117 kABSocialProfileServiceMySpace as string	165
* 6.2.118 kABSocialProfileServiceSinaWeibo as string	165
* 6.2.119 kABSocialProfileServiceTencentWeibo as string	166
* 6.2.120 kABSocialProfileServiceTwitter as string	166
* 6.2.121 kABSocialProfileServiceYelp as string	166
* 6.2.122 kABSocialProfileURLKey as string	166
* 6.2.123 kABSocialProfileUserIdentifierKey as string	166
* 6.2.124 kABSocialProfileUsernameKey as string	167
* 6.2.125 kABSpouseLabel as string	167
* 6.2.126 kABSuffixProperty as string	167
* 6.2.127 kABTitleProperty as string	167
* 6.2.128 kABUIDProperty as string	167
* 6.2.129 kABUpdatedRecords as string	167
* 6.2.130 kABURLsProperty as string	168
* 6.2.131 kABWorkLabel as string	168
* 6.2.132 kABYahooHomeLabel as string	168
* 6.2.133 kABYahooInstantProperty as string	169
* 6.2.134 kABYahooWorkLabel as string	169
* 6.2.135 LocalizedPropertyOrLabel(propertyOrLabel as string) as string	169
* 6.2.136 NewPersonWithVCardRepresentation(data as memoryblock) as ABPersonMBS	169
* 6.2.137 people as ABPersonMBS()	169
* 6.2.138 peopleForAccount(account as ABAccountMBS) as ABPersonMBS()	170
* 6.2.139 peopleForEmail(email as string) as ABPersonMBS()	170
* 6.2.140 persistentAccounts as ABAccountMBS()	171
* 6.2.141 personForUniqueId(uniqueid as string) as ABPersonMBS	171

* 6.2.142	personForUniqueId(uniqueid as string, account as ABAccountMBS) as ABPersonMBS	172
* 6.2.143	recordClassFromUniqueId(uniqueid as string) as string	172
* 6.2.144	recordForUniqueId(uniqueid as string) as ABRecordMBS	172
* 6.2.145	recordForUniqueId(uniqueid as string, account as ABAccountMBS) as ABRecordMBS	173
* 6.2.146	recordsMatchingSearchElement(search as ABSearchElementMBS) as ABRecordMBS()	173
* 6.2.147	removeRecord(record as ABRecordMBS) as boolean	173
* 6.2.148	removeRecord(record as ABRecordMBS, byref error as NSErrorMBS) as boolean	174
* 6.2.149	save as boolean	174
* 6.2.150	save(byref error as NSErrorMBS) as boolean	174
* 6.2.151	searchElementForConjunction(conjunction as Integer, children() as ABSearchElementMBS) as ABSearchElementMBS	174
* 6.2.152	searchElementForGroupProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer) as ABSearchElementMBS	175
* 6.2.153	searchElementForPersonProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer) as ABSearchElementMBS	175
* 6.2.154	setMe(moi as ABPersonMBS)	176
* 6.2.155	sharedAddressbook as ABAddressBookMBS	176
* 6.2.156	sharedAddressbookMT as ABAddressBookMBS	176
* 6.2.158	defaultAccount as ABAccountMBS	177
* 6.2.159	defaultCountryCode as string	177
* 6.2.160	defaultNameOrdering as Integer	177
* 6.2.161	Handle as Integer	177
* 6.2.162	hasUnsavedChanges as boolean	178
* 6.2.163	owner as ABPersonMBS	178
* 6.2.165	DatabaseChanged(Externally as boolean, InsertedRecords() as string, UpdatedRecords() as string, DeletedRecords() as string)	178
– 6.3.1	class ABGroupMBS	181
* 6.3.3	addMember(group as ABPersonMBS) as boolean	181
* 6.3.4	addProperty(propertyName as string, type as Integer) as Integer	182
* 6.3.5	addSubgroup(group as ABGroupMBS) as boolean	183
* 6.3.6	Constructor	183
* 6.3.7	Constructor(addressBook as ABAddressBookMBS)	183
* 6.3.8	members as ABPersonMBS()	183
* 6.3.9	parentGroups as ABGroupMBS()	184
* 6.3.10	properties as string()	184
* 6.3.11	removeMember(group as ABPersonMBS) as boolean	184
* 6.3.12	removeProperties(properties() as string) as Integer	184
* 6.3.13	removeProperty(propertyName as string) as Integer	184
* 6.3.14	removeSubgroup(group as ABGroupMBS) as boolean	185

* 6.3.15	searchElementForProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer) as ABSearchElementMBS	185
* 6.3.16	subgroups as ABGroupMBS()	185
* 6.3.17	typeofProperty(propertyName as string) as Integer	185
* 6.3.19	distributionIdentifierForProperty(propertyName as string, person as ABPersonMBS) as String	186
– 6.4.1	class ABMultiValueMBS	187
* 6.4.3	Constructor	188
* 6.4.4	copy as ABMultiValueMBS	188
* 6.4.5	edit as ABMutableMultiValueMBS	188
* 6.4.6	identifierAtIndex(index as UInt32) as string	188
* 6.4.7	identifiers as string()	188
* 6.4.8	indexForIdentifier(identifier as string) as UInt32	189
* 6.4.9	indexForLabel(label as string) as UInt32	189
* 6.4.10	labelAtIndex(index as UInt32) as string	189
* 6.4.11	labelForIdentifier(identifier as string) as string	190
* 6.4.12	labels as string()	190
* 6.4.13	valueAtIndex(index as UInt32) as Variant	190
* 6.4.14	valueForIdentifier(identifier as string) as Variant	191
* 6.4.15	valueForLabel(label as string) as Variant	191
* 6.4.16	values as Variant()	192
* 6.4.18	Addressbook as ABAddressBookMBS	192
* 6.4.19	Content as Dictionary	192
* 6.4.20	count as Integer	192
* 6.4.21	Description as string	193
* 6.4.22	Handle as Integer	193
* 6.4.23	primaryIdentifier as string	193
* 6.4.24	propertyType as Integer	193
– 6.5.1	class ABMutableMultiValueMBS	195
* 6.5.3	addValue(value as Variant, label as string) as string	195
* 6.5.4	Constructor	195
* 6.5.5	insertValue(value as Variant, label as string, index as UInt32) as string	195
* 6.5.6	removeValueAndLabelAtIndex(index as UInt32) as boolean	196
* 6.5.7	replaceLabelAtIndex(index as UInt32, label as string) as boolean	196
* 6.5.8	replaceValueAtIndex(index as UInt32, value as Variant) as boolean	196
* 6.5.9	setPrimaryIdentifier(identifier as string) as boolean	196
– 6.6.1	class ABPersonMBS	197
* 6.6.3	addProperty(propertyName as string, type as Integer) as Integer	197
* 6.6.4	Constructor	198
* 6.6.5	Constructor(addressBook as ABAddressBookMBS)	198
* 6.6.6	Constructor(vCardData as Memoryblock)	198

* 6.6.7 EditInAddressbook as boolean	199
* 6.6.8 linkedPeople as ABPersonMBS()	199
* 6.6.9 parentGroups as ABGroupMBS()	199
* 6.6.10 properties as string()	199
* 6.6.11 removeProperties(properties() as string) as Integer	199
* 6.6.12 removeProperty(propertyName as string) as Integer	200
* 6.6.13 searchElementForProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer) as ABSearchElementMBS	200
* 6.6.14 setImageData(data as Memoryblock) as boolean	200
* 6.6.15 ShowInAddressbook as boolean	200
* 6.6.16 typeOfProperty(propertyName as string) as Integer	201
* 6.6.17 vCardRepresentation as Memoryblock	201
* 6.6.19 image as NSImageMBS	202
* 6.6.20 imageData as Memoryblock	202
– 6.7.1 class ABPickerMBS	204
* 6.7.3 AddProperty(propertyname as String)	204
* 6.7.4 ClearSearchField	204
* 6.7.5 Create	205
* 6.7.6 DeselectAll	205
* 6.7.7 DeselectGroup(group as ABGroupMBS)	205
* 6.7.8 DeselectIdentifier(person as ABPersonMBS, Identifier as String)	205
* 6.7.9 DeselectPerson(person as ABPersonMBS)	206
* 6.7.10 EditInAddressBook	206
* 6.7.11 InstallEvents(targetwindow as window)	206
* 6.7.12 Properties as string()	206
* 6.7.13 RemoveEvents	206
* 6.7.14 RemoveProperty(propertyname as String)	207
* 6.7.15 SelectedDictionaries as Dictionary()	207
* 6.7.16 SelectedGroups as ABGroupMBS()	207
* 6.7.17 SelectedIdentifiers(person as ABPersonMBS) as string()	207
* 6.7.18 SelectedRecords as ABRecordMBS()	207
* 6.7.19 SelectedStrings as String()	208
* 6.7.20 SelectedValues as Variant()	208
* 6.7.21 SelectGroup(group as ABGroupMBS, ExtendSelection as boolean)	208
* 6.7.22 SelectIdentifier(person as ABPersonMBS, Identifier as String, ExtendSelection as boolean)	208
* 6.7.23 SelectInAddressBook	208
* 6.7.24 SelectPerson(person as ABPersonMBS, ExtendSelection as boolean)	209
* 6.7.26 AllowGroupSelection as Boolean	209
* 6.7.27 AllowMultipleSelection as Boolean	209
* 6.7.28 AllowMultipleValueSelection as Boolean	209
* 6.7.29 AllowSingleValueSelection as Boolean	209

* 6.7.30 Available as Boolean	210
* 6.7.31 DisplayedProperty as String	210
* 6.7.32 Handle as Integer	210
* 6.7.33 Height as Single	210
* 6.7.34 Left as Single	211
* 6.7.35 Top as Single	211
* 6.7.36 Visible as Boolean	211
* 6.7.37 Width as Single	211
* 6.7.38 ColumnTitle(columnTitle as String) as String	211
* 6.7.40 DisplayedPropertyChanged	212
* 6.7.41 GroupDoubleClicked	212
* 6.7.42 GroupSelectionChanged	212
* 6.7.43 NameDoubleClicked	212
* 6.7.44 NameSelectionChanged	212
* 6.7.45 ValueSelectionChanged	213
– 6.8.1 class ABRecordMBS	214
* 6.8.3 Constructor	214
* 6.8.4 removeValueForProperty(propertyName as string) as boolean	214
* 6.8.5 setValue(value as Variant, propertyName as string) as boolean	214
* 6.8.6 setValue(value as Variant, propertyName as string, byref error as NSErrorMBS) as boolean	215
* 6.8.7 valueForProperty(PropertyName as string) as Variant	215
* 6.8.9 account as ABAccountMBS	216
* 6.8.10 Addressbook as ABAddressBookMBS	216
* 6.8.11 Description as string	216
* 6.8.12 DisplayName as string	216
* 6.8.13 Handle as Integer	217
* 6.8.14 isReadOnly as boolean	217
* 6.8.15 uniqueId as string	217
– 6.9.1 class ABSearchElementMBS	219
* 6.9.3 Constructor	220
* 6.9.4 matchesRecord(record as ABRecordMBS) as boolean	220
* 6.9.5 searchElementForConjunction(conjunction as Integer, children() as ABSearchElementMBS) as ABSearchElementMBS	220
* 6.9.7 Addressbook as ABAddressBookMBS	221
* 6.9.8 Description as string	221
* 6.9.9 Handle as Integer	221

	11
• 8 Cocoa	271
– 22.1.1 class Application	1081
* 22.1.3 NSApplicationMBS as NSApplicationMBS	1081

- **8 Cocoa** 271
 - 22.2.1 class ConsoleApplication 1082
 - * 22.2.3 NSApplicationMBS as NSApplicationMBS 1082

	13
• 9 Cocoa Controls	453
– 16.1.1 class Control	939
* 16.1.3 NSControlMBS as NSControlMBS	939
– 9.1.1 class CustomNSSearchFieldMBS	453
* 9.1.3 Constructor	453
* 9.1.4 Constructor(Handle as Integer)	454
* 9.1.5 Constructor(left as Double, top as Double, width as Double, height as Double)	454
* 9.1.6 Destructor	454
* 9.1.8 acceptsFirstMouse(e as NSEventMBS) as boolean	454
* 9.1.9 acceptsFirstResponder as boolean	455
* 9.1.10 becomeFirstResponder as boolean	455
* 9.1.11 beginGestureWithEvent(e as NSEventMBS) as boolean	455
* 9.1.12 canBecomeKeyView as boolean	455
* 9.1.13 Close	455
* 9.1.14 concludeDragOperation(sender as NSDraggingInfoMBS)	456
* 9.1.15 draggingEnded(sender as NSDraggingInfoMBS)	456
* 9.1.16 draggingEntered(sender as NSDraggingInfoMBS) as Integer	456
* 9.1.17 draggingExited(sender as NSDraggingInfoMBS)	457
* 9.1.18 draggingSessionEndedAtPoint(session as NSDraggingSessionMBS, screenPoint as NS-PointMBS, operation as Integer)	457
* 9.1.19 draggingSessionMovedToPoint(session as NSDraggingSessionMBS, screenPoint as NS-PointMBS)	457
* 9.1.20 draggingSessionSourceOperationMaskForDraggingContext(session as NSDraggingSessionMBS, context as Integer) as Integer	458
* 9.1.21 draggingSessionWillBeginAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)	458
* 9.1.22 draggingUpdated(sender as NSDraggingInfoMBS) as Integer	458
* 9.1.23 endGestureWithEvent(e as NSEventMBS) as boolean	459
* 9.1.24 ignoreModifierKeysForDraggingSession(session as NSDraggingSessionMBS) as boolean	459
* 9.1.25 isOpaque as boolean	460
* 9.1.26 keyDown(e as NSEventMBS) as boolean	460
* 9.1.27 keyUp(e as NSEventMBS) as boolean	460
* 9.1.28 magnifyWithEvent(e as NSEventMBS) as boolean	460
* 9.1.29 menuForEvent(e as NSEventMBS, defaultMenu as NSMenuMBS) as NSMenuMBS	460
* 9.1.30 mouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	461
* 9.1.31 mouseDownCanMoveWindow as boolean	461
* 9.1.32 mouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	461
* 9.1.33 mouseEntered(e as NSEventMBS, x as Double, y as Double) as boolean	461
* 9.1.34 mouseExited(e as NSEventMBS, x as Double, y as Double) as boolean	462
* 9.1.35 mouseMoved(e as NSEventMBS, x as Double, y as Double) as boolean	462

* 9.1.36	mouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	462
* 9.1.37	Open	462
* 9.1.38	otherMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	462
* 9.1.39	otherMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	463
* 9.1.40	otherMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	463
* 9.1.41	performDragOperation(sender as NSDraggingInfoMBS) as boolean	463
* 9.1.42	prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean	463
* 9.1.43	pressureChange(e as NSEventMBS) as boolean	464
* 9.1.44	resignFirstResponder as boolean	464
* 9.1.45	rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	464
* 9.1.46	rightMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	464
* 9.1.47	rightMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	465
* 9.1.48	rotateWithEvent(e as NSEventMBS) as boolean	465
* 9.1.49	scrollWheel(e as NSEventMBS) as boolean	465
* 9.1.50	swipeWithEvent(e as NSEventMBS) as boolean	465
* 9.1.51	updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)	465
* 9.1.52	viewDidMoveToWindow	466
* 9.1.53	wantsPeriodicDraggingUpdates as boolean	466
– 9.2.1	class CustomNSTextFieldCellMBS	467
* 9.2.3	Constructor	467
* 9.2.4	superDrawWithFrame(frame as NSRectMBS, view as NSViewMBS)	467
* 9.2.6	cellSize(size as NSSizeMBS) as NSSizeMBS	467
* 9.2.7	Clone(clonedCell as NSTextFieldCellMBS) as CustomNSTextFieldCellMBS	467
* 9.2.8	didDrawWithFrame(cellFrame as NSRectMBS, controlView as NSViewMBS)	468
* 9.2.9	drawWithFrame(cellFrame as NSRectMBS, controlView as NSViewMBS) as boolean	468
* 9.2.10	fieldEditorForView(controlView as NSViewMBS) as NSTextViewMBS	468
* 9.2.11	imageRectForBounds(rect as NSRectMBS) as NSRectMBS	469
* 9.2.12	selectWithFrame(rect as NSRectMBS, controlView as NSViewMBS, text as NSTextMBS, theDelegate as Variant, selStart as Integer, selLength as Integer) as boolean	469
* 9.2.13	setUpFieldEditorAttributes(textObj as NSTextMBS, superFieldEditor as NSTextMBS) as NSTextMBS	469
* 9.2.14	titleRectForBounds(rect as NSRectMBS) as NSRectMBS	469
– 9.3.1	class CustomNSTextFieldMBS	470
* 9.3.3	Constructor	470
* 9.3.4	Constructor(Handle as Integer)	470
* 9.3.5	Constructor(left as Double, top as Double, width as Double, height as Double)	470
* 9.3.6	Destructor	471
* 9.3.8	acceptsFirstMouse(e as NSEventMBS) as boolean	471
* 9.3.9	acceptsFirstResponder as boolean	471
* 9.3.10	becomeFirstResponder as boolean	471

* 9.3.11 beginGestureWithEvent(e as NSEventMBS) as boolean	472
* 9.3.12 canBecomeKeyView as boolean	472
* 9.3.13 Close	472
* 9.3.14 concludeDragOperation(sender as NSDraggingInfoMBS)	472
* 9.3.15 draggingEnded(sender as NSDraggingInfoMBS)	473
* 9.3.16 draggingEntered(sender as NSDraggingInfoMBS) as Integer	473
* 9.3.17 draggingExited(sender as NSDraggingInfoMBS)	473
* 9.3.18 draggingSessionEndedAtPoint(session as NSDraggingSessionMBS, screenPoint as NS-PointMBS, operation as Integer)	474
* 9.3.19 draggingSessionMovedToPoint(session as NSDraggingSessionMBS, screenPoint as NS-PointMBS)	474
* 9.3.20 draggingSessionSourceOperationMaskForDraggingContext(session as NSDraggingSessionMBS, context as Integer) as Integer	474
* 9.3.21 draggingSessionWillBeginAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)	475
* 9.3.22 draggingUpdated(sender as NSDraggingInfoMBS) as Integer	475
* 9.3.23 endGestureWithEvent(e as NSEventMBS) as boolean	476
* 9.3.24 ignoreModifierKeysForDraggingSession(session as NSDraggingSessionMBS) as boolean	476
* 9.3.25 isOpaque as boolean	476
* 9.3.26 keyDown(e as NSEventMBS) as boolean	476
* 9.3.27 keyUp(e as NSEventMBS) as boolean	476
* 9.3.28 magnifyWithEvent(e as NSEventMBS) as boolean	477
* 9.3.29 menuForEvent(e as NSEventMBS, defaultMenu as NSMenuMBS) as NSMenuMBS	477
* 9.3.30 mouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	477
* 9.3.31 mouseDownCanMoveWindow as boolean	477
* 9.3.32 mouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	478
* 9.3.33 mouseEntered(e as NSEventMBS, x as Double, y as Double) as boolean	478
* 9.3.34 mouseExited(e as NSEventMBS, x as Double, y as Double) as boolean	478
* 9.3.35 mouseMoved(e as NSEventMBS, x as Double, y as Double) as boolean	478
* 9.3.36 mouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	478
* 9.3.37 Open	479
* 9.3.38 otherMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	479
* 9.3.39 otherMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	479
* 9.3.40 otherMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	479
* 9.3.41 performDragOperation(sender as NSDraggingInfoMBS) as boolean	479
* 9.3.42 prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean	480
* 9.3.43 pressureChange(e as NSEventMBS) as boolean	480
* 9.3.44 resignFirstResponder as boolean	480
* 9.3.45 rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	481
* 9.3.46 rightMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	481
* 9.3.47 rightMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	481

* 9.3.48 rotateWithEvent(e as NSEventMBS) as boolean	481
* 9.3.49 scrollWheel(e as NSEventMBS) as boolean	481
* 9.3.50 swipeWithEvent(e as NSEventMBS) as boolean	482
* 9.3.51 updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)	482
* 9.3.52 viewDidMoveToWindow	482
* 9.3.53 wantsPeriodicDraggingUpdates as boolean	483
– 9.4.1 class CustomNSTextViewMBS	484
* 9.4.3 Constructor	484
* 9.4.4 Constructor(Handle as Integer)	484
* 9.4.5 Constructor(left as Double, top as Double, width as Double, height as Double)	484
* 9.4.6 Destructor	485
* 9.4.8 acceptsFirstMouse(e as NSEventMBS) as boolean	485
* 9.4.9 acceptsFirstResponder as boolean	485
* 9.4.10 becomeFirstResponder as boolean	485
* 9.4.11 beginGestureWithEvent(e as NSEventMBS) as boolean	486
* 9.4.12 canBecomeKeyView as boolean	486
* 9.4.13 Close	486
* 9.4.14 concludeDragOperation(sender as NSDraggingInfoMBS)	486
* 9.4.15 draggingEnded(sender as NSDraggingInfoMBS)	487
* 9.4.16 draggingEntered(sender as NSDraggingInfoMBS) as Integer	487
* 9.4.17 draggingExited(sender as NSDraggingInfoMBS)	487
* 9.4.18 draggingSessionEndedAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS, operation as Integer)	488
* 9.4.19 draggingSessionMovedToPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)	488
* 9.4.20 draggingSessionSourceOperationMaskForDraggingContext(session as NSDraggingSessionMBS, context as Integer) as Integer	488
* 9.4.21 draggingSessionWillBeginAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)	489
* 9.4.22 draggingUpdated(sender as NSDraggingInfoMBS) as Integer	489
* 9.4.23 endGestureWithEvent(e as NSEventMBS) as boolean	490
* 9.4.24 ignoreModifierKeysForDraggingSession(session as NSDraggingSessionMBS) as boolean	490
* 9.4.25 isOpaque as boolean	490
* 9.4.26 keyDown(e as NSEventMBS) as boolean	490
* 9.4.27 keyUp(e as NSEventMBS) as boolean	490
* 9.4.28 magnifyWithEvent(e as NSEventMBS) as boolean	491
* 9.4.29 menuForEvent(e as NSEventMBS, defaultMenu as NSMenuMBS) as NSMenuMBS	491
* 9.4.30 mouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	491
* 9.4.31 mouseDownCanMoveWindow as boolean	491
* 9.4.32 mouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	492
* 9.4.33 mouseEntered(e as NSEventMBS, x as Double, y as Double) as boolean	492

* 9.4.34 mouseExited(e as NSEventMBS, x as Double, y as Double) as boolean	492
* 9.4.35 mouseMoved(e as NSEventMBS, x as Double, y as Double) as boolean	492
* 9.4.36 mouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	492
* 9.4.37 Open	493
* 9.4.38 otherMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	493
* 9.4.39 otherMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	493
* 9.4.40 otherMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	493
* 9.4.41 performDragOperation(sender as NSDraggingInfoMBS) as boolean	493
* 9.4.42 prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean	494
* 9.4.43 pressureChange(e as NSEventMBS) as boolean	494
* 9.4.44 resignFirstResponder as boolean	494
* 9.4.45 rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean	495
* 9.4.46 rightMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean	495
* 9.4.47 rightMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean	495
* 9.4.48 rotateWithEvent(e as NSEventMBS) as boolean	495
* 9.4.49 scrollWheel(e as NSEventMBS) as boolean	495
* 9.4.50 swipeWithEvent(e as NSEventMBS) as boolean	496
* 9.4.51 updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)	496
* 9.4.52 viewDidMoveToWindow	496
* 9.4.53 wantsPeriodicDraggingUpdates as boolean	497

- **8 Cocoa** 271
 - 22.3.1 class DesktopApplication 1083
 - * 22.3.3 NSApplicationMBS as NSApplicationMBS 1083

	19
• 9 Cocoa Controls	453
– 16.2.1 class DesktopControl	940
* 16.2.3 NSControlMBS as NSControlMBS	940

• 9 Cocoa Controls	453
– 16.3.1 class DesktopLabel	941
* 16.3.3 NSTextFieldMBS as NSTextFieldMBS	941
– 9.5.1 control DesktopNSPathControlControlMBS	498
* 9.5.3 View as NSPathControlMBS	498
* 9.5.5 Action	498
* 9.5.6 BoundsChanged	498
* 9.5.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	499
* 9.5.8 DoubleClick	499
* 9.5.9 FocusLost	499
* 9.5.10 FocusReceived	499
* 9.5.11 FrameChanged	500
* 9.5.12 MenuBarSelected	500
* 9.5.13 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	500
* 9.5.14 MouseDrag(x as Integer, y as Integer)	500
* 9.5.15 MouseUp(x As Integer, y As Integer)	501
* 9.5.16 ScaleFactorChanged(NewFactor as double)	501
* 9.5.17 willDisplayOpenPanel(openPanel as Variant)	501
* 9.5.18 willPopUpMenu(menu as Variant)	501
* 9.5.19 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	502
– 9.6.1 control DesktopNSSearchFieldControlMBS	503
* 9.6.3 View as NSSearchFieldMBS	503
* 9.6.5 Action	503
* 9.6.6 BoundsChanged	504
* 9.6.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	504
* 9.6.8 FocusLost	504
* 9.6.9 FocusReceived	504
* 9.6.10 FrameChanged	505
* 9.6.11 MenuBarSelected	505
* 9.6.12 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	505
* 9.6.13 MouseDrag(x as Integer, y as Integer)	505
* 9.6.14 MouseUp(x as Integer, y as Integer)	506
* 9.6.15 ScaleFactorChanged(NewFactor as Double)	506
* 9.6.16 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	506
* 9.6.17 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	506
* 9.6.18 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	507
* 9.6.19 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean	507
* 9.6.20 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean	507
* 9.6.21 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	507

– 9.7.1 control DesktopNSSecureTextFieldControlMBS	508
* 9.7.3 echosBullets as Boolean	508
* 9.7.4 View as NSSecureTextFieldMBS	508
* 9.7.6 Action	509
* 9.7.7 BoundsChanged	509
* 9.7.8 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	509
* 9.7.9 FocusLost	509
* 9.7.10 FocusReceived	509
* 9.7.11 FrameChanged	510
* 9.7.12 MenuBarSelected	510
* 9.7.13MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	510
* 9.7.14 MouseDrag(x as Integer, y as Integer)	510
* 9.7.15 MouseUp(x as Integer, y as Integer)	511
* 9.7.16 ScaleFactorChanged(NewFactor as Double)	511
* 9.7.17 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	511
* 9.7.18 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	511
* 9.7.19 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	512
* 9.7.20 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean	512
* 9.7.21 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean	512
* 9.7.22 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	512
– 9.8.1 control DesktopNSSegmentedControlControlMBS	514
* 9.8.3 View as NSSegmentedControlMBS	514
* 9.8.5 Action	514
* 9.8.6 BoundsChanged	514
* 9.8.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	515
* 9.8.8 FocusLost	515
* 9.8.9 FocusReceived	515
* 9.8.10 FrameChanged	515
* 9.8.11 MenuBarSelected	515
* 9.8.12MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	516
* 9.8.13 MouseDrag(x as Integer, y as Integer)	516
* 9.8.14 MouseUp(x As Integer, y As Integer)	516
* 9.8.15 ScaleFactorChanged(NewFactor as double)	516
* 9.8.16 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	517
– 9.9.1 control DesktopNSSplitViewControlMBS	518
* 9.9.3 View as NSSplitViewMBS	518
* 9.9.5 BoundsChanged	518
* 9.9.6 canCollapseSubview(subview as NSViewMBS) as Boolean	518
* 9.9.7 constrainMaxCoordinate(proposedMaximumPosition as double, dividerIndex as Integer) as Double	519

* 9.9.8 constrainMinCoordinate(proposedMinimumPosition as double, dividerIndex as Integer) as Double	519
* 9.9.9 constrainSplitPosition(proposedPosition as double, dividerIndex as Integer) as Double	520
* 9.9.10 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	520
* 9.9.11 drawDivider(graphics as NSGraphicsMBS, Rect as NSRectMBS) as Boolean	520
* 9.9.12 FocusLost	521
* 9.9.13 FocusReceived	521
* 9.9.14 FrameChanged	522
* 9.9.15 MenuBarSelected	522
* 9.9.16MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	522
* 9.9.17 MouseDrag(x as Integer, y as Integer)	522
* 9.9.18 MouseUp(x As Integer, y As Integer)	523
* 9.9.19 resizeSubviewsWithOldSize(oldSize as NSSizeMBS)	523
* 9.9.20 ScaleFactorChanged(NewFactor as double)	523
* 9.9.21 shouldAdjustSizeOfSubview(view as NSViewMBS) as Boolean	523
* 9.9.22 shouldHideDivider(dividerIndex as Integer) as Boolean	524
* 9.9.23 splitViewDidResizeSubviews	524
* 9.9.24 splitViewWillResizeSubviews	524
* 9.9.25 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	524
– 9.10.1 control DesktopNSTextFieldControlMBS	526
* 9.10.3 View as NSTextFieldMBS	526
* 9.10.5 Action	526
* 9.10.6 BoundsChanged	526
* 9.10.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	527
* 9.10.8 FocusLost	527
* 9.10.9 FocusReceived	527
* 9.10.10 FrameChanged	527
* 9.10.11 MenuBarSelected	528
* 9.10.12MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	528
* 9.10.13 MouseDrag(x as Integer, y as Integer)	528
* 9.10.14 MouseUp(x as Integer, y as Integer)	528
* 9.10.15 ScaleFactorChanged(NewFactor as Double)	529
* 9.10.16 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	529
* 9.10.17 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	529
* 9.10.18 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	529
* 9.10.19 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean	530
* 9.10.20 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean	530
* 9.10.21 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	530
– 9.11.1 control DesktopNSTextViewControlMBS	531

	23
* 9.11.3 AcceptTabs as Boolean	531
* 9.11.4 Scrollview as Variant	531
* 9.11.5 View as NSTextViewMBS	531
* 9.11.7 BoundsChanged	532
* 9.11.8 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	532
* 9.11.9 FocusLost	532
* 9.11.10 FocusReceived	532
* 9.11.11 FrameChanged	533
* 9.11.12 MenuBarSelected	533
* 9.11.13MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	533
* 9.11.14 MouseDrag(x as Integer, y as Integer)	533
* 9.11.15 MouseUp(x as Integer, y as Integer)	534
* 9.11.16 ScaleFactorChanged(NewFactor as Double)	534
* 9.11.17 shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string) as boolean	534
* 9.11.18 textDidBeginEditing	534
* 9.11.19 textDidChange	534
* 9.11.20 textDidEndEditing	535
* 9.11.21 textShouldBeginEditing as boolean	535
* 9.11.22 textShouldEndEditing as boolean	535
* 9.11.23 textViewDidChangeSelection	535
* 9.11.24 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	535
– 9.12.1 class DesktopSearchField	536
* 9.12.3 NSSearchFieldMBS as NSSearchFieldMBS	536
– 9.13.1 class DesktopSegmentedButton	537
* 9.13.3 NSSegmentedControlMBS as NSSegmentedControlMBS	537
– 9.14.1 class DesktopTabPanel	538
* 9.14.3 NSTabViewMBS as NSTabViewMBS	538
– 9.15.1 class DesktopTextArea	539
* 9.15.3 NSTextFieldMBS as NSTextFieldMBS	539
* 9.15.4 NSTextViewMBS as NSTextViewMBS	540
* 9.15.6 RTFDataMBS as Memoryblock	540

- **9 Cocoa Controls** 453
 - 16.4.1 class DesktopTextField 942
 - * 16.4.3 NSTextFieldMBS as NSTextFieldMBS 942
 - * 16.4.4 NSTextViewMBS as NSTextViewMBS 942

	25
• 17 Drag & Drop	947
– 17.1.1 class DragItem	947
* 17.1.3 NSDraggingInfoMBS as NSDraggingInfoMBS	947

• 19 Instant Message	1025
– 19.1.1 class IMServiceMBS	1025
* 19.1.3 imageFileForStatus(status as Integer) as folderitem	1026
* 19.1.4 imageNameForStatus(status as Integer) as string	1026
* 19.1.5 imageURLForStatus(status as Integer) as string	1026
* 19.1.6 IMCapabilityAudioConference as string	1026
* 19.1.7 IMCapabilityDirectIM as string	1026
* 19.1.8 IMCapabilityFileSharing as string	1027
* 19.1.9 IMCapabilityFileTransfer as string	1027
* 19.1.10 IMCapabilityText as string	1027
* 19.1.11 IMCapabilityVideoConference as string	1027
* 19.1.12 IMPersonAVBusyKey as string	1027
* 19.1.13 IMPersonCapabilitiesKey as string	1027
* 19.1.14 IMPersonEmailKey as string	1028
* 19.1.15 IMPersonFirstNameKey as string	1028
* 19.1.16 IMPersonIdleSinceKey as string	1028
* 19.1.17 IMPersonLastNameKey as string	1028
* 19.1.18 IMPersonPictureDataKey as string	1029
* 19.1.19 IMPersonScreenNameKey as string	1029
* 19.1.20 IMPersonServiceNameKey as string	1029
* 19.1.21 IMPersonStatusKey as string	1029
* 19.1.22 IMPersonStatusMessageKey as string	1029
* 19.1.23 infoForAllScreenNames as dictionary()	1030
* 19.1.24 infoForPreferredScreenNames as dictionary()	1030
* 19.1.25 infoForScreenName(name as string) as dictionary	1030
* 19.1.26 LocalizedName as String	1030
* 19.1.27 LocalizedShortName as String	1031
* 19.1.28 Name as String	1031
* 19.1.29 peopleWithScreenName(screenName as string) as ABPersonMBS()	1032
* 19.1.30 screenNamesForPerson(person as ABPersonMBS) as string()	1032
* 19.1.31 Status as Integer	1032
* 19.1.33 Handle as Integer	1033
– 19.2.1 class InstantMessageMBS	1034
* 19.2.3 allServices as IMServiceMBS()	1034
* 19.2.4 Available as boolean	1034
* 19.2.5 imageFileForStatus(status as Integer) as folderitem	1034
* 19.2.6 imageNameForStatus(status as Integer) as string	1035
* 19.2.7 imageURLForStatus(status as Integer) as string	1035
* 19.2.8 myIdleTime as Double	1036
* 19.2.9 myStatus as Integer	1036
* 19.2.10 notificationCenter as NSNotificationCenterMBS	1036

* 19.2.11 serviceWithName(name as string) as IMServiceMBS	1036
* 19.2.13 MyStatusChanged	1037
* 19.2.14 PersonInfoChanged(info as dictionary)	1037
* 19.2.15 PersonStatusChanged(info as dictionary)	1037
* 19.2.16 ServiceStatusChanged	1037
* 19.2.17 StatusImagesChangedAppearance	1037

- **9 Cocoa Controls** 453
 - 16.5.1 class Label 943
 - * 16.5.3 NSTextFieldMBS as NSTextFieldMBS 943

	29
• 5 Accessibility	131
– ?? Globals	??
* 5.1.1 InstallNSAccessibilityPatchMBS	131

• 9 Cocoa Controls	453
– 9.16.1 class NSActionCellMBS	541
* 9.16.3 Constructor(image as NSImageMBS)	541
* 9.16.4 Constructor(text as string)	541

	31
• 8 Cocoa	271
– 8.1.1 class NSAlertMBS	271
* 8.1.3 addButtonWithTitle(title as string) as Variant	272
* 8.1.4 alertWithError(error as NSErrorMBS) as NSAlertMBS	273
* 8.1.5 alertWithMessageText(MessageText as string, defaultButton as string = "", alternateButton as string = "", otherButton as string = "", informativeText as string = "") as NSAlertMBS	273
* 8.1.6 beginSheetModalForWindow(win as DesktopWindow)	273
* 8.1.7 beginSheetModalForWindow(win as NSWindowMBS)	273
* 8.1.8 beginSheetModalForWindow(win as window)	274
* 8.1.9 buttons as Variant()	274
* 8.1.10 close	275
* 8.1.11 Constructor	275
* 8.1.12 Destructor	275
* 8.1.13 layout	275
* 8.1.14 runModal as Integer	275
* 8.1.15 SetTextAlignment(value as Integer)	276
* 8.1.17 accessoryView as NSViewMBS	276
* 8.1.18 alertStyle as Integer	276
* 8.1.19 helpAnchor as String	277
* 8.1.20 icon as NSImageMBS	277
* 8.1.21 informativeText as String	277
* 8.1.22 messageText as String	277
* 8.1.23 showsHelp as Boolean	277
* 8.1.24 ShowsSuppressionButton as Boolean	278
* 8.1.25 suppressionButton as Variant	278
* 8.1.26 TimedOut as Boolean	279
* 8.1.27 timeOut as Double	279
* 8.1.28 window as Variant	279
* 8.1.30 SheetDidEnd(returnCode as Integer)	279
* 8.1.31 ShowHelp as boolean	279

• 7 Apple Script	223
– 7.1.1 class NSAppleEventDescriptorMBS	223
* 7.1.3 appleEventWithEventClass(eventClass as string, eventID as string, targetDescriptor as NSAppleEventDescriptorMBS, returnID as Int16, transactionID as UInt32) as NSAppleEventDescriptorMBS	224
* 7.1.4 attributeDescriptorForKeyword(keyword as string) as NSAppleEventDescriptorMBS	225
* 7.1.5 coerceToDescriptorType(descriptorType as string) as NSAppleEventDescriptorMBS	225
* 7.1.6 Constructor	226
* 7.1.7 copy as NSAppleEventDescriptorMBS	226
* 7.1.8 currentProcessDescriptor as NSAppleEventDescriptorMBS	226
* 7.1.9 descriptorAtIndex(index as Integer) as NSAppleEventDescriptorMBS	226
* 7.1.10 descriptorForKeyword(keyword as string) as NSAppleEventDescriptorMBS	227
* 7.1.11 descriptorWithAlias(item as folderitem) as NSAppleEventDescriptorMBS	227
* 7.1.12 descriptorWithApplicationURL(fileURL as string) as NSAppleEventDescriptorMBS	228
* 7.1.13 descriptorWithApplicationURL(item as folderitem) as NSAppleEventDescriptorMBS	228
* 7.1.14 descriptorWithBoolean(value as Boolean) as NSAppleEventDescriptorMBS	229
* 7.1.15 descriptorWithBundleIdentifier(BundleID as String) as NSAppleEventDescriptorMBS	229
* 7.1.16 descriptorWithCurrentProcessSerialNumber as NSAppleEventDescriptorMBS	229
* 7.1.17 descriptorWithDate(value as date) as NSAppleEventDescriptorMBS	230
* 7.1.18 descriptorWithDateTime(value as dateTime) as NSAppleEventDescriptorMBS	230
* 7.1.19 descriptorWithDescriptorType(descriptorType as string, data as memoryblock) as NSAppleEventDescriptorMBS	230
* 7.1.20 descriptorWithDescriptorType(descriptorType as string, data as memoryblock, offset as UInt32, length as UInt32) as NSAppleEventDescriptorMBS	231
* 7.1.21 descriptorWithDouble(value as Double) as NSAppleEventDescriptorMBS	231
* 7.1.22 descriptorWithEnumCode(enumerator as string) as NSAppleEventDescriptorMBS	231
* 7.1.23 descriptorWithFileURL(fileURL as string) as NSAppleEventDescriptorMBS	232
* 7.1.24 descriptorWithFileURL(item as folderitem) as NSAppleEventDescriptorMBS	232
* 7.1.25 descriptorWithInt16(value as Int16) as NSAppleEventDescriptorMBS	232
* 7.1.26 descriptorWithInt32(value as Int32) as NSAppleEventDescriptorMBS	233
* 7.1.27 descriptorWithProcessIdentifier(PID as Integer) as NSAppleEventDescriptorMBS	233
* 7.1.28 descriptorWithSingle(value as single) as NSAppleEventDescriptorMBS	234
* 7.1.29 descriptorWithString(text as string) as NSAppleEventDescriptorMBS	234
* 7.1.30 descriptorWithTypeCode(typeCode as string) as NSAppleEventDescriptorMBS	234
* 7.1.31 descriptorWithUInt32(value as UInt32) as NSAppleEventDescriptorMBS	235
* 7.1.32 insertDescriptor(descriptor as NSAppleEventDescriptorMBS, index as Integer)	235
* 7.1.33 keywordForDescriptorAtIndex(index as Integer) as string	235
* 7.1.34 listDescriptor as NSAppleEventDescriptorMBS	236

* 7.1.35 nullDescriptor as NSAppleEventDescriptorMBS	236
* 7.1.36 paramDescriptorForKeyword(keyword as string) as NSAppleEventDescriptorMBS	236
* 7.1.37 print	237
* 7.1.38 recordDescriptor as NSAppleEventDescriptorMBS	237
* 7.1.39 removeDescriptorAtIndex(index as Integer)	237
* 7.1.40 removeDescriptorWithKeyword(keyword as string)	237
* 7.1.41 removeParamDescriptorWithKeyword(keyword as string)	238
* 7.1.42 send(options as Integer, timeoutInSeconds as Double, byref error as NSErrorMBS) as NSAppleEventDescriptorMBS	238
* 7.1.43 setAttributeDescriptor(descriptor as NSAppleEventDescriptorMBS, keyword as string)	239
* 7.1.44 setDescription(descriptor as NSAppleEventDescriptorMBS, keyword as string)	239
* 7.1.45 setParamDescriptor(descriptor as NSAppleEventDescriptorMBS, keyword as string)	239
* 7.1.47 aeDesc as Ptr	240
* 7.1.48 applicationURLValue as String	240
* 7.1.49 booleanValue as boolean	240
* 7.1.50 bundleIDValue as String	241
* 7.1.51 data as Memoryblock	241
* 7.1.52 dateTimeValue as DateTime	241
* 7.1.53 dateValue as date	241
* 7.1.54 description as string	242
* 7.1.55 descriptorType as string	242
* 7.1.56 doubleValue as Double	242
* 7.1.57 enumCodeValue as string	243
* 7.1.58 eventClass as string	243
* 7.1.59 eventID as string	243
* 7.1.60 fileURLValue as String	244
* 7.1.61 Handle as Integer	244
* 7.1.62 int16Value as Int16	244
* 7.1.63 int32Value as Int32	244
* 7.1.64 isRecordDescriptor as Boolean	245
* 7.1.65 numberOfItems as Integer	245
* 7.1.66 processIDValue as Integer	246
* 7.1.67 returnID as Int16	246
* 7.1.68 singleValue as single	246
* 7.1.69 stringValue as string	247
* 7.1.70 transactionID as Int32	247
* 7.1.71 typeCodeValue as string	247
* 7.1.72 UInt32Value as UInt32	247
– 7.2.1 class NSAppleEventHandlerMBS	249
* 7.2.3 Constructor	249

* 7.2.4 Destructor	249
* 7.2.6 Handle as Integer	249
* 7.2.8 handleAppleEvent(theEvent as NSAppleEventDescriptorMBS, replyEvent as NSAppleEventDescriptorMBS)	249
– 7.3.1 class NSAppleEventManagerMBS	251
* 7.3.3 appleEventForSuspensionID(id as NSAppleEventManagerSuspensionIDMBS) as NSAppleEventDescriptorMBS	251
* 7.3.4 Constructor	251
* 7.3.5 currentAppleEvent as NSAppleEventDescriptorMBS	252
* 7.3.6 currentReplyAppleEvent as NSAppleEventDescriptorMBS	252
* 7.3.7 NSAppleEventManagerWillProcessFirstEventNotification as string	252
* 7.3.8 removeEventHandlerForEventClass(eventClass as string, eventID as string)	252
* 7.3.9 replyAppleEventForSuspensionID(id as NSAppleEventManagerSuspensionIDMBS) as NSAppleEventDescriptorMBS	253
* 7.3.10 resumeWithSuspensionID(id as NSAppleEventManagerSuspensionIDMBS)	253
* 7.3.11 setCurrentAppleEventAndReplyEventWithSuspensionID(id as NSAppleEventManagerSuspensionIDMBS)	253
* 7.3.12 setEventHandler(handler as NSAppleEventHandlerMBS, eventClass as string, eventID as string)	253
* 7.3.13 suspendCurrentAppleEvent as NSAppleEventManagerSuspensionIDMBS	254
* 7.3.15 Handle as Integer	254
– 7.4.1 class NSAppleEventManagerSuspensionIDMBS	255
* 7.4.3 Constructor	255
* 7.4.5 Handle as Integer	255
– 7.5.1 class NSAppleScriptMBS	256
* 7.5.3 compile as boolean	257
* 7.5.4 compile(byref error as dictionary) as boolean	258
* 7.5.5 Constructor(file as folderitem, byref error as Dictionary)	258
* 7.5.6 Constructor(source as string)	259
* 7.5.7 Constructor(sourceLines() as string)	259
* 7.5.8 Constructor(URL as string, byref error as Dictionary)	260
* 7.5.9 copy as NSAppleScriptMBS	261
* 7.5.10 DeterminePermissionToAutomateTarget(target as NSAppleEventDescriptorMBS, theAEEEventClass as string = "****", theAEEEventID as String = "****", askUserIfNeeded as boolean) as Integer	261
* 7.5.11 execute as NSAppleEventDescriptorMBS	262
* 7.5.12 execute(byref error as dictionary) as NSAppleEventDescriptorMBS	263
* 7.5.13 executeAppleEvent(event as NSAppleEventDescriptorMBS, byref error as dictionary) as NSAppleEventDescriptorMBS	263
* 7.5.14 executeSubroutine(Name as String, parameters() as NSAppleEventDescriptorMBS, byref error as dictionary) as NSAppleEventDescriptorMBS	264
* 7.5.15 NSAppleScriptErrorAppName as string	265

	35
* 7.5.16 NSErrorBriefMessage as string	265
* 7.5.17 NSErrorMessage as string	266
* 7.5.18 NSErrorNumber as string	266
* 7.5.19 NSErrorRange as string	267
* 7.5.20 properties as string()	267
* 7.5.21 setValueDescriptorForProperty(propertyName as string, value as NSErrorDescriptorMBS) as boolean	268
* 7.5.22 valueDescriptorForProperty(propertyName as string) as NSErrorDescriptorMBS	268
* 7.5.24 Handle as Integer	268
* 7.5.25 isCompiled as boolean	269
* 7.5.26 richTextSource as NSAttributedStringMBS	269
* 7.5.27 source as string	270

• 8 Cocoa	271
– 8.2.1 class <code>NSApplicationDelegateMBS</code>	281
* 8.2.3 <code>applicationDidBecomeActive(Notification as NSNotificationMBS)</code>	282
* 8.2.4 <code>applicationDidChangeScreenParameters(Notification as NSNotificationMBS)</code>	282
* 8.2.5 <code>applicationDidDecodeRestorableState(coder as NSCoderMBS)</code>	282
* 8.2.6 <code>applicationDidFailToRegisterForRemoteNotificationsWithError(error as NSErrorMBS)</code>	282
* 8.2.7 <code>applicationDidFinishLaunching(Notification as NSNotificationMBS)</code>	282
* 8.2.8 <code>applicationDidHide(Notification as NSNotificationMBS)</code>	283
* 8.2.9 <code>applicationDidReceiveRemoteNotification(userInfo as Dictionary)</code>	283
* 8.2.10 <code>applicationDidRegisterForRemoteNotificationsWithDeviceToken(deviceToken as memoryblock)</code>	283
* 8.2.11 <code>applicationDidResignActive(Notification as NSNotificationMBS)</code>	283
* 8.2.12 <code>applicationDidUnhide(Notification as NSNotificationMBS)</code>	284
* 8.2.13 <code>applicationDidUpdate(Notification as NSNotificationMBS)</code>	284
* 8.2.14 <code>applicationDockMenu as NSMenuMBS</code>	284
* 8.2.15 <code>applicationOpenFile(filename as string) as boolean</code>	284
* 8.2.16 <code>applicationOpenFiles(fileNames() as string) as boolean</code>	285
* 8.2.17 <code>applicationOpenFileWithoutUI(filename as string) as boolean</code>	285
* 8.2.18 <code>applicationOpenTempFile(filename as string) as boolean</code>	285
* 8.2.19 <code>applicationOpenUntitledFile as boolean</code>	286
* 8.2.20 <code>applicationPrintFile(filename as string) as boolean</code>	286
* 8.2.21 <code>applicationPrintFiles(fileNames() as string, printSettings as dictionary, showPrintPanels as boolean) as boolean</code>	286
* 8.2.22 <code>applicationShouldHandleReopen(hasVisibleWindows as boolean) as boolean</code>	287
* 8.2.23 <code>applicationShouldOpenUntitledFile as boolean</code>	288
* 8.2.24 <code>applicationShouldTerminate as Integer</code>	288
* 8.2.25 <code>applicationShouldTerminateAfterLastWindowClosed as boolean</code>	288
* 8.2.26 <code>applicationWillBecomeActive(Notification as NSNotificationMBS)</code>	289
* 8.2.27 <code>applicationWillEncodeRestorableState(coder as NSCoderMBS)</code>	289
* 8.2.28 <code>applicationWillFinishLaunching(Notification as NSNotificationMBS)</code>	289
* 8.2.29 <code>applicationWillHide(Notification as NSNotificationMBS)</code>	289
* 8.2.30 <code>applicationWillPresentError(error as NSErrorMBS) as NSErrorMBS</code>	290
* 8.2.31 <code>applicationWillResignActive(Notification as NSNotificationMBS)</code>	290
* 8.2.32 <code>applicationWillTerminate(Notification as NSNotificationMBS)</code>	290
* 8.2.33 <code>applicationWillUnhide(Notification as NSNotificationMBS)</code>	290
* 8.2.34 <code>applicationWillUpdate(Notification as NSNotificationMBS)</code>	291
* 8.2.35 <code>restoreWindowWithIdentifier(identifier as string, state as NSCoderMBS, byref resultWindow as Variant, byref error as NSErrorMBS) as boolean</code>	291
* 8.2.36 <code>userDidAcceptCloudKitShareWithMetadata(metadata as Variant)</code>	292
– 8.3.1 class <code>NSApplicationMBS</code>	294

* 8.3.3 activateIgnoringOtherApps(flag as boolean)	295
* 8.3.4 addWindowsItem(win as NSWindowMBS, title as string, isFilename as boolean)	295
* 8.3.5 arrangeInFront	296
* 8.3.6 cancelUserAttentionRequest(request as Integer)	296
* 8.3.7 changeWindowsItem(win as NSWindowMBS, title as string, isFilename as boolean)	296
* 8.3.8 completeStateRestoration	297
* 8.3.9 Constructor	297
* 8.3.10 deactivate	298
* 8.3.11 disableRelaunchOnLogin	298
* 8.3.12 discardEventsMatchingMask(mask as Integer, beforeEvent as NSEventMBS)	298
* 8.3.13 enabledRemoteNotificationTypes as Integer	300
* 8.3.14 enableRelaunchOnLogin	300
* 8.3.15 extendStateRestoration	300
* 8.3.16 hide	301
* 8.3.17 hideOtherApplications	301
* 8.3.18 invalidateRestorableState	301
* 8.3.19 miniaturizeAll	302
* 8.3.20 modalWindow as NSWindowMBS	302
* 8.3.21 nextEventMatchingMask(mask as Integer, untilDate as date, mode as String, dequeueFlag as boolean) as NSEventMBS	302
* 8.3.22 nextEventMatchingMask(mask as Integer, untilDate as dateTime, mode as String, dequeueFlag as boolean) as NSEventMBS	304
* 8.3.23 NSAppKitVersionNumber as Double	304
* 8.3.24 NSApplicationDidBecomeActiveNotification as string	305
* 8.3.25 NSApplicationDidChangeScreenParametersNotification as string	305
* 8.3.26 NSApplicationDidFinishLaunchingNotification as string	305
* 8.3.27 NSApplicationDidFinishRestoringWindowsNotification as string	305
* 8.3.28 NSApplicationDidHideNotification as string	306
* 8.3.29 NSApplicationDidResignActiveNotification as string	306
* 8.3.30 NSApplicationDidUnhideNotification as string	306
* 8.3.31 NSApplicationDidUpdateNotification as string	306
* 8.3.32 NSApplicationLaunchIsDefaultLaunchKey as string	307
* 8.3.33 NSApplicationLaunchRemoteNotificationKey as string	307
* 8.3.34 NSApplicationLaunchUserNotificationKey as string	307
* 8.3.35 NSApplicationWillBecomeActiveNotification as string	308
* 8.3.36 NSApplicationWillFinishLaunchingNotification as string	308
* 8.3.37 NSApplicationWillHideNotification as string	308
* 8.3.38 NSApplicationWillResignActiveNotification as string	308
* 8.3.39 NSApplicationWillTerminateNotification as string	308
* 8.3.40 NSApplicationWillUnhideNotification as string	309
* 8.3.41 NSApplicationWillUpdateNotification as string	309
* 8.3.42 orderFrontCharacterPalette	309

* 8.3.43	orderFrontStandardAboutPanel	309
* 8.3.44	orderFrontStandardAboutPanelWithOptions(options as dictionary)	310
* 8.3.45	OverlayApplicationIconImage(image as NSImageMBS)	310
* 8.3.46	postEvent(e as NSEventMBS, atStart as boolean)	312
* 8.3.47	preventWindowOrdering	312
* 8.3.48	registerForRemoteNotificationTypes(type as Integer)	312
* 8.3.49	removeWindowsItem(win as NSWindowMBS)	313
* 8.3.50	replyToApplicationShouldTerminate(reply as boolean)	313
* 8.3.51	replyToOpenOrPrint(reply as Integer)	313
* 8.3.52	requestUserAttention(type as Integer) as Integer	314
* 8.3.53	runModalForWindow(win as NSWindowMBS) as Integer	314
* 8.3.54	runPageLayout	315
* 8.3.55	sendEvent(theEvent as NSEventMBS)	315
* 8.3.56	sharedApplication as NSApplicationMBS	315
* 8.3.57	showHelp	316
* 8.3.58	startDictation	316
* 8.3.59	stopDictation	316
* 8.3.60	terminate	316
* 8.3.61	unhide	317
* 8.3.62	unhideAllApplications	317
* 8.3.63	unhideWithoutActivation	317
* 8.3.64	unregisterForRemoteNotifications	318
* 8.3.65	updateWindows	318
* 8.3.66	updateWindowsItem(win as NSWindowMBS)	318
* 8.3.67	windows as NSWindowMBS()	319
* 8.3.68	windowWithWindowNumber(windowNumber as Integer) as NSWindowMBS	319
* 8.3.70	activationPolicy as Integer	320
* 8.3.71	applicationIconImage as NSImageMBS	320
* 8.3.72	currentEvent as NSEventMBS	321
* 8.3.73	currentSystemPresentationOptions as Integer	321
* 8.3.74	dockTile as NSDockTileMBS	321
* 8.3.75	Handle as Integer	322
* 8.3.76	helpMenu as NSMenuMBS	322
* 8.3.77	isActive as Boolean	323
* 8.3.78	isFullKeyboardAccessEnabled as Boolean	323
* 8.3.79	isHidden as Boolean	323
* 8.3.80	isRunning as Boolean	324
* 8.3.81	keyWindow as NSWindowMBS	324
* 8.3.82	mainMenu as NSMenuMBS	324
* 8.3.83	mainWindow as NSWindowMBS	325
* 8.3.84	presentationOptions as Integer	325
* 8.3.85	servicesProvider as NSServiceProviderMBS	326
* 8.3.86	userInterfaceLayoutDirection as Integer	326
* 8.3.87	windowsMenu as NSMenuMBS	327

	39
• 9 Cocoa Controls	453
– 9.17.1 class NSButtonCellMBS	543
* 9.17.3 Constructor(image as NSImageMBS)	543
* 9.17.4 Constructor(text as string)	544
* 9.17.6 alternateImage as NSImageMBS	544
* 9.17.7 alternateTitle as String	544
* 9.17.8 attributedAlternateTitle as NSAttributedStringMBS	545
* 9.17.9 attributedTitle as NSAttributedStringMBS	545
* 9.17.10 backgroundColor as NSColorMBS	545
* 9.17.11 imageDimsWhenDisabled as Boolean	545
* 9.17.12 imagePosition as Integer	546
* 9.17.13 imageScaling as Integer	546
* 9.17.14 showsBorderOnlyWhileMouseInside as Boolean	546
* 9.17.15 sound as Variant	546
– 9.18.1 class NSCellMBS	547
* 9.18.3 acceptsFirstResponder as boolean	547
* 9.18.4 calcDrawInfo(theRect as NSRectMBS)	547
* 9.18.5 cellSize as NSSizeMBS	548
* 9.18.6 cellSizeForBounds(theRect as NSRectMBS) as NSSizeMBS	548
* 9.18.7 compare(otherCell as NSCellMBS) as Integer	548
* 9.18.8 Constructor(image as NSImageMBS)	548
* 9.18.9 Constructor(text as string)	549
* 9.18.10 defaultFocusRingType as Integer	549
* 9.18.11 defaultMenu as NSMenuMBS	550
* 9.18.12 drawingRectForBounds(theRect as NSRectMBS) as NSRectMBS	550
* 9.18.13 highlightColorWithFrame(theRect as NSRectMBS, controlView as NSViewMBS) as NSColorMBS	550
* 9.18.14 imageRectForBounds(theRect as NSRectMBS) as NSRectMBS	550
* 9.18.15 isEntryAcceptable(aString as string) as boolean	551
* 9.18.16 mnemonic as string	551
* 9.18.17 nextState as Integer	551
* 9.18.18 performClick	551
* 9.18.19 prefersTrackingUntilMouseUp as boolean	551
* 9.18.20 sendActionOn(mask as Integer) as Integer	552
* 9.18.21 setNextState	552
* 9.18.22 setTitleWithMnemonic(stringWithAmpersand as string)	552
* 9.18.23 titleRectForBounds(theRect as NSRectMBS) as NSRectMBS	553
* 9.18.24 wantsNotificationForMarkedText as boolean	553
* 9.18.26 alignment as Integer	553
* 9.18.27 allowsEditingTextAttributes as boolean	553
* 9.18.28 allowsMixedState as boolean	553

* 9.18.29 allowsUndo as boolean	554
* 9.18.30 attributedStringValue as NSAttributedStringMBS	554
* 9.18.31 backgroundStyle as Integer	554
* 9.18.32 baseWritingDirection as Integer	555
* 9.18.33 Bezeled as boolean	555
* 9.18.34 Bordered as boolean	555
* 9.18.35 className as string	555
* 9.18.36 classPath as string	555
* 9.18.37 Continuous as boolean	556
* 9.18.38 controlSize as Integer	556
* 9.18.39 controlTint as Integer	556
* 9.18.40 controlView as NSViewMBS	556
* 9.18.41 doubleValue as Double	557
* 9.18.42 Editable as boolean	557
* 9.18.43 Enabled as boolean	557
* 9.18.44 floatValue as Double	557
* 9.18.45 font as NSFontMBS	557
* 9.18.46 Handle as Integer	558
* 9.18.47 hasValidObjectValue as boolean	558
* 9.18.48 Highlighted as boolean	558
* 9.18.49 Identifier as String	558
* 9.18.50 image as NSImageMBS	559
* 9.18.51 importsGraphics as boolean	559
* 9.18.52 interiorBackgroundStyle as Integer	559
* 9.18.53 intValue as Integer	559
* 9.18.54 isOpaque as boolean	560
* 9.18.55 keyEquivalent as string	560
* 9.18.56 lineBreakMode as Integer	560
* 9.18.57 menu as NSMenuItemMBS	560
* 9.18.58 mnemonicLocation as Integer	561
* 9.18.59 refusesFirstResponder as boolean	561
* 9.18.60 Scrollable as boolean	561
* 9.18.61 Selectable as boolean	561
* 9.18.62 sendsActionOnEndEditing as boolean	561
* 9.18.63 showsFirstResponder as boolean	562
* 9.18.64 state as Integer	562
* 9.18.65 stringValue as string	562
* 9.18.66 tag as Integer	563
* 9.18.67 title as string	563
* 9.18.68 truncatesLastVisibleLine as boolean	563
* 9.18.69 type as Integer	563
* 9.18.70 userInterfaceLayoutDirection as Integer	564

	41
* 9.18.71 usesSingleLineMode as boolean	564
* 9.18.72 wraps as boolean	564
* 9.18.73 cellAttribute(aParameter as Integer) as Integer	565
* 9.18.74 focusRingType as Integer	565

• 10 Cocoa Drawing	735
– 10.1.1 class NSColorPanelMBS	735
* 10.1.3 attachColorList(list as NSColorListMBS)	736
* 10.1.4 Constructor	736
* 10.1.5 detachColorList(list as NSColorListMBS)	736
* 10.1.6 GetColor(byref red as single, byref green as single, byref blue as single, byref alpha as single) as boolean	736
* 10.1.7 GetColorFromDrag as color	736
* 10.1.8 GetColorFromDrag(byref red as single, byref green as single, byref blue as single, byref alpha as single) as boolean	737
* 10.1.9 orderFrontColorPanel	737
* 10.1.10 SetColor(red as single, green as single, blue as single, alpha as single)	737
* 10.1.11 setColor(value as NSColorMBS)	737
* 10.1.12 setContinuous(value as boolean)	738
* 10.1.13 setMode(value as Integer)	738
* 10.1.14 SetPickerMode(value as Integer)	738
* 10.1.15 setShowsAlpha(value as boolean)	739
* 10.1.16 SharedColorPanelExists as boolean	739
* 10.1.18 accessoryView as NSViewMBS	739
* 10.1.19 alpha as Double	739
* 10.1.20 ColorValue as Color	739
* 10.1.21 getColor as NSColorMBS	740
* 10.1.22 getColorAsRGB as NSColorMBS	740
* 10.1.23 isContinuous as boolean	740
* 10.1.24 mode as Integer	740
* 10.1.25 showsAlpha as boolean	741
* 10.1.27 Changed	741
* 10.1.28 DidMove	741
* 10.1.29 GotFocus	741
* 10.1.30 Hidden	742
* 10.1.31 LostFocus	742
* 10.1.32 Shown	742
* 10.1.33 WillClose	742

	43
• 9 Cocoa Controls	453
– 9.19.1 class NSControlMBS	568
* 9.19.3 calcSize	568
* 9.19.4 ConnectActionEvent	569
* 9.19.5 Constructor	569
* 9.19.6 Constructor(Handle as Integer)	569
* 9.19.7 Constructor(left as Double, top as Double, width as Double, height as Double)	570
* 9.19.8 Destructor	570
* 9.19.9 EnableEvents	570
* 9.19.10 performClick	570
* 9.19.11 selectCell(Cell as NSCellMBS)	570
* 9.19.12 setNeedsDisplay	571
* 9.19.13 sizeToFit	571
* 9.19.14 validateEditing	571
* 9.19.16 ActionSelector as String	571
* 9.19.17 alignment as Integer	572
* 9.19.18 attributedStringValue as NSAttributedStringMBS	572
* 9.19.19 baseWritingDirection as Integer	573
* 9.19.20 cell as Variant	573
* 9.19.21 controlSize as Integer	573
* 9.19.22 currentEditor as NSTextMBS	574
* 9.19.23 doubleValue as Double	574
* 9.19.24 font as NSFontMBS	574
* 9.19.25 ignoresMultiClick as boolean	575
* 9.19.26 integerValue as Integer	575
* 9.19.27 intValue as Integer	575
* 9.19.28 isContinuous as boolean	576
* 9.19.29 isEnabled as boolean	576
* 9.19.30 refusesFirstResponder as boolean	576
* 9.19.31 selectedCell as NSCellMBS	576
* 9.19.32 selectedTag as Integer	577
* 9.19.33 stringValue as string	577
* 9.19.34 tag as Integer	577
* 9.19.36 Action	577
* 9.19.37 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	577
* 9.19.38 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	578
* 9.19.39 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	578
* 9.19.40 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean	579
* 9.19.41 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean	579

- **13 Cocoa Regular Expressions** 831
 - 13.1.1 class NSDataDetectorMBS 831
 - * 13.1.3 Constructor(checkingTypes as Integer, byref error as NSErrorMBS) 832
 - * 13.1.4 Constructor(script as string, options as Integer, byref error as NSErrorMBS) 833
 - * 13.1.5 copy as NSDataDetectorMBS 833
 - * 13.1.6 dataDetectorWithTypes(checkingTypes as Integer, byref error as NSErrorMBS) as NSDataDetectorMBS 833
 - * 13.1.8 checkingTypes as Integer 834

	45
• 8 Cocoa	271
– 8.4.1 class NSDockTileMBS	329
* 8.4.3 Constructor	330
* 8.4.4 display	330
* 8.4.5 owner as Variant	330
* 8.4.6 size as NSSizeMBS	331
* 8.4.8 Handle as Integer	331
* 8.4.9 badgeLabel as string	331
* 8.4.10 contentView as NSViewMBS	332
* 8.4.11 showsApplicationBadge as boolean	332

• 17 Drag & Drop	947
– 17.2.1 class NSDraggingImageComponentMBS	948
* 17.2.3 Constructor(key as string)	948
* 17.2.4 draggingImageComponentWithKey(key as string) as NSDraggingImageComponentMBS	948
* 17.2.5 NSDraggingImageComponentIconKey as string	948
* 17.2.6 NSDraggingImageComponentLabelKey as string	949
* 17.2.8 Handle as Integer	949
* 17.2.9 contents as Variant	949
* 17.2.10 frame as NSRectMBS	949
* 17.2.11 key as string	950
– 17.3.1 class NSDraggingInfoMBS	951
* 17.3.3 Constructor	951
* 17.3.4 Constructor(Handle as Integer)	951
* 17.3.5 namesOfPromisedFilesDroppedAtDestination(dropDestination as FolderItem) as string()	951
* 17.3.6 promisedFilesDroppedAtDestination(dropDestination as FolderItem) as FolderItem()	952
* 17.3.7 slideDraggedImageTo(screenPoint as NSPointMBS)	952
* 17.3.9 animatesToDestination as boolean	953
* 17.3.10 draggedImage as Variant	953
* 17.3.11 draggedImageLocation as NSPointMBS	953
* 17.3.12 draggingDestinationWindow as Variant	954
* 17.3.13 draggingFormation as Integer	954
* 17.3.14 draggingLocation as NSPointMBS	954
* 17.3.15 draggingPasteboard as Variant	954
* 17.3.16 draggingSequenceNumber as Integer	955
* 17.3.17 draggingSource as Variant	955
* 17.3.18 draggingSourceOperationMask as Integer	955
* 17.3.19 Handle as Integer	956
* 17.3.20 numberOfValidItemsForDrop as Integer	956
– 17.4.1 class NSDraggingItemMBS	958
* 17.4.3 Constructor(item as NSPasteboardItemMBS)	958
* 17.4.4 item as Variant	958
* 17.4.5 setDraggingFrame(frame as NSRectMBS, contents as Variant)	959
* 17.4.7 Handle as Integer	959
* 17.4.8 draggingFrame as NSRectMBS	959
– 17.5.1 class NSDraggingSessionMBS	961
* 17.5.3 Constructor	961
* 17.5.4 draggingLeaderIndex as Integer	961
* 17.5.5 draggingLocation as NSPointMBS	961

	47
* 17.5.6 draggingPasteboard as NSPasteboardMBS	962
* 17.5.7 draggingSequenceNumber as Integer	962
* 17.5.9 Handle as Integer	962
* 17.5.10 animatesToStartingPositionsOnCancelOrFail as boolean	962
* 17.5.11 draggingFormation as Integer	963

• 14 Cocoa Tasks	859
– 14.1.1 class NSFileHandleMBS	859
* 14.1.3 acceptConnectionInBackgroundAndNotify	860
* 14.1.4 AvailableBytes as Integer	860
* 14.1.5 availableData as MemoryBlock	860
* 14.1.6 closeFile	860
* 14.1.7 Constructor	861
* 14.1.8 fileDescriptor as Integer	861
* 14.1.9 fileHandleForReadingAtPath(path as folderitem) as NSFileHandleMBS	862
* 14.1.10 fileHandleForReadingAtPath(path as string) as NSFileHandleMBS	862
* 14.1.11 fileHandleForReadingFromFile(URL as folderitem, byref error as NSErrorMBS) as NSFileHandleMBS	862
* 14.1.12 fileHandleForReadingFromURL(URL as string, byref error as NSErrorMBS) as NSFileHandleMBS	863
* 14.1.13 fileHandleForUpdatingAtPath(path as folderitem) as NSFileHandleMBS	864
* 14.1.14 fileHandleForUpdatingAtPath(path as string) as NSFileHandleMBS	864
* 14.1.15 fileHandleForUpdatingFile(URL as folderitem, byref error as NSErrorMBS) as NSFileHandleMBS	864
* 14.1.16 fileHandleForUpdatingURL(URL as string, byref error as NSErrorMBS) as NSFileHandleMBS	865
* 14.1.17 fileHandleForWritingAtPath(path as folderitem) as NSFileHandleMBS	865
* 14.1.18 fileHandleForWritingAtPath(path as string) as NSFileHandleMBS	866
* 14.1.19 fileHandleForWritingToFile(URL as folderitem, byref error as NSErrorMBS) as NSFileHandleMBS	866
* 14.1.20 fileHandleForWritingToURL(URL as string, byref error as NSErrorMBS) as NSFileHandleMBS	867
* 14.1.21 fileHandleWithFileDescriptor(fd as Integer) as NSFileHandleMBS	867
* 14.1.22 fileHandleWithFileDescriptor(fd as Integer, closeOnDealloc as boolean) as NSFileHandleMBS	867
* 14.1.23 fileHandleWithNullDevice as NSFileHandleMBS	867
* 14.1.24 fileHandleWithStandardError as NSFileHandleMBS	868
* 14.1.25 fileHandleWithStandardInput as NSFileHandleMBS	868
* 14.1.26 fileHandleWithStandardOutput as NSFileHandleMBS	868
* 14.1.27 NSFileHandleConnectionAcceptedNotification as string	869
* 14.1.28 NSFileHandleDataAvailableNotification as string	869
* 14.1.29 NSFileHandleNotificationDataItem as string	869
* 14.1.30 NSFileHandleNotificationFileHandleItem as string	870
* 14.1.31 NSFileHandleNotificationMonitorModes as string	870
* 14.1.32 NSFileHandleOperationException as string	870
* 14.1.33 NSFileHandleReadCompletionNotification as string	870
* 14.1.34 NSFileHandleReadToEndOfFileCompletionNotification as string	870
* 14.1.35 readDataOfLength(length as Integer) as MemoryBlock	871

* 14.1.36 readDataToEndOfFile as MemoryBlock	872
* 14.1.37 readInBackgroundAndNotify	872
* 14.1.38 readToEndOfFileInBackgroundAndNotify	872
* 14.1.39 seekToEndOfFile as UInt64	873
* 14.1.40 seekToFileOffset(offset as UInt64)	873
* 14.1.41 synchronizeFile	873
* 14.1.42 truncateFileAtOffset(offset as UInt64)	874
* 14.1.43 waitForDataInBackgroundAndNotify	874
* 14.1.44 writeData(data as MemoryBlock)	874
* 14.1.46 Handle as Integer	875
* 14.1.47 offsetInFile as UInt64	875

- 18 iCloud 965
 - 18.1.1 class NSFileManagerMBS 965
 - * 18.1.3 attributesOfFileSystemForPath(item as folderitem, byref error as NSErrorMBS) as Dictionary 966
 - * 18.1.4 attributesOfFileSystemForPath(path as string, byref error as NSErrorMBS) as Dictionary 966
 - * 18.1.5 attributesOfItemAtPath(item as folderitem, byref error as NSErrorMBS) as Dictionary 967
 - * 18.1.6 attributesOfItemAtPath(path as string, byref error as NSErrorMBS) as Dictionary 968
 - * 18.1.7 changeCurrentDirectory(folder as folderitem) as boolean 968
 - * 18.1.8 changeCurrentDirectory(path as string) as boolean 969
 - * 18.1.9 Constructor 969
 - * 18.1.10 containerFolderForSecurityApplicationGroupIdentifier(groupIdentifier as string) as folderItem 969
 - * 18.1.11 containerURLForSecurityApplicationGroupIdentifier(groupIdentifier as string) as string 970
 - * 18.1.12 contentsEqual(path1 as folderitem, path2 as folderitem) as boolean 970
 - * 18.1.13 copyItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean 971
 - * 18.1.14 copyItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean 971
 - * 18.1.15 copyItemMT(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean 972
 - * 18.1.16 copyItemMT(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean 973
 - * 18.1.17 createDirectory(Path as folderItem, createIntermediates as boolean = true, attrs as Dictionary = nil, byref error as NSErrorMBS) as boolean 974
 - * 18.1.18 createDirectory(Path as String, createIntermediates as boolean = true, attrs as Dictionary = nil, byref error as NSErrorMBS) as boolean 974
 - * 18.1.19 createFile(Path as folderItem, contents as MemoryBlock, attrs as Dictionary = nil) as boolean 975
 - * 18.1.20 createFile(Path as String, contents as MemoryBlock, attrs as Dictionary = nil) as boolean 976
 - * 18.1.21 createSymbolicLink(file as folderitem, destFile as folderitem, byref error as NSErrorMBS) as boolean 977
 - * 18.1.22 createSymbolicLink(path as string, destPath as string, byref error as NSErrorMBS) as boolean 977
 - * 18.1.23 destinationOfSymbolicLinkAtPath(file as folderitem, byref error as NSErrorMBS) as string 978
 - * 18.1.24 destinationOfSymbolicLinkAtPath(path as string, byref error as NSErrorMBS) as string 978
 - * 18.1.25 displayName(path as folderitem) as string 979
 - * 18.1.26 evictUbiquitousItem(item as folderitem, byref error as NSErrorMBS) as boolean 979

* 18.1.27 fileExists(path as folderitem) as boolean	980
* 18.1.28 fileExists(path as folderitem, byref isDirectory as boolean) as boolean	981
* 18.1.29 FileForUbiquityContainerIdentifier(containerIdentifier as string) as folderitem	982
* 18.1.30 fileManagerWithAuthorization(authorization as NSWorkspaceAuthorizationMBS) as NSFileManagerMBS	982
* 18.1.31 homeDirectoryForUser(Name as string) as FolderItem	984
* 18.1.32 isDeletableFile(path as folderitem) as boolean	984
* 18.1.33 isExecutableFile(path as folderitem) as boolean	985
* 18.1.34 isReadableFile(path as folderitem) as boolean	985
* 18.1.35 isUbiquitousItem(item as folderitem) as boolean	986
* 18.1.36 isUbiquitousItem(URL as string) as boolean	987
* 18.1.37 isWritableFile(path as folderitem) as boolean	987
* 18.1.38 lastPathComponent(pathOrURL as string) as string	988
* 18.1.39 linkItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean	988
* 18.1.40 linkItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean	989
* 18.1.41 moveItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean	989
* 18.1.42 moveItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean	990
* 18.1.43 NSFileAppendOnly as string	991
* 18.1.44 NSFileBusy as string	991
* 18.1.45 NSFileCreationDate as string	991
* 18.1.46 NSFileDeviceIdentifier as string	991
* 18.1.47 NSFileExtensionHidden as string	992
* 18.1.48 NSFileGroupOwnerAccountID as string	992
* 18.1.49 NSFileGroupOwnerAccountName as string	992
* 18.1.50 NSFileHFSCreatorCode as string	993
* 18.1.51 NSFileHFSTypeCode as string	993
* 18.1.52 NSFileImmutable as string	994
* 18.1.53 NSFileManagerUnmountDissentingProcessIdentifierErrorKey as string	994
* 18.1.54 NSFileModificationDate as string	994
* 18.1.55 NSFileOwnerAccountID as string	994
* 18.1.56 NSFileOwnerAccountName as string	995
* 18.1.57 NSFilePosixPermissions as string	996
* 18.1.58 NSFileReferenceCount as string	996
* 18.1.59 NSFileSize as string	996
* 18.1.60 NSFileSystemFileNumber as string	997
* 18.1.61 NSFileSystemFreeNodes as string	997
* 18.1.62 NSFileSystemFreeSize as string	997
* 18.1.63 NSFileSystemNodes as string	997

* 18.1.64 NSFileSystemNumber as string	998
* 18.1.65 NSFileSystemSize as string	998
* 18.1.66 NSFileType as string	998
* 18.1.67 NSFileTypeBlockSpecial as string	998
* 18.1.68 NSFileTypeCharacterSpecial as string	999
* 18.1.69 NSFileTypeDirectory as string	999
* 18.1.70 NSFileTypeRegular as string	999
* 18.1.71 NSFileTypeSocket as string	1000
* 18.1.72 NSFileTypeSymbolicLink as string	1000
* 18.1.73 NSFileTypeUnknown as string	1000
* 18.1.74 NSUbiquityIdentityDidChangeNotification as string	1000
* 18.1.75 pathExtension(pathOrURL as string) as string	1000
* 18.1.76 removeItem(file as folderitem, byref error as NSErrorMBS) as boolean	1001
* 18.1.77 removeItem(path as string, byref error as NSErrorMBS) as boolean	1001
* 18.1.78 replaceItem(originalItem as FolderItem, newItem as FolderItem, backupItemName as String, options as integer, byref resultingItem as FolderItem, byref error as NSErrorMBS) as Boolean	1001
* 18.1.79 replaceItemAtURL(originalItemURL as string, newItemURL as String, backupItemName as String, options as integer, byref resultingURL as String, byref error as NSErrorMBS) as Boolean	1003
* 18.1.80 setAttributes(attributesDic as dictionary, item as folderitem, byref error as NSErrorMBS) as boolean	1004
* 18.1.81 setAttributes(attributesDic as dictionary, path as string, byref error as NSErrorMBS) as boolean	1005
* 18.1.82 setUbiquitous(flag as boolean, item as folderitem, destitem as folderitem, byref error as NSErrorMBS) as boolean	1005
* 18.1.83 setUbiquitous(flag as boolean, item as folderitem, destURL as string, byref error as NSErrorMBS) as boolean	1006
* 18.1.84 startDownloadingUbiquitousItem(item as folderitem, byref error as NSErrorMBS) as boolean	1007
* 18.1.85 startDownloadingUbiquitousItem(URL as string, byref error as NSErrorMBS) as boolean	1008
* 18.1.86 stringByAbbreviatingWithTildeInPath(path as string) as string	1008
* 18.1.87 stringByAppendingPathComponent(path as string, Component as string) as string	1009
* 18.1.88 stringByAppendingPathExtension(path as string, Extension as string) as string	1009
* 18.1.89 stringByDeletingLastPathComponent(path as string) as string	1010
* 18.1.90 stringByDeletingPathExtension(path as string) as string	1011
* 18.1.91 stringByExpandingTildeInPath(path as string) as string	1011
* 18.1.92 stringByResolvingSymlinksInPath(path as string) as string	1011
* 18.1.93 stringByStandardizingPath(path as string) as string	1012
* 18.1.94 trashItem(file as folderItem, byref Resulting as folderItem, byref error as NSErrorMBS) as boolean	1012

- * 18.1.95 trashItem(Path as String, byref Resulting as FolderItem, byref error as NSErrorMBS) as boolean 1013
- * 18.1.96 URLByAppendingPathComponent(URL as string, pathComponent as string) as string 1013
- * 18.1.97 URLByAppendingPathComponent(URL as string, pathComponent as string, isDirectory as boolean) as string 1014
- * 18.1.98 URLByAppendingPathExtension(URL as string, pathExtension as string) as string 1014
- * 18.1.99 URLByDeletingLastPathComponent(URL as string) as string 1015
- * 18.1.100 URLByDeletingPathExtension(URL as string) as string 1015
- * 18.1.101 URLByResolvingSymlinksInPath(URL as string) as string 1015
- * 18.1.102 URLByStandardizingPath(URL as string) as string 1015
- * 18.1.103 URLForPublishingUbiquitousItem(item as folderitem, byref expirationDate as date, byref error as NSErrorMBS) as string 1017
- * 18.1.104 URLForPublishingUbiquitousItem(URL as string, byref expirationDate as date, byref error as NSErrorMBS) as string 1018
- * 18.1.105 URLForUbiquityContainerIdentifier(containerIdentifier as string) as string 1018
- * 18.1.107 currentDirectory as folderitem 1019
- * 18.1.108 currentDirectoryPath as string 1020
- * 18.1.109 defaultManager as NSFileManagerMBS 1020
- * 18.1.110 Handle as Integer 1021
- * 18.1.111 homeDirectoryForCurrentUser as FolderItem 1021
- * 18.1.112 temporaryDirectory as FolderItem 1021

• 8 Cocoa	271
– 8.5.1 class NSFontManagerMBS	334
* 8.5.3 addCollection(collectionName as String, Options as Integer = 0) as Boolean	334
* 8.5.4 addFontDescriptorsToCollection(descriptors() as NSFontDescriptorMBS, collectionName as String)	335
* 8.5.5 availableFontFamilies as string()	335
* 8.5.6 availableFontNamesMatchingFontDescriptor(descriptor as NSFontDescriptorMBS) as String()	336
* 8.5.7 availableFontNamesWithTraits(traits as Integer) as string()	336
* 8.5.8 availableFonts as string()	337
* 8.5.9 availableMembersOfFontFamily(FontFamily as string) as Variant()	337
* 8.5.10 collectionNames as string()	338
* 8.5.11 Constructor	339
* 8.5.12 convertAttributes(dic as dictionary) as dictionary	339
* 8.5.13 convertFont(font as NSFontMBS) as NSFontMBS	339
* 8.5.14 convertFontToFace(font as NSFontMBS, face as string) as NSFontMBS	339
* 8.5.15 convertFontToFamily(font as NSFontMBS, family as string) as NSFontMBS	340
* 8.5.16 convertFontToHaveTrait(font as NSFontMBS, trait as Integer) as NSFontMBS	340
* 8.5.17 convertFontToNotHaveTrait(font as NSFontMBS, trait as Integer) as NSFontMBS	341
* 8.5.18 convertFontToSize(font as NSFontMBS, size as Double) as NSFontMBS	341
* 8.5.19 convertFontTraits(traits as Integer) as Integer	342
* 8.5.20 convertWeightOfFont(font as NSFontMBS, up as boolean) as NSFontMBS	342
* 8.5.21 fontDescriptorsInCollection(collectionName as String) as NSFontDescriptorMBS()	343
* 8.5.22 fontHasTraits(fontName as string, Traits as Integer) as boolean	343
* 8.5.23 isMultiple as boolean	344
* 8.5.24 orderFrontFontPanel	344
* 8.5.25 orderFrontStylesPanel	344
* 8.5.26 removeCollection(collectionName as String) as Boolean	344
* 8.5.27 removeFontDescriptorFromCollection(descriptor as NSFontDescriptorMBS, collectionName as String)	344
* 8.5.28 selectedFont as NSFontMBS	345
* 8.5.29 setSelectedAttributes(dic as dictionary, isMultiple as boolean)	345
* 8.5.30 setSelectedFont(font as NSFontMBS, isMultiple as boolean)	345
* 8.5.31 sharedFontManager as NSFontManagerMBS	345
* 8.5.32 traitsOfFont(font as NSFontMBS) as Integer	346
* 8.5.33 weightOfFont(font as NSFontMBS) as Integer	346
* 8.5.35 Handle as Integer	346
* 8.5.36 Enabled as boolean	347
– 8.6.1 class NSFontPanelMBS	349
* 8.6.3 Constructor	349
* 8.6.4 convertAttributes(old as dictionary) as dictionary	349

	55
* 8.6.5 convertFont(oldFont as NSFontMBS) as NSFontMBS	349
* 8.6.6 Destructor	350
* 8.6.7 panelConvertFont(font as NSFontMBS) as NSFontMBS	350
* 8.6.8 reloadDefaultFontFamilies	350
* 8.6.9 setPanelFont(font as NSFontMBS, isMultiple as boolean)	350
* 8.6.10 sharedFontPanel as NSFontPanelMBS	351
* 8.6.11 sharedFontPanelExists as boolean	351
* 8.6.12 worksWhenModal as boolean	351
* 8.6.14 accessoryView as NSViewMBS	351
* 8.6.15 Enabled as boolean	351
* 8.6.17 changeAttributes	352
* 8.6.18 changeFont	352
* 8.6.19 validModesForFontPanel as Integer	352

• 10 Cocoa Drawing	735
– 10.2.1 class NSGraphicsMBS	744
* 10.2.3 addClip(path as NSBezierPathMBS)	745
* 10.2.4 boundingRectWithSize(text as NSAttributedStringMBS, size as NSSizeMBS, options as Integer = 0) as NSRectMBS	745
* 10.2.5 boundingRectWithSize(text as string, size as NSSizeMBS, options as Integer = 0, DicAttributes as dictionary = nil) as NSRectMBS	746
* 10.2.6 clipRect(r as NSRectMBS)	747
* 10.2.7 concat(transform as NSAffineTransformMBS)	747
* 10.2.8 ConcatTransform(NSAffineTransform as Variant)	747
* 10.2.9 Constructor	747
* 10.2.10 Constructor(targetImage as NSBitmapImageRepMBS)	748
* 10.2.11 Constructor(targetImage as NSImageMBS)	749
* 10.2.12 Constructor(targetView as NSViewMBS)	749
* 10.2.13 Constructor(targetWindow as DesktopWindow)	750
* 10.2.14 Constructor(targetWindow as NSWindowMBS)	750
* 10.2.15 Constructor(targetWindow as window)	751
* 10.2.16 drawAtPoint(image as NSImageMBS, x as Double, y as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)	751
* 10.2.17 drawAtPoint(text as NSAttributedStringMBS, point as NSPointMBS)	752
* 10.2.18 drawAtPoint(text as string, point as NSPointMBS, DicAttributes as dictionary = nil)	752
* 10.2.19 drawInRect(image as NSImageMBS, x as Double, y as Double, w as Double, h as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)	753
* 10.2.20 drawInRect(text as NSAttributedStringMBS, rect as NSRectMBS)	754
* 10.2.21 drawInRect(text as string, rect as NSRectMBS, DicAttributes as dictionary = nil)	755
* 10.2.22 drawPicture(image as Picture, x as Double, y as Double, w as Double, h as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)	755
* 10.2.23 drawRect(x as Double, y as Double, w as Double, h as Double)	755
* 10.2.24 DrawWindowBackground(x as Double, y as Double, w as Double, h as Double)	756
* 10.2.25 drawWithRect(text as NSAttributedStringMBS, rect as NSRectMBS, options as Integer)	756
* 10.2.26 eraseRect(x as Double, y as Double, w as Double, h as Double)	757
* 10.2.27 fill(path as NSBezierPathMBS)	757
* 10.2.28 fillRect(r as NSRectMBS)	758
* 10.2.29 fillRect(x as Double, y as Double, w as Double, h as Double)	758
* 10.2.30 fillRect(x as Double, y as Double, w as Double, h as Double, operation as Integer)	759
* 10.2.31 flushGraphics	759
* 10.2.32 graphicsContext as NSGraphicsMBS	759

* 10.2.33 graphicsContextWithCGContext(targetCGContext as Variant, initialFlippedState as boolean = false) as NSGraphicsMBS	759
* 10.2.34 graphicsContextWithCGContextHandle(targetCGContextRef as Integer, initialFlippedState as boolean = false) as NSGraphicsMBS	760
* 10.2.35 graphicsContextWithCGContextHandle(targetCGContextRef as Ptr, initialFlippedState as boolean = false) as NSGraphicsMBS	761
* 10.2.36 graphicsContextWithNSBitmapImageRep(targetImage as NSBitmapImageRepMBS) as NSGraphicsMBS	761
* 10.2.37 graphicsContextWithNSImage(targetImage as NSImageMBS) as NSGraphicsMBS	762
* 10.2.38 graphicsContextWithNSView(targetView as NSViewMBS) as NSGraphicsMBS	762
* 10.2.39 graphicsContextWithNSWindow(targetNSWindow as NSWindowMBS) as NSGraphicsMBS	763
* 10.2.40 graphicsContextWithWindow(targetWindow as DesktopWindow) as NSGraphicsMBS	763
* 10.2.41 graphicsContextWithWindow(targetWindow as window) as NSGraphicsMBS	763
* 10.2.42 graphicsPort as Variant	764
* 10.2.43 highlightRect(x as Double, y as Double, w as Double, h as Double)	764
* 10.2.44 invalidate	764
* 10.2.45 isDrawingToScreen as boolean	764
* 10.2.46 isFlipped as boolean	765
* 10.2.47 restoreGraphicsState	765
* 10.2.48 saveGraphicsState	765
* 10.2.49 ScaleCoordinates(x as Double, y as Double)	765
* 10.2.50 set(transform as NSAffineTransformMBS)	766
* 10.2.51 setClip(path as NSBezierPathMBS)	766
* 10.2.52 setColor(c as NSColorMBS)	766
* 10.2.53 SetColorBW(white as Double, alpha as Double = 1.0)	766
* 10.2.54 SetColorCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0)	767
* 10.2.55 SetColorHSV(hue as Double, saturation as Double, brightness as Double, alpha as Double = 1.0)	767
* 10.2.56 SetColorRGB(red as Double, green as Double, blue as Double, alpha as Double = 1.0)	767
* 10.2.57 setCurrentContext	767
* 10.2.58 setFillColor(c as NSColorMBS)	767
* 10.2.59 setStrokeColor(c as NSColorMBS)	768
* 10.2.60 SetTransform(NSAffineTransform as Variant)	768
* 10.2.61 sizeWithAttributes(text as string, DicAttributes as dictionary = nil) as NSSizeMBS	768
* 10.2.62 stroke(path as NSBezierPathMBS)	769
* 10.2.63 strokeLine(point1 as NSPointMBS, point2 as NSPointMBS)	769
* 10.2.64 strokeLine(x1 as Double, y1 as Double, x2 as Double, y2 as Double)	770

* 10.2.65 strokeRect(r as NSRectMBS)	770
* 10.2.66 TranslateCoordinates(x as Double, y as Double)	771
* 10.2.68 Handle as Integer	771
* 10.2.69 Owner as Variant	771
* 10.2.70 Valid as Boolean	771
* 10.2.71 imageInterpolation as Integer	772
* 10.2.72 shouldAntialias as boolean	772

	59
• 9 Cocoa Controls	453
– 9.20.1 class NSImageCellMBS	580
* 9.20.3 Constructor(image as NSImageMBS)	580
* 9.20.4 Constructor(text as string)	581
* 9.20.6 imageAlignment as Integer	581
* 9.20.7 imageFrameStyle as Integer	581
* 9.20.8 imageScaling as Integer	581

• 8 Cocoa	271
– 8.7.1 class NSLayoutManagerMBS	354
* 8.7.3 addTextContainer(container as NSTextContainerMBS)	355
* 8.7.4 characterIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS, byref partialFraction as Double) as Integer	355
* 8.7.5 Constructor	356
* 8.7.6 glyphIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS) as Integer	356
* 8.7.7 glyphIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS, byref partialFraction as Double) as Integer	357
* 8.7.8 glyphRangeForTextContainer(container as NSTextContainerMBS) as NSRangeMBS	357
* 8.7.9 lineFragmentRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS) as NSRectMBS	358
* 8.7.10 lineFragmentRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS, withoutAdditionalLayout as boolean) as NSRectMBS	358
* 8.7.11 lineFragmentUsedRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS) as NSRectMBS	359
* 8.7.12 lineFragmentUsedRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS, withoutAdditionalLayout as boolean) as NSRectMBS	359
* 8.7.13 locationForGlyphAtIndex(glyphIndex as Integer) as NSPointMBS	360
* 8.7.14 rangeOfNominallySpacedGlyphsContainingIndex(glyphIndex as Integer) as NSRangeMBS	361
* 8.7.15 rectArrayForCharacterRange(charRange as NSRangeMBS, selCharRange as NSRangeMBS, container as NSTextContainerMBS, byref rectCount as Integer) as NSRectMBS()	361
* 8.7.16 rectArrayForGlyphRange(glyphRange as NSRangeMBS, selGlyphRange as NSRangeMBS, container as NSTextContainerMBS, byref rectCount as Integer) as NSRectMBS()	362
* 8.7.17 removeTextContainerAtIndex(index as Integer)	362
* 8.7.18 replaceGlyphAtIndex(glyphIndex as Integer, newGlyph as Integer)	363
* 8.7.19 replaceTextStorage(newTextStorage as NSTextStorageMBS)	363
* 8.7.20 setCharacterIndex(charIndex as Integer, glyphIndex as Integer)	363
* 8.7.21 setExtraLineFragmentRect(fragmentRect as NSRectMBS, usedRect as NSRectMBS, TextContainer as NSTextContainerMBS)	364
* 8.7.22 setLineFragmentRect(fragmentRect as NSRectMBS, glyphRange as NSRangeMBS, usedRect as NSRectMBS)	364
* 8.7.23 usedRectForTextContainer(container as NSTextContainerMBS) as NSRectMBS	365
* 8.7.25 allowsNonContiguousLayout as boolean	365
* 8.7.26 attributedString as NSAttributedStringMBS	365
* 8.7.27 backgroundLayoutEnabled as boolean	365
* 8.7.28 font as NSFontMBS	366
* 8.7.29 Handle as Integer	366
* 8.7.30 hasNonContiguousLayout as boolean	366
* 8.7.31 hyphenationFactor as Double	366
* 8.7.32 showInvisibleCharacters as boolean	367

* 8.7.33	showsControlCharacters as boolean	367
* 8.7.34	showsInvisibleCharacters as boolean	367
* 8.7.35	textColor as NSColorMBS	368
* 8.7.36	textStorage as NSTextStorageMBS	368
* 8.7.37	usesFontLeading as Boolean	368
* 8.7.38	usesScreenFonts as boolean	369
* 8.7.39	InvisibleCharMapping(character as Integer) as string	369

• 20 Linguistic	1039
– 20.1.1 class NSLinguisticTaggerMBS	1039
* 20.1.3 availableTagSchemesForLanguage(Language as String) as String()	1040
* 20.1.4 availableTagSchemesForUnit(unit as Integer, Language as String) as String()	1040
* 20.1.5 Constructor(tagSchemes() as String, options as integer = 0)	1040
* 20.1.6 dominantLanguageForString(text as String) as String	1041
* 20.1.7 NSLinguisticTagAdjective as String	1041
* 20.1.8 NSLinguisticTagAdverb as String	1041
* 20.1.9 NSLinguisticTagClassifier as String	1041
* 20.1.10 NSLinguisticTagCloseParenthesis as String	1042
* 20.1.11 NSLinguisticTagCloseQuote as String	1042
* 20.1.12 NSLinguisticTagConjunction as String	1042
* 20.1.13 NSLinguisticTagDash as String	1042
* 20.1.14 NSLinguisticTagDeterminer as String	1042
* 20.1.15 NSLinguisticTagIdiom as String	1043
* 20.1.16 NSLinguisticTagInterjection as String	1043
* 20.1.17 NSLinguisticTagNoun as String	1043
* 20.1.18 NSLinguisticTagNumber as String	1043
* 20.1.19 NSLinguisticTagOpenParenthesis as String	1043
* 20.1.20 NSLinguisticTagOpenQuote as String	1043
* 20.1.21 NSLinguisticTagOrganizationName as String	1044
* 20.1.22 NSLinguisticTagOther as String	1044
* 20.1.23 NSLinguisticTagOtherPunctuation as String	1044
* 20.1.24 NSLinguisticTagOtherWhitespace as String	1044
* 20.1.25 NSLinguisticTagOtherWord as String	1044
* 20.1.26 NSLinguisticTagParagraphBreak as String	1045
* 20.1.27 NSLinguisticTagParticle as String	1045
* 20.1.28 NSLinguisticTagPersonalName as String	1045
* 20.1.29 NSLinguisticTagPlaceName as String	1045
* 20.1.30 NSLinguisticTagPreposition as String	1045
* 20.1.31 NSLinguisticTagPronoun as String	1045
* 20.1.32 NSLinguisticTagPunctuation as String	1046
* 20.1.33 NSLinguisticTagSchemeLanguage as String	1046
* 20.1.34 NSLinguisticTagSchemeLemma as String	1046
* 20.1.35 NSLinguisticTagSchemeLexicalClass as String	1046
* 20.1.36 NSLinguisticTagSchemeNameType as String	1046
* 20.1.37 NSLinguisticTagSchemeNameTypeOrLexicalClass as String	1047
* 20.1.38 NSLinguisticTagSchemeScript as String	1047
* 20.1.39 NSLinguisticTagSchemeTokenType as String	1047
* 20.1.40 NSLinguisticTagSentenceTerminator as String	1047
* 20.1.41 NSLinguisticTagVerb as String	1047

* 20.1.42	NSLinguisticTagWhitespace as String	1048
* 20.1.43	NSLinguisticTagWord as String	1048
* 20.1.44	NSLinguisticTagWordJoiner as String	1048
* 20.1.45	orthographyAtIndex(charIndex as integer, byref effectiveRange as NSRangeMBS) as NSOrthographyMBS	1048
* 20.1.46	sentenceRangeForRange(range as NSRangeMBS) as NSRangeMBS	1049
* 20.1.47	setOrthography(orthography as NSOrthographyMBS, range as NSRangeMBS)	1049
* 20.1.48	tagAtIndex(charIndex as Integer, Scheme as String, byref tokenRange as NSRangeMBS, byref sentenceRange as NSRangeMBS) as String	1049
* 20.1.49	tagAtIndex(charIndex as Integer, unit as Integer, Scheme as String, byref tokenRange as NSRangeMBS) as String	1050
* 20.1.50	tagForString(text as string, charIndex as Integer, unit as Integer, Scheme as String, orthography as NSOrthographyMBS, byref tokenRange as NSRangeMBS) as String	1050
* 20.1.51	tagSchemes as String()	1051
* 20.1.52	TagsForString(text as string, range as NSRangeMBS, unit as Integer, Scheme as String, options as Integer, orthography as NSOrthographyMBS) as NSLinguisticValueMBS()	1051
* 20.1.53	tagsInRange(range as NSRangeMBS, Scheme as String, options as Integer) as NSLinguisticValueMBS()	1052
* 20.1.54	tagsInRange(range as NSRangeMBS, unit as Integer, Scheme as String, options as Integer) as NSLinguisticValueMBS()	1052
* 20.1.55	tokenRangeAtIndex(charIndex as Integer, Unit as Integer) as NSRangeMBS	1052
* 20.1.57	dominantLanguage as String	1053
* 20.1.58	Handle as Integer	1053
* 20.1.59	Text as String	1053
– 20.2.1	class NSLinguisticValueMBS	1055
* 20.2.3	Constructor	1055
* 20.2.5	sentenceRange as NSRangeMBS	1055
* 20.2.6	Tag as String	1055
* 20.2.7	Text as String	1056
* 20.2.8	tokenRange as NSRangeMBS	1056

• 9 Cocoa Controls	453
– 9.21.1 class NSMenuItemCellMBS	584
* 9.21.3 calcSize	584
* 9.21.4 Constructor(image as NSImageMBS)	584
* 9.21.5 Constructor(text as string)	584
* 9.21.7 menuItem as NSMenuItemMBS	585
* 9.21.8 needsDisplay as Boolean	585
* 9.21.9 needsSizing as Boolean	585
* 9.21.10 tag as Integer	585

	65
• 15 Cocoa Text	891
– 15.1.1 class NSMutableParagraphStyleMBS	891
* 15.1.3 addTabStop(tabstop as NSTextTabMBS)	892
* 15.1.4 Constructor	892
* 15.1.5 removeTabStop(tabstop as NSTextTabMBS)	892
* 15.1.6 setAlignment(alignment as Integer)	892
* 15.1.7 setBaseWritingDirection(writingDirection as Integer)	893
* 15.1.8 setDefaultTabInterval(value as Double)	893
* 15.1.9 setFirstLineHeadIndent(value as Double)	893
* 15.1.10 setHeaderLevel(level as Integer)	893
* 15.1.11 setHeadIndent(value as Double)	894
* 15.1.12 setHyphenationFactor(value as Double)	894
* 15.1.13 setLineBreakMode(mode as Integer)	894
* 15.1.14 setLineHeightMultiple(value as Double)	894
* 15.1.15 setLineSpacing(value as Double)	895
* 15.1.16 setMaximumLineHeight(value as Double)	896
* 15.1.17 setMinimumLineHeight(value as Double)	896
* 15.1.18 setParagraphSpacing(value as Double)	896
* 15.1.19 setParagraphSpacingBefore(value as Double)	896
* 15.1.20 setParagraphStyle(ParagraphStyle as NSParagraphStyleMBS)	896
* 15.1.21 setTabStops(tabStops() as NSTextTabMBS)	897
* 15.1.22 setTailIndent(value as Double)	897
* 15.1.23 setTextBlocks(TextBlocks() as NSTextBlockMBS)	897
* 15.1.24 setTextLists(TextLists() as NSTextListMBS)	897
* 15.1.25 setTighteningFactorForTruncation(value as Double)	897

• 21 Navigation	1061
– 21.1.1 class NSOpenPanelMBS	1061
* 21.1.3 beginForDirectory(path as folderitem, name as string, filetypes() as string)	1061
* 21.1.4 beginSheetForDirectory(path as folderitem, name as string, filetypes() as string, targetWindow as DesktopWindow)	1062
* 21.1.5 beginSheetForDirectory(path as folderitem, name as string, filetypes() as string, targetWindow as window)	1063
* 21.1.6 Constructor	1063
* 21.1.7 Files as FolderItem()	1063
* 21.1.8 Files(index as UInt32) as folderitem	1064
* 21.1.9 runModalForDirectory(path as folderitem, name as string, filetypes as string) as Integer	1064
* 21.1.10 runModalForDirectory(path as folderitem, name as string, filetypes() as string) as Integer	1064
* 21.1.11 runModalForTypes(filetypes as string) as Integer	1064
* 21.1.12 runModalForTypes(filetypes() as string) as Integer	1065
* 21.1.13 URL(index as UInt32) as string	1065
* 21.1.14 URLs as String()	1065
* 21.1.16 AccessoryViewDisclosed as Boolean	1066
* 21.1.17 allowsMultipleSelection as boolean	1066
* 21.1.18 canChooseDirectories as boolean	1066
* 21.1.19 canChooseFiles as boolean	1066
* 21.1.20 canDownloadUbiquitousContents as Boolean	1066
* 21.1.21 canResolveUbiquitousConflicts as Boolean	1067
* 21.1.22 FilesCount as UInt32	1067
* 21.1.23 resolvesAliases as boolean	1068

• 20 Linguistic	1039
– 20.3.1 class NSOrthographyMBS	1057
* 20.3.3 allLanguages as String()	1057
* 20.3.4 allScripts as String()	1057
* 20.3.5 Constructor(script as string, map as dictionary)	1058
* 20.3.6 copy as NSOrthographyMBS	1058
* 20.3.7 defaultOrthographyForLanguage(language as string) as NSOrthographyMBS	1058
* 20.3.8 dominantLanguageForScript(script as string) as String	1058
* 20.3.9 languagesForScript(script as string) as String()	1058
* 20.3.10 orthographyWithDominantScript(script as string, map as dictionary) as NSOrthographyMBS	1059
* 20.3.12 dominantLanguage as String	1059
* 20.3.13 dominantScript as String	1059
* 20.3.14 Handle as Integer	1059
* 20.3.15 languageMap as Dictionary	1060

- **12 Cocoa Printing** 777
 - 12.1.1 class NSPageLayoutMBS 777
 - * 12.1.3 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow) 777
 - * 12.1.4 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS) 778
 - * 12.1.5 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window) 778
 - * 12.1.6 Constructor 778
 - * 12.1.7 pageLayout as NSPageLayoutMBS 779
 - * 12.1.8 printInfo as NSPrintInfoMBS 779
 - * 12.1.9 runModal as Integer 779
 - * 12.1.10 runModalWithPrintInfo(printInfo as NSPrintInfoMBS) as Integer 779
 - * 12.1.11 runPageLayout 780
 - * 12.1.13 Handle as Integer 780
 - * 12.1.15 printPanelDidEnd(returnCode as Integer) 780

	69
• 15 Cocoa Text	891
– 15.2.1 class NSParagraphStyleMBS	899
* 15.2.3 Constructor	899
* 15.2.4 copy as NSParagraphStyleMBS	899
* 15.2.5 defaultParagraphStyle as NSParagraphStyleMBS	900
* 15.2.6 defaultWritingDirectionForLanguage(languageName as string) as Integer	900
* 15.2.7 mutableCopy as NSMutableParagraphStyleMBS	900
* 15.2.8 NSCenterTextAlignment as Integer	900
* 15.2.9 NSRightTextAlignment as Integer	901
* 15.2.10 tabStops as NSTextTabMBS()	901
* 15.2.11 textBlocks as NSTextBlockMBS()	901
* 15.2.12 textLists as NSTextListMBS()	901
* 15.2.14 alignment as Integer	901
* 15.2.15 baseWritingDirection as Integer	902
* 15.2.16 defaultTabInterval as Double	902
* 15.2.17 firstLineHeadIndent as Double	902
* 15.2.18 firstTabStop as NSTextTabMBS	902
* 15.2.19 firstTextList as NSTextListMBS	903
* 15.2.20 Handle as Integer	903
* 15.2.21 headerLevel as Integer	903
* 15.2.22 headIndent as Double	903
* 15.2.23 hyphenationFactor as Double	903
* 15.2.24 lineBreakMode as Integer	904
* 15.2.25 lineHeightMultiple as Double	904
* 15.2.26 lineSpacing as Double	904
* 15.2.27 maximumLineHeight as Double	904
* 15.2.28 minimumLineHeight as Double	905
* 15.2.29 paragraphSpacing as Double	905
* 15.2.30 paragraphSpacingBefore as Double	905
* 15.2.31 tailIndent as Double	905
* 15.2.32 tighteningFactorForTruncation as Double	906

• 9 Cocoa Controls	453
– 9.22.1 class NSPathComponentCellMBS	586
* 9.22.3 Constructor(text as string)	586
* 9.22.5 File as folderitem	586
* 9.22.6 Image as NSImageMBS	586
* 9.22.7 URL as string	587
– 9.23.1 control NSPathControlControlMBS	588
* 9.23.3 View as NSPathControlMBS	588
* 9.23.5 Action	588
* 9.23.6 BoundsChanged	588
* 9.23.7 Close	589
* 9.23.8 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean	589
* 9.23.9 ContextualMenuAction(hitItem as MenuItem) as Boolean	589
* 9.23.10 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	589
* 9.23.11 DoubleClick	589
* 9.23.12 EnableMenuItems	590
* 9.23.13 FrameChanged	590
* 9.23.14 GotFocus	590
* 9.23.15 LostFocus	590
* 9.23.16MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	591
* 9.23.17 MouseDrag(x as Integer, y as Integer)	591
* 9.23.18 MouseUp(x As Integer, y As Integer)	591
* 9.23.19 Open	591
* 9.23.20 ScaleFactorChanged(NewFactor as double)	592
* 9.23.21 willDisplayOpenPanel(openPanel as Variant)	592
* 9.23.22 willPopUpMenu(menu as Variant)	592
* 9.23.23 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	592
– 9.24.1 class NSPathControlMBS	594
* 9.24.3 clickedPathComponentCell as NSPathComponentCellMBS	594
* 9.24.4 Constructor	595
* 9.24.5 Constructor(Handle as Integer)	595
* 9.24.6 Constructor(left as Double, top as Double, width as Double, height as Double)	595
* 9.24.7 pathComponentCells as NSPathComponentCellMBS()	596
* 9.24.8 setDraggingSourceOperationMask(mask as Integer, local as boolean)	596
* 9.24.9 setPathComponentCells(cells() as NSPathComponentCellMBS)	596
* 9.24.11 backgroundColor as NSColorMBS	597
* 9.24.12 File as folderitem	597
* 9.24.13 menu as NSMenuMBS	597
* 9.24.14 pathStyle as Integer	597
* 9.24.15 URL as string	598

	71
* 9.24.17 DoubleClick	598
* 9.24.18 willDisplayOpenPanel(openPanel as Variant)	598
* 9.24.19 willPopUpMenu(menu as Variant)	598

• 14 Cocoa Tasks	859
– 14.2.1 class NSPipeMBS	876
* 14.2.3 Constructor	876
* 14.2.4 fileHandleForReading as NSFileHandleMBS	876
* 14.2.5 fileHandleForWriting as NSFileHandleMBS	876
* 14.2.6 pipe as NSPipeMBS	877
* 14.2.8 Handle as Integer	877

• 9 Cocoa Controls	453
– 9.25.1 class NSPopUpButtonCellMBS	600
* 9.25.3 addItemWithTitle(itemTitles() as string)	600
* 9.25.4 addItemWithTitle(title as string)	601
* 9.25.5 Constructor(image as NSImageMBS)	601
* 9.25.6 Constructor(text as string, pullsDown as boolean)	601
* 9.25.7 dismissPopUp	601
* 9.25.8 indexOfItem(item as NSMenuItemMBS) as Integer	602
* 9.25.9 indexOfItemWithTag(tag as Integer) as Integer	602
* 9.25.10 indexOfItemWithTitle(title as String) as Integer	602
* 9.25.11 insertItemWithTitle(title as string, atIndex as Integer)	602
* 9.25.12 itemArray as NSMenuItemMBS()	603
* 9.25.13 itemAtIndex(Index as Integer) as NSMenuItemMBS	603
* 9.25.14 itemTitleAtIndex(Index as Integer) as String	603
* 9.25.15 itemTitles as String()	603
* 9.25.16 itemWithTitle(title as String) as NSMenuItemMBS	604
* 9.25.17 removeAllItems	604
* 9.25.18 removeItemAtIndex(Index as Integer)	604
* 9.25.19 removeItemWithTitle(title as string)	604
* 9.25.20 selectItem(item as NSMenuItemMBS)	604
* 9.25.21 selectItemAtIndex(Index as Integer)	605
* 9.25.22 selectItemWithTag(tag as Integer) as boolean	605
* 9.25.23 selectItemWithTitle(title as string)	605
* 9.25.24 setTitle(title as string)	606
* 9.25.25 synchronizeTitleAndSelectedItem	606
* 9.25.27 altersStateOfSelectedItem as Boolean	606
* 9.25.28 arrowPosition as Integer	606
* 9.25.29 autoenablesItems as Boolean	607
* 9.25.30 indexOfSelectedItem as Integer	607
* 9.25.31 lastItem as NSMenuItemMBS	607
* 9.25.32 menu as NSMenuMBS	607
* 9.25.33 numberOfItems as Integer	608
* 9.25.34 preferredEdge as Integer	608
* 9.25.35 pullsDown as Boolean	608
* 9.25.36 selectedItem as NSMenuItemMBS	609
* 9.25.37 titleOfSelectedItem as String	609
* 9.25.38 usesItemFromMenu as Boolean	609

• 12 Cocoa Printing	777
– 12.2.1 class NSPrinterMBS	781
* 12.2.3 booleanForKey(key as string, table as string) as boolean	781
* 12.2.4 Constructor(name as string = "")	781
* 12.2.5 copy as NSPrinterMBS	782
* 12.2.6 defaultPrinter as NSPrinterMBS	782
* 12.2.7 deviceDescription as Dictionary	782
* 12.2.8 floatForKey(key as string, table as string) as Double	782
* 12.2.9 intForKey(key as string, table as string) as Integer	782
* 12.2.10 isKey(key as string, table as string) as boolean	783
* 12.2.11 languageLevel as Integer	783
* 12.2.12 name as string	783
* 12.2.13 pageSizeForPaper(paperName as string) as NSSizeMBS	783
* 12.2.14 printerNames as string()	784
* 12.2.15 printerTypes as string()	784
* 12.2.16 printerWithName(name as string) as NSPrinterMBS	784
* 12.2.17 printerWithType(type as string) as NSPrinterMBS	784
* 12.2.18 rectForKey(key as string, table as string) as NSRectMBS	785
* 12.2.19 sizeForKey(key as string, table as string) as NSSizeMBS	785
* 12.2.20 statusForTable(paperName as string) as Integer	785
* 12.2.21 stringForKey(key as string, table as string) as string	786
* 12.2.22 stringListForKey(key as string, table as string) as string()	786
* 12.2.23 type as string	786
* 12.2.25 Handle as Integer	786
– 12.3.1 class NSPrintInfoMBS	788
* 12.3.3 Constructor	789
* 12.3.4 Constructor(attributes as Dictionary)	790
* 12.3.5 Constructor(Data as Memoryblock)	790
* 12.3.6 copy as NSPrintInfoMBS	790
* 12.3.7 defaultPrinter as NSPrinterMBS	790
* 12.3.8 NSPrintAllPages as string	790
* 12.3.9 NSPrintBottomMargin as string	791
* 12.3.10 NSPrintCancelJob as string	791
* 12.3.11 NSPrintCopies as string	791
* 12.3.12 NSPrintDetailedErrorReporting as string	791
* 12.3.13 NSPrintFaxNumber as string	791
* 12.3.14 NSPrintFirstPage as string	792
* 12.3.15 NSPrintHeaderAndFooter as string	792
* 12.3.16 NSPrintHorizontallyCentered as string	792
* 12.3.17 NSPrintHorizontalPagination as string	792
* 12.3.18 NSPrintJobDisposition as string	792

* 12.3.19 NSPrintJobSavingFileNameExtensionHidden as string	792
* 12.3.20 NSPrintJobSavingURL as string	793
* 12.3.21 NSPrintLastPage as string	793
* 12.3.22 NSPrintLeftMargin as string	793
* 12.3.23 NSPrintMustCollate as string	793
* 12.3.24 NSPrintOrientation as string	793
* 12.3.25 NSPrintPagesAcross as string	794
* 12.3.26 NSPrintPagesDown as string	794
* 12.3.27 NSPrintPaperName as string	794
* 12.3.28 NSPrintPaperSize as string	794
* 12.3.29 NSPrintPreviewJob as string	794
* 12.3.30 NSPrintPrinter as string	795
* 12.3.31 NSPrintPrinterName as string	795
* 12.3.32 NSPrintReversePageOrder as string	795
* 12.3.33 NSPrintRightMargin as string	795
* 12.3.34 NSPrintSaveJob as string	795
* 12.3.35 NSPrintScalingFactor as string	795
* 12.3.36 NSPrintSelectionOnly as string	796
* 12.3.37 NSPrintSpoolJob as string	796
* 12.3.38 NSPrintTime as string	796
* 12.3.39 NSPrintTopMargin as string	796
* 12.3.40 NSPrintVerticallyCentered as string	796
* 12.3.41 NSPrintVerticalPagination as string	797
* 12.3.42 SetSaveDestination(file as folderitem)	797
* 12.3.43 setSharedPrintInfo(printInfo as NSPrintInfoMBS)	797
* 12.3.44 setUpPrintOperationDefaultValues	798
* 12.3.45 sharedPrintInfo as NSPrintInfoMBS	798
* 12.3.47 bottomMargin as Double	798
* 12.3.48 data as Memoryblock	798
* 12.3.49 dictionary as dictionary	799
* 12.3.50 Handle as Integer	799
* 12.3.51 HorizontallyCentered as boolean	799
* 12.3.52 horizontalPagination as Integer	800
* 12.3.53 imageablePageBounds as NSRectMBS	800
* 12.3.54 jobDisposition as string	800
* 12.3.55 leftMargin as Double	801
* 12.3.56 localizedPaperName as string	801
* 12.3.57 orientation as Integer	801
* 12.3.58 paperName as string	801
* 12.3.59 paperSize as NSSizeMBS	802
* 12.3.60 printer as NSPrinterMBS	802
* 12.3.61 printerName as String	802

* 12.3.62	printSettings as dictionary	802
* 12.3.63	rightMargin as Double	802
* 12.3.64	scalingFactor as Double	803
* 12.3.65	SelectionOnly as boolean	803
* 12.3.66	SetupString as Memoryblock	803
* 12.3.67	topMargin as Double	804
* 12.3.68	VerticallyCentered as boolean	804
* 12.3.69	verticalPagination as Integer	804
– 12.4.1	class NSPrintOperationMBS	806
* 12.4.3	Constructor	806
* 12.4.4	Constructor(other as NSPrintOperationMBS)	807
* 12.4.5	Constructor(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil)	807
* 12.4.6	Constructor(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil)	808
* 12.4.7	Constructor(view as NSViewMBS)	809
* 12.4.8	Constructor(view as NSViewMBS, printInfo as NSPrintInfoMBS)	809
* 12.4.9	context as NSGraphicsMBS	810
* 12.4.10	currentOperation as NSPrintOperationMBS	810
* 12.4.11	currentPage as Integer	810
* 12.4.12	data as Memoryblock	810
* 12.4.13	Destructor	810
* 12.4.14	EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS	811
* 12.4.15	EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS	811
* 12.4.16	EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS	812
* 12.4.17	EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string) as NSPrintOperationMBS	812
* 12.4.18	isCopyingOperation as boolean	813
* 12.4.19	NSPrintOperationExistsException as string	813
* 12.4.20	pageRange as NSRangeMBS	813
* 12.4.21	PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS	813
* 12.4.22	PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS	814
* 12.4.23	PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS	815
* 12.4.24	PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string) as NSPrintOperationMBS	815
* 12.4.25	preferredRenderingQuality as Integer	816
* 12.4.26	printOperationWithView(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS	816

* 12.4.27 printOperationWithView(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS	817
* 12.4.28 printOperationWithView(view as NSViewMBS) as NSPrintOperationMBS	818
* 12.4.29 printOperationWithView(view as NSViewMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS	819
* 12.4.30 runOperation as boolean	819
* 12.4.31 runOperationModalForWindow(win as DesktopWindow)	819
* 12.4.32 runOperationModalForWindow(win as NSWindowMBS)	820
* 12.4.33 runOperationModalForWindow(win as window)	820
* 12.4.34 setCurrentOperation(operation as NSPrintOperationMBS)	820
* 12.4.35 view as NSViewMBS	820
* 12.4.37 Handle as Integer	821
* 12.4.38 canSpawnSeparateThread as boolean	821
* 12.4.39 jobTitle as string	821
* 12.4.40 pageOrder as Integer	822
* 12.4.41 printInfo as NSPrintInfoMBS	822
* 12.4.42 printPanel as NSPrintPanelMBS	822
* 12.4.43 showsPrintPanel as boolean	822
* 12.4.44 showsProgressPanel as boolean	822
* 12.4.46 printOperationDidRun(success as boolean)	823
– 12.5.1 class NSPrintPanelMBS	824
* 12.5.3 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow)	824
* 12.5.4 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS)	824
* 12.5.5 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window)	825
* 12.5.6 Constructor	825
* 12.5.7 NSPrintAllPresetsJobStyleHint as string	825
* 12.5.8 NSPrintNoPresetsJobStyleHint as string	825
* 12.5.9 NSPrintPhotoJobStyleHint as string	826
* 12.5.10 printInfo as NSPrintInfoMBS	826
* 12.5.11 printPanel as NSPrintPanelMBS	826
* 12.5.12 runModal as Integer	826
* 12.5.13 runModalWithPrintInfo(printInfo as NSPrintInfoMBS) as Integer	827
* 12.5.15 Handle as Integer	827
* 12.5.16 defaultButtonTitle as string	827
* 12.5.17 helpAnchor as string	827
* 12.5.18 jobStyleHint as string	828
* 12.5.19 options as Integer	828
* 12.5.21 printPanelDidEnd(returnCode as Integer)	828

• 13 Cocoa Regular Expressions	831
– 13.2.1 class NSRegularExpressionMBS	835
* 13.2.3 Constructor(script as string, options as Integer, byref error as NSErrorMBS)	835
* 13.2.4 copy as NSRegularExpressionMBS	836
* 13.2.5 enumerateMatches(text as string, options as Integer, range as NSRangeMBS = nil)	836
* 13.2.6 escapedPatternForString(text as String) as String	837
* 13.2.7 escapedTemplateForString(text as String) as String	837
* 13.2.8 firstMatch(text as string, options as Integer, range as NSRangeMBS = nil) as NSTextCheckingResultMBS	838
* 13.2.9 matches(text as string, options as Integer, range as NSRangeMBS = nil) as NSTextCheckingResultMBS()	838
* 13.2.10 numberOfMatches(text as string, options as Integer, range as NSRangeMBS = nil) as Integer	839
* 13.2.11 rangeOfFirstMatch(text as string, options as Integer, range as NSRangeMBS = nil) as NSRangeMBS	839
* 13.2.12 regularExpressionWithPattern(pattern as String, options as Integer, byref error as NSErrorMBS) as NSDataDetectorMBS	839
* 13.2.13 replaceMatches(byref text as string, options as Integer, range as NSRangeMBS = nil, template as String) as Integer	840
* 13.2.14 replacementStringForResult(result as NSTextCheckingResultMBS, text as string, offset as Integer, template as String) as String	840
* 13.2.15 stringByReplacingMatches(text as string, options as Integer, range as NSRangeMBS = nil, template as String) as String	841
* 13.2.17 Handle as Integer	841
* 13.2.18 numberOfCaptureGroups as Integer	841
* 13.2.19 options as Integer	841
* 13.2.20 pattern as String	842
* 13.2.22 enumerateMatch(text as String, options as Integer, result as NSTextCheckingResultMBS, flags as Integer, byref stop as Boolean)	842

	79
• 8 Cocoa	271
– 8.8.1 class NSRunLoopMBS	370
* 8.8.3 AddDummyPort	370
* 8.8.4 allModes as string()	370
* 8.8.5 Constructor	371
* 8.8.6 currentRunLoop as NSRunLoopMBS	371
* 8.8.7 mainRunLoop as NSRunLoopMBS	371
* 8.8.8 NSDefaultRunLoopMode as string	371
* 8.8.9 NSRunLoopCommonModes as string	371
* 8.8.10 run	372
* 8.8.11 run(Seconds as Double)	372
* 8.8.12 runMode(Mode as string, Seconds as Double) as boolean	372
* 8.8.13 runModeUntilDate(Mode as string, limitDate as date) as boolean	372
* 8.8.14 runModeUntilDate(Mode as string, limitDate as dateTime) as boolean	373
* 8.8.15 runOnce	374
* 8.8.16 runUntilDate(limitDate as date)	374
* 8.8.17 runUntilDate(limitDate as dateTime)	374
* 8.8.19 currentMode as String	375
* 8.8.20 Handle as Integer	375

• 22 Process	1081
– 22.4.1 class NSRunningApplicationMBS	1084
* 22.4.3 activateWithOptions(options as Integer) as boolean	1084
* 22.4.4 Constructor	1085
* 22.4.5 currentApplication as NSRunningApplicationMBS	1085
* 22.4.6 forceTerminate as boolean	1085
* 22.4.7 hide as boolean	1086
* 22.4.8 runningApplications as NSRunningApplicationMBS()	1086
* 22.4.9 runningApplicationsWithBundleIdentifier(bundleID as string) as NSRunningApplicationMBS()	1087
* 22.4.10 runningApplicationWithProcessIdentifier(pid as Integer) as NSRunningApplicationMBS	1087
* 22.4.11 terminate as boolean	1088
* 22.4.12 unhide as boolean	1088
* 22.4.14 activationPolicy as Integer	1089
* 22.4.15 active as boolean	1089
* 22.4.16 bundleIdentifier as string	1089
* 22.4.17 bundleURL as string	1090
* 22.4.18 executableArchitecture as Integer	1090
* 22.4.19 executableURL as string	1090
* 22.4.20 finishedLaunching as boolean	1091
* 22.4.21 Handle as Integer	1091
* 22.4.22 hidden as boolean	1091
* 22.4.23 icon as NSImageMBS	1092
* 22.4.24 launchDate as date	1092
* 22.4.25 launchDateTime as DateTime	1092
* 22.4.26 localizedName as string	1093
* 22.4.27 ownsMenuBar as boolean	1093
* 22.4.28 processIdentifier as Integer	1093
* 22.4.29 terminated as boolean	1094

• 21 Navigation	1061
– 21.2.1 class NSSavePanelMBS	1069
* 21.2.3 allowedFileTypes as string()	1070
* 21.2.4 beginSheetForDirectory(path as folderitem, name as string, targetWindow as DesktopWindow)	1070
* 21.2.5 beginSheetForDirectory(path as folderitem, name as string, targetWindow as window)	1070
* 21.2.6 Cancel	1071
* 21.2.7 Constructor	1071
* 21.2.8 File as folderitem	1071
* 21.2.9 FileTypeForHFSType(hfstype as string) as string	1071
* 21.2.10 HideNSNavNodePopUpButton	1072
* 21.2.11 Ok	1072
* 21.2.12 runModal as Integer	1072
* 21.2.13 runModalForDirectory(path as folderitem, name as string) as Integer	1072
* 21.2.14 setAllowedFileTypes filetype as string)	1073
* 21.2.15 setAllowedFileTypes(filetypes() as string)	1073
* 21.2.16 setTagNames(tagNames() as string)	1073
* 21.2.17 tagNames as string()	1073
* 21.2.18 validateVisibleColumns	1074
* 21.2.20 accessoryView as NSViewMBS	1074
* 21.2.21 allowsOtherFileTypes as boolean	1074
* 21.2.22 canCreateDirectories as boolean	1075
* 21.2.23 canSelectHiddenExtension as boolean	1075
* 21.2.24 Directory as folderitem	1075
* 21.2.25 directoryURL as string	1075
* 21.2.26 isExpanded as boolean	1075
* 21.2.27 isExtensionHidden as boolean	1075
* 21.2.28 Message as string	1076
* 21.2.29 NameFieldLabel as string	1076
* 21.2.30 nameFieldStringValue as string	1076
* 21.2.31 Prompt as string	1076
* 21.2.32 requiredFileType as string	1077
* 21.2.33 showsHiddenFiles as boolean	1077
* 21.2.34 showsTagField as Boolean	1077
* 21.2.35 Title as string	1078
* 21.2.36 treatsFilePackagesAsDirectories as boolean	1078
* 21.2.38 compareFilename(name1 as string, name2 as string, caseSensitive as boolean) as Integer	1078
* 21.2.39 directoryDidChange(path as string, folder as folderitem)	1079
* 21.2.40 isValidFilename(path as string, item as folderitem) as boolean	1079

* 21.2.41	panelSelectionDidChange	1079
* 21.2.42	savePanelDidEnd(ReturnCode as Integer)	1079
* 21.2.43	shouldShowFilename(path as string, item as folderitem) as boolean	1079
* 21.2.44	userEnteredFilename(filename as string, confirmed as boolean) as string	1080
* 21.2.45	willExpand(expanding as boolean)	1080

	83
• 8 Cocoa	271
– 8.9.1 class NSSavePanelObserverMBS	376
* 8.9.3 Enabled as Boolean	376
* 8.9.5 DidShowPanel(panel as NSSavePanelMBS, Result as Integer)	376
* 8.9.6 WillShowPanel(panel as NSSavePanelMBS)	376

• 9 Cocoa Controls	453
– 9.26.1 class NSSearchFieldCellMBS	610
* 9.26.3 Constructor(image as NSImageMBS)	610
* 9.26.4 Constructor(text as string)	610
* 9.26.5 recentSearches as String()	611
* 9.26.6 resetCancelButtonCell	611
* 9.26.7 resetSearchButtonCell	611
* 9.26.8 setRecentSearches(RecentSearches() as String = nil)	611
* 9.26.10 cancelButtonCell as NSButtonCellMBS	612
* 9.26.11 maximumRecents as Integer	612
* 9.26.12 recentsAutosaveName as String	612
* 9.26.13 searchButtonCell as NSButtonCellMBS	612
* 9.26.14 searchMenuTemplate as NSMenuMBS	612
* 9.26.15 sendsSearchStringImmediately as Boolean	613
* 9.26.16 sendsWholeSearchString as Boolean	613
– 9.27.1 control NSSearchFieldControlMBS	614
* 9.27.3 View as NSSearchFieldMBS	614
* 9.27.5 Action	614
* 9.27.6 BoundsChanged	615
* 9.27.7 Close	615
* 9.27.8 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean	615
* 9.27.9 ContextualMenuAction(hitItem as MenuItem) as Boolean	615
* 9.27.10 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	615
* 9.27.11 EnableMenuItems	616
* 9.27.12 FrameChanged	616
* 9.27.13 GotFocus	616
* 9.27.14 LostFocus	616
* 9.27.15MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	616
* 9.27.16 MouseDrag(x as Integer, y as Integer)	617
* 9.27.17 MouseUp(x as Integer, y as Integer)	617
* 9.27.18 Open	617
* 9.27.19 ScaleFactorChanged(NewFactor as Double)	617
* 9.27.20 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	618
* 9.27.21 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	618
* 9.27.22 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	618
* 9.27.23 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean	619
* 9.27.24 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean	619
* 9.27.25 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	619

	85
– 9.28.1 class NSSearchFieldMBS	620
* 9.28.3 Constructor	620
* 9.28.4 Constructor(Handle as Integer)	621
* 9.28.5 Constructor(left as Double, top as Double, width as Double, height as Double)	621
* 9.28.6 recentSearches as string()	622
* 9.28.7 setRecentSearches(values() as string)	622
* 9.28.9 Scrollable as Boolean	622
* 9.28.10 maximumRecents as Integer	622
* 9.28.11 recentsAutosaveName as string	623
* 9.28.12 searchMenuTemplate as NSMenuMBS	623
* 9.28.13 sendsSearchStringImmediately as boolean	623
* 9.28.14 sendsWholeSearchString as boolean	623
– 9.29.1 control NSSecureTextFieldControlMBS	625
* 9.29.3 echosBullets as Boolean	625
* 9.29.4 View as NSSecureTextFieldMBS	625
* 9.29.6 Action	626
* 9.29.7 BoundsChanged	626
* 9.29.8 Close	626
* 9.29.9 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean	626
* 9.29.10 ContextualMenuItemAction(hitItem as MenuItem) as Boolean	626
* 9.29.11 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	627
* 9.29.12 EnableMenuItems	627
* 9.29.13 FrameChanged	627
* 9.29.14 GotFocus	627
* 9.29.15 LostFocus	627
* 9.29.16MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	628
* 9.29.17 MouseDrag(x as Integer, y as Integer)	628
* 9.29.18 MouseUp(x as Integer, y as Integer)	628
* 9.29.19 Open	628
* 9.29.20 ScaleFactorChanged(NewFactor as Double)	629
* 9.29.21 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	629
* 9.29.22 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	629
* 9.29.23 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	630
* 9.29.24 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean	630
* 9.29.25 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean	630
* 9.29.26 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	630
– 9.30.1 class NSSecureTextFieldMBS	631
* 9.30.3 Constructor	631

* 9.30.4 Constructor(Handle as Integer)	631
* 9.30.5 Constructor(left as Double, top as Double, width as Double, height as Double)	632
* 9.30.7 echosBullets as boolean	632
– 9.31.1 control NSSegmentedControlControlMBS	633
* 9.31.3 View as NSSegmentedControlMBS	633
* 9.31.5 Action	633
* 9.31.6 BoundsChanged	633
* 9.31.7 Close	634
* 9.31.8 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean	634
* 9.31.9 ContextualMenuItemAction(hitItem as MenuItem) as Boolean	634
* 9.31.10 didCloseContextualMenu(menu as NSMenuItemMBS, NSEvent as NSEventMBS)	634
* 9.31.11 EnableMenuItems	634
* 9.31.12 FrameChanged	635
* 9.31.13 GotFocus	635
* 9.31.14 LostFocus	635
* 9.31.15MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	635
* 9.31.16 MouseDrag(x as Integer, y as Integer)	636
* 9.31.17 MouseUp(x As Integer, y As Integer)	636
* 9.31.18 Open	636
* 9.31.19 ScaleFactorChanged(NewFactor as double)	636
* 9.31.20 willShowContextualMenu(menu as NSMenuItemMBS, NSEvent as NSEventMBS)	636
– 9.32.1 class NSSegmentedControlMBS	638
* 9.32.3 Constructor	639
* 9.32.4 Constructor(Handle as Integer)	639
* 9.32.5 Constructor(left as Double, top as Double, width as Double, height as Double)	640
* 9.32.6 makeNextSegmentKey	640
* 9.32.7 makePreviousSegmentKey	640
* 9.32.8 selectSegmentWithTag(Tag as Integer) as Boolean	640
* 9.32.10 cellTrackingMode as Integer	641
* 9.32.11 doubleValueForSelectedSegment as Double	641
* 9.32.12 segmentCount as Integer	641
* 9.32.13 segmentStyle as Integer	641
* 9.32.14 selectedSegment as Integer	642
* 9.32.15 springLoaded as Boolean	642
* 9.32.16 trackingMode as Integer	642
* 9.32.17 imageForSegment(segment as Integer) as NSImageMBS	642
* 9.32.18 imageScalingForSegment(segment as Integer) as Integer	643
* 9.32.19 isEnabledForSegment(segment as Integer) as Boolean	643
* 9.32.20 isSelectedForSegment(segment as Integer) as Boolean	643
* 9.32.21 labelForSegment(segment as Integer) as string	643

* 9.32.22 menuForSegment(segment as Integer) as NSMenuMBS	644
* 9.32.23 tagForSegment(segment as Integer) as Integer	644
* 9.32.24 ToolTipForSegment(segment as Integer) as string	644
* 9.32.25 widthForSegment(segment as Integer) as Double	645

• 8 Cocoa	271
– 8.10.1 class NSServiceProviderMBS	378
* 8.10.3 Constructor	378
* 8.10.4 Destructor	378
* 8.10.6 Handle as Integer	378
* 8.10.8 readSelectionFromPasteboard(pboard as NSPasteboardMBS) as Boolean	379
* 8.10.9 ServiceInvoked(pboard as NSPasteboardMBS, userData as string, byref error as string)	379
* 8.10.10 validRequestor(sendType as String, returnType as String) as Boolean	379
* 8.10.11 writeSelectionToPasteboard(pboard as NSPasteboardMBS, types() as String) as Boolean	379

• 23 Speech	1097
– 23.1.1 class NSSpeechRecognizerMBS	1097
* 23.1.3 commands as string()	1097
* 23.1.4 Destructor	1097
* 23.1.5 SetCommands(commands() as string)	1098
* 23.1.6 StartListening	1098
* 23.1.7 StopListening	1098
* 23.1.9 BlocksOtherRecognizers as boolean	1098
* 23.1.10 DisplayedCommandsTitle as string	1098
* 23.1.11 ListensInForegroundOnly as boolean	1099
* 23.1.13 DidRecognizeCommand(command as string)	1099
– 23.2.1 class NSSpeechSynthesizerMBS	1100
* 23.2.3 addSpeechDictionary(speechDictionary as dictionary)	1100
* 23.2.4 attributesForVoice(voice as String) as NSVoiceMBS	1100
* 23.2.5 availableVoice(index as Integer) as String	1100
* 23.2.6 availableVoices as String()	1101
* 23.2.7 availableVoicesCount as Integer	1101
* 23.2.8 Constructor	1101
* 23.2.9 Constructor(voice as string)	1101
* 23.2.10 continueSpeaking	1102
* 23.2.11 defaultVoice as String	1102
* 23.2.12 Destructor	1102
* 23.2.13 isAnyApplicationSpeaking as boolean	1102
* 23.2.14 NSSpeechCharacterModeProperty as String	1102
* 23.2.15 NSSpeechCommandDelimiterProperty as String	1103
* 23.2.16 NSSpeechCommandPrefix as String	1103
* 23.2.17 NSSpeechCommandSuffix as String	1104
* 23.2.18 NSSpeechCurrentVoiceProperty as String	1104
* 23.2.19 NSSpeechDictionaryAbbreviations as String	1104
* 23.2.20 NSSpeechDictionaryEntryPhonemes as String	1104
* 23.2.21 NSSpeechDictionaryEntrySpelling as String	1105
* 23.2.22 NSSpeechDictionaryLocaleIdentifier as String	1105
* 23.2.23 NSSpeechDictionaryModificationDate as String	1105
* 23.2.24 NSSpeechDictionaryPronunciations as String	1105
* 23.2.25 NSSpeechErrorCount as String	1106
* 23.2.26 NSSpeechErrorNewestCharacterOffset as String	1106
* 23.2.27 NSSpeechErrorNewestCode as String	1106
* 23.2.28 NSSpeechErrorOldestCharacterOffset as String	1106
* 23.2.29 NSSpeechErrorOldestCode as String	1107
* 23.2.30 NSSpeechErrorsProperty as String	1107
* 23.2.31 NSSpeechInputModeProperty as String	1107

* 23.2.32 NSSpeechModeLiteral as String	1107
* 23.2.33 NSSpeechModeNormal as String	1108
* 23.2.34 NSSpeechModePhoneme as String	1108
* 23.2.35 NSSpeechModeText as String	1108
* 23.2.36 NSSpeechNumberModeProperty as String	1108
* 23.2.37 NSSpeechOutputToFileURLProperty as String	1109
* 23.2.38 NSSpeechPhonemeInfoExample as String	1109
* 23.2.39 NSSpeechPhonemeInfoHiliteEnd as String	1109
* 23.2.40 NSSpeechPhonemeInfoHiliteStart as String	1110
* 23.2.41 NSSpeechPhonemeInfoOpcode as String	1110
* 23.2.42 NSSpeechPhonemeInfoSymbol as String	1110
* 23.2.43 NSSpeechPhonemeSymbolsProperty as String	1110
* 23.2.44 NSSpeechPitchBaseProperty as String	1111
* 23.2.45 NSSpeechPitchModProperty as String	1111
* 23.2.46 NSSpeechRateProperty as String	1112
* 23.2.47 NSSpeechRecentSyncProperty as String	1112
* 23.2.48 NSSpeechResetProperty as String	1113
* 23.2.49 NSSpeechStatusNumberOfCharactersLeft as String	1113
* 23.2.50 NSSpeechStatusOutputBusy as String	1113
* 23.2.51 NSSpeechStatusOutputPaused as String	1113
* 23.2.52 NSSpeechStatusPhonemeCode as String	1114
* 23.2.53 NSSpeechStatusProperty as String	1114
* 23.2.54 NSSpeechSynthesizerInfoIdentifier as String	1114
* 23.2.55 NSSpeechSynthesizerInfoProperty as String	1115
* 23.2.56 NSSpeechSynthesizerInfoVersion as String	1115
* 23.2.57 NSSpeechVolumeProperty as String	1115
* 23.2.58 objectForProperty(PropertyName as string, byref error as NSErrorMBS) as Variant	1116
* 23.2.59 pauseSpeakingAtBoundary(boundary as Integer)	1116
* 23.2.60 phonemesFromText(text as string) as string	1116
* 23.2.61 setObjectForProperty(value as Variant, PropertyName as string, byref error as NSErrorMBS) as boolean	1117
* 23.2.62 SetVoice(voice as string) as boolean	1117
* 23.2.63 StartSpeakingString(text as string) as boolean	1117
* 23.2.64 StartSpeakingString(text as string, file as folderitem) as boolean	1118
* 23.2.65 startSpeakingString(Text as String, URL as string) as boolean	1118
* 23.2.66 StopSpeaking	1118
* 23.2.67 stopSpeakingAtBoundary(boundary as Integer)	1119
* 23.2.69 IsSpeaking as boolean	1119
* 23.2.70 rate as Double	1119
* 23.2.71 UsesFeedbackWindow as boolean	1119
* 23.2.72 Voice as string	1120

- * 23.2.73 volume as Double 1120
- * 23.2.75 didEncounterErrorAtIndex(characterIndex as Integer, text as string, message as string) 1120
- * 23.2.76 didEncounterSyncMessage(message as string) 1121
- * 23.2.77 didFinishSpeaking(finishedSpeaking as boolean) 1121
- * 23.2.78 willSpeakPhoneme(phonemeOpcode as Integer) 1121
- * 23.2.79 willSpeakWord(Position as Integer, Length as Integer, Text as String) 1121

- **24 Spell Checking** 1129
 - 24.1.1 class NSSpellCheckerMBS 1129
 - * 24.1.3 availableLanguages as string() 1130
 - * 24.1.4 checkGrammarOfString(text as string, start as Integer, language as string, wrap as boolean) as NSRangeMBS 1130
 - * 24.1.5 checkGrammarOfString(text as string, start as Integer, language as string, wrap as boolean, Details() as dictionary) as NSRangeMBS 1131
 - * 24.1.6 checkSpellingOfString(text as string, start as Integer) as NSRangeMBS 1131
 - * 24.1.7 checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean) as NSRangeMBS 1131
 - * 24.1.8 checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean, byref WordCount as Integer) as NSRangeMBS 1132
 - * 24.1.9 checkString(text as string, range as NSRangeMBS = nil, checkingTypes as Int64 = -1, options as Dictionary = nil, byref orthography as NSOrthographyMBS, byref wordCount as Integer) as NSTextCheckingResultMBS() 1132
 - * 24.1.10 completionsForPartialWordRange(start as Integer, length as Integer, text as string, language as string="") as string() 1132
 - * 24.1.11 correctionForWordRange(range as NSRangeMBS, text as string, language as string) as string 1133
 - * 24.1.12 countWordsInString(word as string, language as string="") as Integer 1133
 - * 24.1.13 deletesAutospaceBetweenString(precedingString as string, followingString as string, language as String = "") as Boolean 1134
 - * 24.1.14 dismissCorrectionIndicatorForView(view as NSViewMBS) 1134
 - * 24.1.15 forgetWord(word as string) 1134
 - * 24.1.16 guessesForWord(range as NSRangeMBS, word as string, language as string) as string() 1134
 - * 24.1.17 guessesForWord(word as string) as string() 1135
 - * 24.1.18 hasLearnedWord(word as string) as boolean 1135
 - * 24.1.19 ignoredWords as string() 1135
 - * 24.1.20 ignoreWord(word as string) 1136
 - * 24.1.21 isAutomaticCapitalizationEnabled as boolean 1136
 - * 24.1.22 isAutomaticDashSubstitutionEnabled as boolean 1136
 - * 24.1.23 isAutomaticPeriodSubstitutionEnabled as boolean 1136
 - * 24.1.24 isAutomaticQuoteSubstitutionEnabled as boolean 1136
 - * 24.1.25 isAutomaticSpellingCorrectionEnabled as boolean 1137
 - * 24.1.26 isAutomaticTextCompletionEnabled as boolean 1137
 - * 24.1.27 isAutomaticTextReplacementEnabled as boolean 1137
 - * 24.1.28 languageForWordRange(range as NSRangeMBS, text as string, orthography as NSOrthographyMBS = nil) as string 1137
 - * 24.1.29 languageMenuEntries as string() 1137
 - * 24.1.30 learnWord(word as string) 1138
 - * 24.1.31 menuForResult(TextCheckingResult as NSTextCheckingResultMBS, checkedString as string, options as Dictionary = nil, atLocation as NSPointMBS, view as NSViewMBS) as NSMenuMBS 1138

* 24.1.32 NSSpellCheckerDidChangeAutomaticCapitalizationNotification as string	1139
* 24.1.33 NSSpellCheckerDidChangeAutomaticDashSubstitutionNotification as string	1139
* 24.1.34 NSSpellCheckerDidChangeAutomaticPeriodSubstitutionNotification as string	1139
* 24.1.35 NSSpellCheckerDidChangeAutomaticQuoteSubstitutionNotification as string	1139
* 24.1.36 NSSpellCheckerDidChangeAutomaticSpellingCorrectionNotification as string	1139
* 24.1.37 NSSpellCheckerDidChangeAutomaticTextCompletionNotification as string	1140
* 24.1.38 NSSpellCheckerDidChangeAutomaticTextReplacementNotification as string	1140
* 24.1.39 NSTextCheckingDocumentAuthorKey as string	1140
* 24.1.40 NSTextCheckingDocumentTitleKey as string	1140
* 24.1.41 NSTextCheckingDocumentURLKey as string	1140
* 24.1.42 NSTextCheckingOrthographyKey as string	1141
* 24.1.43 NSTextCheckingQuotesKey as string	1141
* 24.1.44 NSTextCheckingReferenceDateKey as string	1141
* 24.1.45 NSTextCheckingReferenceTimeZoneKey as string	1141
* 24.1.46 NSTextCheckingRegularExpressionsKey as string	1142
* 24.1.47 NSTextCheckingReplacementsKey as string	1142
* 24.1.48 NSTextCheckingSelectedRangeKey as string	1142
* 24.1.49 preventsAutocorrectionBeforeString(text as string, language as String = "") as Boolean	1142
* 24.1.50 recordResponse(response as Integer, correction as string, word as string, language as String = "")	1143
* 24.1.51 requestCandidatesForSelectedRange(selectedRange as NSRangeMBS, stringToCheck as string, types as Int64 = -1, options as Dictionary = nil, tag as Variant = nil) as Integer	1143
* 24.1.52 requestCheckingOfString(stringToCheck as string, range as NSRangeMBS, types as Int64 = -1, options as Dictionary = nil, tag as Variant = nil) as Integer	1143
* 24.1.53 setIgnoredWords(words() as string)	1144
* 24.1.54 setLanguage(language as string) as boolean	1144
* 24.1.55 sharedSpellCheckerExists as boolean	1145
* 24.1.56 spellingPanel as NSPanelMBS	1145
* 24.1.57 unlearnWord(word as string)	1145
* 24.1.58 updatePanels	1145
* 24.1.59 updateSpellingPanelWithGrammarString(lang as string, detail as dictionary)	1145
* 24.1.60 updateSpellingPanelWithMisspelledWord(word as string)	1146
* 24.1.61 userPreferredLanguages as string()	1146
* 24.1.62 userQuotesArrayForLanguage(lang as string) as string()	1146
* 24.1.63 userReplacementsDictionary as dictionary	1147
* 24.1.65 accessoryView as NSViewMBS	1147
* 24.1.66 automaticallyIdentifiesLanguages as boolean	1147
* 24.1.67 Handle as Integer	1148
* 24.1.68 language as string	1148
* 24.1.69 Length as Integer	1148

* 24.1.70 Location as Integer	1149
* 24.1.71 substitutionsPanel as NSPanelMBS	1149
* 24.1.72 substitutionsPanelAccessoryViewController as NSViewControllerMBS	1149
* 24.1.73 Tag as Integer	1149
* 24.1.74 WordFieldValue as string	1150
* 24.1.76 Correct	1150
* 24.1.77 FindNext	1150
* 24.1.78 Ignore	1150
* 24.1.79 requestCandidatesForSelectedRangeCompleted(sequenceNumber as Integer, candidates() as NSTextCheckingResultMBS, stringToCheck as String, selectedRange as NSRangeMBS, checkingTypes as Int64, options as Dictionary, tag as Variant)	1150
* 24.1.80 requestCheckingOfStringCompleted(sequenceNumber as Integer, results() as NSTextCheckingResultMBS, orthography as NSOrthographyMBS, wordCount as Integer, stringToCheck as String, Range as NSRangeMBS, checkingTypes as Int64, options as Dictionary, tag as Variant)	1151

	95
• 9 Cocoa Controls	453
– 9.33.1 control NSSplitViewControlMBS	647
* 9.33.3 View as NSSplitViewMBS	647
* 9.33.5 BoundsChanged	647
* 9.33.6 canCollapseSubview(subview as NSViewMBS) as Boolean	647
* 9.33.7 Close	648
* 9.33.8 constrainMaxCoordinate(proposedMaximumPosition as double, dividerIndex as Integer) as Double	648
* 9.33.9 constrainMinCoordinate(proposedMinimumPosition as double, dividerIndex as Integer) as Double	649
* 9.33.10 constrainSplitPosition(proposedPosition as double, dividerIndex as Integer) as Double	649
* 9.33.11 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean	649
* 9.33.12 ContextualMenuItemAction(hitItem as MenuItem) as Boolean	650
* 9.33.13 didCloseContextualMenu(menu as NSMenuItemMBS, NSEvent as NSEventMBS)	650
* 9.33.14 drawDivider(graphics as NSGraphicsMBS, Rect as NSRectMBS) as Boolean	650
* 9.33.15 EnableMenuItems	651
* 9.33.16 FrameChanged	651
* 9.33.17 GotFocus	651
* 9.33.18 LostFocus	651
* 9.33.19MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	652
* 9.33.20 MouseDrag(x as Integer, y as Integer)	652
* 9.33.21 MouseUp(x As Integer, y As Integer)	652
* 9.33.22 Open	652
* 9.33.23 resizeSubviewsWithOldSize(oldSize as NSSizeMBS)	653
* 9.33.24 ScaleFactorChanged(NewFactor as double)	653
* 9.33.25 shouldAdjustSizeOfSubview(view as NSViewMBS) as Boolean	653
* 9.33.26 shouldHideDivider(dividerIndex as Integer) as Boolean	654
* 9.33.27 splitViewDidResizeSubviews	654
* 9.33.28 splitViewWillResizeSubviews	654
* 9.33.29 willShowContextualMenu(menu as NSMenuItemMBS, NSEvent as NSEventMBS)	654
– 9.34.1 class NSSplitViewMBS	655
* 9.34.3 addArrangedSubview(view as NSViewMBS)	655
* 9.34.4 adjustSubviews	656
* 9.34.5 Constructor	656
* 9.34.6 Constructor(Handle as Integer)	656
* 9.34.7 Constructor(left as double, top as double, width as double, height as double)	657
* 9.34.8 insertArrangedSubview(view as NSViewMBS, index as Integer)	657
* 9.34.9 isSubviewCollapsed(view as NSViewMBS) as Boolean	657
* 9.34.10 maxPossiblePositionOfDividerAtIndex(dividerIndex as Integer) as Double	658

* 9.34.11 minPossiblePositionOfDividerAtIndex(dividerIndex as Integer) as Double	658
* 9.34.12 removeArrangedSubview(view as NSViewMBS)	659
* 9.34.13 setPosition(position as double, dividerIndex as Integer)	659
* 9.34.15 arrangesAllSubviews as Boolean	659
* 9.34.16 autosaveName as String	660
* 9.34.17 dividerColor as NSColorMBS	660
* 9.34.18 dividerStyle as Integer	660
* 9.34.19 dividerThickness as Double	660
* 9.34.20 Vertical as Boolean	661
* 9.34.21 holdingPriorityForSubview(subviewIndex as Integer) as Integer	661

• 25 Statusitem	1153
– 25.1.1 class NSStatusItemMBS	1153
* 25.1.3 Available as boolean	1154
* 25.1.4 Close	1154
* 25.1.5 CreateMenu as boolean	1154
* 25.1.6 CreateMenu(length as single) as boolean	1154
* 25.1.7 CreateMenuMiddle(length as single) as boolean	1155
* 25.1.8 CreateMenuRight(length as single) as boolean	1156
* 25.1.9 DrawStatusBarBackground(x as Double, y as Double, width as Double, height as Double, highlight as boolean)	1157
* 25.1.10 MenuIsVertical as boolean	1157
* 25.1.11 MenuThickness as Double	1157
* 25.1.12 popUpStatusItemMenu(menu as NSMenuMBS)	1157
* 25.1.13 SendActionOn(mode as Integer)	1158
* 25.1.15 alternateImage as NSImageMBS	1158
* 25.1.16 attributedTitle as NSAttributedStringMBS	1158
* 25.1.17 Button as Variant	1158
* 25.1.18 Enabled as boolean	1159
* 25.1.19 Handle as Integer	1159
* 25.1.20 Height as single	1159
* 25.1.21 HighlightMode as boolean	1159
* 25.1.22 image as NSImageMBS	1160
* 25.1.23 Left as single	1160
* 25.1.24 Length as single	1160
* 25.1.25 Menu as NSMenuMBS	1160
* 25.1.26 Title as String	1161
* 25.1.27 ToolTip as String	1161
* 25.1.28 Top as single	1161
* 25.1.29 View as NSViewMBS	1161
* 25.1.30 Width as single	1161
* 25.1.31 Window as NSWindowMBS	1162
* 25.1.33 Action	1162
* 25.1.34 DoubleAction	1162

• 9 Cocoa Controls	453
– 9.35.1 class NSTabViewItemMBS	663
* 9.35.3 Constructor(identifier as Variant)	663
* 9.35.5 color as NSColorMBS	663
* 9.35.6 Enabled as Boolean	663
* 9.35.7 Handle as Integer	664
* 9.35.8 identifier as Variant	664
* 9.35.9 image as NSImageMBS	664
* 9.35.10 initialFirstResponder as NSViewMBS	664
* 9.35.11 label as string	664
* 9.35.12 tabState as Integer	665
* 9.35.13 tableView as NSTabViewMBS	665
* 9.35.14 toolTip as string	665
* 9.35.15 view as NSViewMBS	665
– 9.36.1 class NSTabViewMBS	667
* 9.36.3 addTabViewItem(tableViewItem as NSTabViewItemMBS)	667
* 9.36.4 Constructor	667
* 9.36.5 Constructor(Handle as Integer)	668
* 9.36.6 Constructor(left as Double, top as Double, width as Double, height as Double)	668
* 9.36.7 contentRect as NSRectMBS	668
* 9.36.8 indexOfTabViewItem(tableViewItem as NSTabViewItemMBS) as Integer	669
* 9.36.9 indexOfTabViewItemWithIdentifier(identifier as Variant) as Integer	669
* 9.36.10 insertTabViewItem(tableViewItem as NSTabViewItemMBS, atIndex as Integer)	669
* 9.36.11 minimumSize as NSSizeMBS	669
* 9.36.12 numberOfTabViewItems as Integer	669
* 9.36.13 removeTabViewItem(tableViewItem as NSTabViewItemMBS)	669
* 9.36.14 selectedTabViewItem as NSTabViewItemMBS	670
* 9.36.15 selectFirstTabViewItem	670
* 9.36.16 selectLastTabViewItem	670
* 9.36.17 selectNextTabViewItem	670
* 9.36.18 selectPreviousTabViewItem	670
* 9.36.19 selectTabViewItem(tableViewItem as NSTabViewItemMBS)	671
* 9.36.20 selectTabViewItemAtIndex(index as Integer)	671
* 9.36.21 selectTabViewItemWithIdentifier(identifier as Variant)	671
* 9.36.22 tabViewItemAtIndex(index as Integer) as NSTabViewItemMBS	671
* 9.36.23 tabViewItemAtPoint(x as Double, y as Double) as NSTabViewItemMBS	671
* 9.36.24 tabViewItems as NSTabViewItemMBS()	672
* 9.36.26 allowsTruncatedLabels as boolean	672
* 9.36.27 controlSize as Integer	672
* 9.36.28 controlTint as Integer	672
* 9.36.29 drawsBackground as boolean	672
* 9.36.30 font as NSFontMBS	673
* 9.36.31 tableViewType as Integer	673

	99
• 14 Cocoa Tasks	859
– 14.3.1 class NSTaskMBS	878
* 14.3.3 arguments as string()	879
* 14.3.4 Constructor	879
* 14.3.5 Destructor	879
* 14.3.6 interrupt	879
* 14.3.7 launch	879
* 14.3.8 launchedTaskWithLaunchPath(path as string, arguments() as string) as NSTaskMBS	880
* 14.3.9 NSTaskDidTerminateNotification as string	881
* 14.3.10 resume as boolean	881
* 14.3.11 setArguments(arguments() as string)	881
* 14.3.12 setStandardError(p as NSFileHandleMBS)	882
* 14.3.13 setStandardError(p as NSPipeMBS)	882
* 14.3.14 setStandardInput(p as NSFileHandleMBS)	882
* 14.3.15 setStandardInput(p as NSPipeMBS)	883
* 14.3.16 setStandardOutput(p as NSFileHandleMBS)	883
* 14.3.17 setStandardOutput(p as NSPipeMBS)	884
* 14.3.18 standardError as Variant	884
* 14.3.19 standardInput as Variant	884
* 14.3.20 standardOutput as Variant	884
* 14.3.21 suspend as boolean	885
* 14.3.22 terminate	885
* 14.3.23 waitUntilExit	885
* 14.3.25 currentDirectoryPath as string	886
* 14.3.26 Handle as Integer	886
* 14.3.27 isRunning as boolean	886
* 14.3.28 launchPath as string	886
* 14.3.29 processIdentifier as Integer	886
* 14.3.30 qualityOfService as Integer	887
* 14.3.31 terminationReason as Integer	887
* 14.3.32 terminationStatus as Integer	888
* 14.3.33 environment as dictionary	888
* 14.3.35 Terminated	889

• 13 Cocoa Regular Expressions	831
– 13.3.1 class NSTextCheckingResultMBS	844
* 13.3.3 addressCheckingResult(Range as NSRangeMBS, components as Dictionary) as NSTextCheckingResultMBS	845
* 13.3.4 alternativeStrings as String()	845
* 13.3.5 Constructor	845
* 13.3.6 copy as NSTextCheckingResultMBS	845
* 13.3.7 correctionCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS	845
* 13.3.8 correctionCheckingResult(Range as NSRangeMBS, replacementString as String, alternativeStrings() as String) as NSTextCheckingResultMBS	846
* 13.3.9 dashCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS	846
* 13.3.10 dateCheckingResult(Range as NSRangeMBS, date as Date) as NSTextCheckingResultMBS	846
* 13.3.11 dateCheckingResult(Range as NSRangeMBS, date as Date, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS	847
* 13.3.12 dateCheckingResult(Range as NSRangeMBS, date as DateTime) as NSTextCheckingResultMBS	847
* 13.3.13 dateCheckingResult(Range as NSRangeMBS, date as DateTime, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS	848
* 13.3.14 grammarCheckingResult(Range as NSRangeMBS, details() as Dictionary) as NSTextCheckingResultMBS	848
* 13.3.15 grammarDetails as Dictionary()	848
* 13.3.16 linkCheckingResult(Range as NSRangeMBS, URL as String) as NSTextCheckingResultMBS	849
* 13.3.17 NSTextCheckingAirlineKey as String	849
* 13.3.18 NSTextCheckingCityKey as String	849
* 13.3.19 NSTextCheckingCountryKey as String	849
* 13.3.20 NSTextCheckingFlightKey as String	849
* 13.3.21 NSTextCheckingJobTitleKey as String	850
* 13.3.22 NSTextCheckingNameKey as String	850
* 13.3.23 NSTextCheckingOrganizationKey as String	850
* 13.3.24 NSTextCheckingPhoneKey as String	850
* 13.3.25 NSTextCheckingStateKey as String	851
* 13.3.26 NSTextCheckingStreetKey as String	851
* 13.3.27 NSTextCheckingZIPKey as String	851
* 13.3.28 orthographyCheckingResult(Range as NSRangeMBS, orthography as NSOrthographyMBS) as NSTextCheckingResultMBS	851
* 13.3.29 phoneNumberCheckingResult(Range as NSRangeMBS, phoneNumber as String) as NSTextCheckingResultMBS	852
* 13.3.30 quoteCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS	852

	101
* 13.3.31 rangeAtIndex(index as Integer) as NSRangeMBS	852
* 13.3.32 rangeWithName(name as string) as NSRangeMBS	852
* 13.3.33 replacementCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS	853
* 13.3.34 resultByAdjustingRangesWithOffset(offset as Integer) as NSTextCheckingResultMBS	853
* 13.3.35 spellCheckingResult(Range as NSRangeMBS) as NSTextCheckingResultMBS	853
* 13.3.36 transitInformationCheckingResult(Range as NSRangeMBS, components as Dictionary) as NSTextCheckingResultMBS	853
* 13.3.38 addressComponents as Dictionary	854
* 13.3.39 components as Dictionary	854
* 13.3.40 date as Date	855
* 13.3.41 dateTime as DateTime	855
* 13.3.42 duration as Double	855
* 13.3.43 Handle as Integer	855
* 13.3.44 numberOfRanges as Integer	856
* 13.3.45 orthography as NSOrthographyMBS	856
* 13.3.46 phoneNumber as String	856
* 13.3.47 range as NSRangeMBS	857
* 13.3.48 regularExpression as NSRegularExpressionMBS	857
* 13.3.49 replacementString as String	857
* 13.3.50 resultType as Integer	857
* 13.3.51 timeZone as NSTimeZoneMBS	857
* 13.3.52 URL as String	858

• 15 Cocoa Text	891
– 15.3.1 class NSTextContainerMBS	908
* 15.3.3 Constructor	908
* 15.3.4 Constructor(size as NSSizeMBS)	908
* 15.3.5 containsPoint(p as NSPointMBS) as boolean	909
* 15.3.6 isSimpleRectangularTextContainer as boolean	909
* 15.3.7 replaceLayoutManager(l as NSLayoutManagerMBS)	909
* 15.3.9 Handle as Integer	910
* 15.3.10 containerSize as NSSizeMBS	910
* 15.3.11 heightTracksTextView as boolean	910
* 15.3.12 layoutManager as NSLayoutManagerMBS	910
* 15.3.13 lineFragmentPadding as Double	910
* 15.3.14 textView as NSTextViewMBS	911
* 15.3.15 widthTracksTextView as boolean	911

	103
• 9 Cocoa Controls	453
– 9.37.1 class NSTextFieldCellMBS	675
* 9.37.3 allowedInputSourceLocales as string()	676
* 9.37.4 Constructor(text as string)	676
* 9.37.5 setAllowedInputSourceLocales(Identifiers() as string)	676
* 9.37.6 setUpFieldEditorAttributes(textobj as NSTextMBS) as NSTextMBS	676
* 9.37.7 setWantsNotificationForMarkedText(value as boolean)	677
* 9.37.9 backgroundColor as NSColorMBS	677
* 9.37.10 bezelStyle as Integer	677
* 9.37.11 drawsBackground as boolean	677
* 9.37.12 placeholderAttributedString as NSAttributedStringMBS	677
* 9.37.13 placeholderString as string	678
* 9.37.14 textColor as NSColorMBS	678
– 9.38.1 control NSTextFieldControlMBS	679
* 9.38.3 View as NSTextFieldMBS	679
* 9.38.5 Action	679
* 9.38.6 BoundsChanged	679
* 9.38.7 Close	680
* 9.38.8 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean	680
* 9.38.9 ContextualMenuAction(hitItem as MenuItem) as Boolean	680
* 9.38.10 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	680
* 9.38.11 EnableMenuItems	680
* 9.38.12 FrameChanged	681
* 9.38.13 GotFocus	681
* 9.38.14 LostFocus	681
* 9.38.15MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	681
* 9.38.16 MouseDrag(x as Integer, y as Integer)	682
* 9.38.17 MouseUp(x as Integer, y as Integer)	682
* 9.38.18 Open	682
* 9.38.19 ScaleFactorChanged(NewFactor as Double)	682
* 9.38.20 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	683
* 9.38.21 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	683
* 9.38.22 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)	683
* 9.38.23 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean	684
* 9.38.24 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean	684
* 9.38.25 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)	684
– 9.39.1 class NSTextFieldMBS	685
* 9.39.3 Constructor	685

* 9.39.4	Constructor(Handle as Integer)	685
* 9.39.5	Constructor(left as Double, top as Double, width as Double, height as Double)	686
* 9.39.6	selectText	686
* 9.39.8	AllowsCharacterPickerTouchBarItem as Boolean	687
* 9.39.9	allowsEditingTextAttributes as boolean	687
* 9.39.10	AutomaticTextCompletionEnabled as Boolean	687
* 9.39.11	backgroundColor as NSColorMBS	687
* 9.39.12	Bezeled as boolean	687
* 9.39.13	bezelStyle as Integer	688
* 9.39.14	Bordered as boolean	688
* 9.39.15	drawsBackground as boolean	688
* 9.39.16	Editable as boolean	688
* 9.39.17	importsGraphics as boolean	689
* 9.39.18	placeholderAttributedString as NSAttributedStringMBS	689
* 9.39.19	placeholderString as String	689
* 9.39.20	Selectable as boolean	689
* 9.39.21	textColor as NSColorMBS	689
– 9.40.1	class NSTextFinderMBS	691
* 9.40.3	cancelFindIndicator	691
* 9.40.4	Constructor	692
* 9.40.5	performAction(operation as Integer)	692
* 9.40.6	validateAction(operation as Integer) as Boolean	692
* 9.40.8	client as NSViewMBS	692
* 9.40.9	findBarController as NSViewMBS	693
* 9.40.10	Handle as Integer	693
* 9.40.11	incrementalSearchingEnabled as Boolean	693
* 9.40.12	incrementalSearchingShouldDimContentView as Boolean	693

	105
• 15 Cocoa Text	891
– 15.4.1 class NSTextMBS	913
* 15.4.3 alignCenter	913
* 15.4.4 alignLeft	913
* 15.4.5 alignRight	913
* 15.4.6 changeFont	914
* 15.4.7 checkSpelling	914
* 15.4.8 Constructor	914
* 15.4.9 Constructor(Handle as Integer)	914
* 15.4.10 Constructor(left as Double, top as Double, width as Double, height as Double)	915
* 15.4.11 copy	915
* 15.4.12 copyFont	915
* 15.4.13 copyRuler	916
* 15.4.14 cut	916
* 15.4.15 delete	916
* 15.4.16 isRulerVisible as boolean	916
* 15.4.17 maxSizeHeight as Double	916
* 15.4.18 maxSizeWidth as Double	916
* 15.4.19 minSizeHeight as Double	917
* 15.4.20 minSizeWidth as Double	917
* 15.4.21 NSCenterTextAlignment as Integer	917
* 15.4.22 NSRightTextAlignment as Integer	917
* 15.4.23 paste	917
* 15.4.24 pasteFont	917
* 15.4.25 pasteRuler	918
* 15.4.26 readRTFDFromFile(file as folderitem) as boolean	918
* 15.4.27 replaceCharactersInRangeWithRTF(start as Integer, length as Integer, rtfData as MemoryBlock)	918
* 15.4.28 replaceCharactersInRangeWithRTFD(start as Integer, length as Integer, rtfData as MemoryBlock)	918
* 15.4.29 replaceCharactersInRangeWithString(start as Integer, length as Integer, text as string)	919
* 15.4.30 RTFDFromRange(start as Integer, length as Integer) as MemoryBlock	919
* 15.4.31 RTFFromRange(start as Integer, length as Integer) as MemoryBlock	919
* 15.4.32 scrollRangeToVisible(start as Integer, length as Integer)	920
* 15.4.33 selectAll	920
* 15.4.34 setFontForRange(font as NSFontMBS, start as Integer, length as Integer)	920
* 15.4.35 setMaxSize(width as Double, height as Double)	920
* 15.4.36 setMinSize(width as Double, height as Double)	921
* 15.4.37 setTextColorForRange(colorValue as NSColorMBS, start as Integer, length as Integer)	921
* 15.4.38 showGuessPanel	921

* 15.4.39	sizeToFit	921
* 15.4.40	subscript	921
* 15.4.41	superscript	922
* 15.4.42	textLength as Integer	922
* 15.4.43	toggleRuler	922
* 15.4.44	underline	922
* 15.4.45	unscript	922
* 15.4.46	writeRTFDToFile(file as folderitem, atomically as boolean) as boolean	923
* 15.4.48	alignment as Integer	923
* 15.4.49	backgroundColor as NSColorMBS	923
* 15.4.50	baseWritingDirection as Integer	923
* 15.4.51	drawsBackground as boolean	923
* 15.4.52	Enabled as boolean	924
* 15.4.53	font as NSFontMBS	924
* 15.4.54	importsGraphics as boolean	924
* 15.4.55	isEditable as boolean	924
* 15.4.56	isFieldEditor as boolean	924
* 15.4.57	isHorizontallyResizable as boolean	925
* 15.4.58	isRichText as boolean	925
* 15.4.59	isSelectable as boolean	925
* 15.4.60	isVerticallyResizable as boolean	925
* 15.4.61	selectedRange as NSRangeMBS	926
* 15.4.62	text as string	926
* 15.4.63	textColor as NSColorMBS	926
* 15.4.64	usesFontPanel as boolean	926
* 15.4.66	textDidBeginEditing	927
* 15.4.67	textDidChange	927
* 15.4.68	textDidEndEditing	927
* 15.4.69	textShouldBeginEditing as boolean	927
* 15.4.70	textShouldEndEditing as boolean	927
– 15.5.1	class NSTextStorageMBS	931
* 15.5.3	addLayoutManager(1 as NSLayoutManagerMBS)	931
* 15.5.4	changeInLength as Integer	931
* 15.5.5	Constructor	932
* 15.5.6	editedMask as Integer	932
* 15.5.7	editedRange as NSRangeMBS	932
* 15.5.8	ensureAttributesAreFixedInRange(Range as NSRangeMBS)	932
* 15.5.9	fixesAttributesLazily as boolean	933
* 15.5.10	invalidateAttributesInRange(Range as NSRangeMBS)	933
* 15.5.11	processEditing	933
* 15.5.12	removeLayoutManager(1 as NSLayoutManagerMBS)	933

	107
– 15.6.1 class NSTextTabMBS	935
* 15.6.3 Constructor	935
* 15.6.4 Constructor(alignment as Integer, location as Double, options as dictionary)	935
* 15.6.5 Constructor(type as Integer, location as Double)	936
* 15.6.6 copy as NSTextTabMBS	936
* 15.6.8 alignment as Integer	936
* 15.6.9 Handle as Integer	936
* 15.6.10 location as Double	936
* 15.6.11 options as Dictionary	937
* 15.6.12 tabStopType as Integer	937

• 9 Cocoa Controls	453
– 9.41.1 control NSTextViewControlMBS	695
* 9.41.3 AcceptTabs as Boolean	695
* 9.41.4 Scrollview as Variant	695
* 9.41.5 View as NSTextViewMBS	695
* 9.41.7 BoundsChanged	696
* 9.41.8 Close	696
* 9.41.9 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean	696
* 9.41.10 ContextualMenuItemAction(hitItem as MenuItem) as Boolean	696
* 9.41.11 didCloseContextualMenu(menu as NSMenuItemMBS, NSEvent as NSEventMBS)	697
* 9.41.12 EnableMenuItems	697
* 9.41.13 FrameChanged	697
* 9.41.14 GotFocus	697
* 9.41.15 LostFocus	697
* 9.41.16MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean	698
* 9.41.17 MouseDrag(x as Integer, y as Integer)	698
* 9.41.18 MouseUp(x as Integer, y as Integer)	698
* 9.41.19 Open	698
* 9.41.20 ScaleFactorChanged(NewFactor as Double)	699
* 9.41.21 shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string) as boolean	699
* 9.41.22 textDidBeginEditing	699
* 9.41.23 textDidChange	699
* 9.41.24 textDidEndEditing	699
* 9.41.25 textShouldBeginEditing as boolean	700
* 9.41.26 textShouldEndEditing as boolean	700
* 9.41.27 textViewDidChangeSelection	700
* 9.41.28 willShowContextualMenu(menu as NSMenuItemMBS, NSEvent as NSEventMBS)	700
– 9.42.1 class NSTextViewMBS	701
* 9.42.3 alignJustified	701
* 9.42.4 breakUndoCoalescing	702
* 9.42.5 changeAttributes	702
* 9.42.6 changeColor	702
* 9.42.7 changeDocumentBackgroundColor	702
* 9.42.8 checkTextInDocument	702
* 9.42.9 checkTextInSelection	703
* 9.42.10 complete	703
* 9.42.11 Constructor	703
* 9.42.12 Constructor(Handle as Integer)	703
* 9.42.13 Constructor(left as Double, top as Double, width as Double, height as Double)	704

	109
* 9.42.14 didChangeText	704
* 9.42.15 insertText(attributedString as NSAttributedStringMBS)	704
* 9.42.16 insertText(text as string)	705
* 9.42.17 invalidateTextContainerOrigin	706
* 9.42.18 loosenKerning	706
* 9.42.19 lowerBaseline	706
* 9.42.20 orderFrontLinkPanel	707
* 9.42.21 orderFrontListPanel	707
* 9.42.22 orderFrontSpacingPanel	707
* 9.42.23 orderFrontSubstitutionsPanel	707
* 9.42.24 orderFrontTablePanel	707
* 9.42.25 outline	707
* 9.42.26 pasteAsPlainText	708
* 9.42.27 pasteAsRichText	708
* 9.42.28 performFindPanelAction(FindAction as Integer)	708
* 9.42.29 performFindPanelAction(sender as object)	709
* 9.42.30 raiseBaseline	709
* 9.42.31 replaceTextContainer(textContainer as NSTextContainerMBS)	709
* 9.42.32 shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string = "") as Boolean	710
* 9.42.33 showFindIndicatorForRange(charRange as NSRangeMBS)	710
* 9.42.34 startSpeaking	711
* 9.42.35 stopSpeaking	711
* 9.42.36 tightenKerning	711
* 9.42.37 toggleAutomaticDashSubstitution	711
* 9.42.38 toggleAutomaticDataDetection	711
* 9.42.39 toggleAutomaticLinkDetection	712
* 9.42.40 toggleAutomaticQuoteSubstitution	712
* 9.42.41 toggleAutomaticSpellingCorrection	712
* 9.42.42 toggleAutomaticTextReplacement	712
* 9.42.43 toggleBold	712
* 9.42.44 toggleContinuousSpellChecking	713
* 9.42.45 toggleGrammarChecking	713
* 9.42.46 toggleItalic	713
* 9.42.47 toggleSmartInsertDelete	714
* 9.42.48 toggleTraditionalCharacterShape	714
* 9.42.49 turnOffKerning	714
* 9.42.50 turnOffLigatures	714
* 9.42.51 updateDragTypeRegistration	714
* 9.42.52 updateFontPanel	715
* 9.42.53 updateRuler	715
* 9.42.54 useAllLigatures	715

* 9.42.55 useStandardKerning	715
* 9.42.56 useStandardLigatures	715
* 9.42.58 acceptsGlyphInfo as boolean	716
* 9.42.59 allowsDocumentBackgroundColorChange as boolean	716
* 9.42.60 allowsImageEditing as boolean	716
* 9.42.61 allowsUndo as boolean	716
* 9.42.62 AutomaticDashSubstitutionEnabled as boolean	717
* 9.42.63 AutomaticDataDetectionEnabled as boolean	717
* 9.42.64 AutomaticLinkDetectionEnabled as boolean	717
* 9.42.65 AutomaticQuoteSubstitutionEnabled as boolean	718
* 9.42.66 AutomaticSpellingCorrectionEnabled as boolean	718
* 9.42.67 AutomaticTextReplacementEnabled as boolean	718
* 9.42.68 backgroundColor as NSColorMBS	719
* 9.42.69 Bold as Boolean	719
* 9.42.70 ContinuousSpellCheckingEnabled as boolean	719
* 9.42.71 defaultParagraphStyle as Variant	719
* 9.42.72 displaysLinkToolTips as boolean	720
* 9.42.73 enabledTextCheckingTypes as Int64	720
* 9.42.74 GrammarCheckingEnabled as boolean	721
* 9.42.75 insertionPointColor as NSColorMBS	721
* 9.42.76 isCoalescingUndo as boolean	721
* 9.42.77 Italic as Boolean	721
* 9.42.78 layoutManager as NSLayoutManagerMBS	722
* 9.42.79 linkTextAttributes as dictionary	722
* 9.42.80 markedTextAttributes as dictionary	722
* 9.42.81 RTFData as Memoryblock	723
* 9.42.82 RulerVisible as boolean	723
* 9.42.83 selectedTextAttributes as dictionary	723
* 9.42.84 smartInsertDeleteEnabled as boolean	724
* 9.42.85 spellCheckerDocumentTag as Integer	724
* 9.42.86 textContainer as NSTextContainerMBS	724
* 9.42.87 textContainerInset as NSSizeMBS	724
* 9.42.88 textContainerOrigin as NSPointMBS	725
* 9.42.89 textStorage as NSTextStorageMBS	725
* 9.42.90 typingAttributes as dictionary	726
* 9.42.91 usesFindBar as Boolean	726
* 9.42.92 usesFindPanel as boolean	726
* 9.42.93 usesFontPanel as boolean	726
* 9.42.94 usesInspectorBar as Boolean	726
* 9.42.95 usesRuler as boolean	727
* 9.42.97 shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string) as boolean	727
* 9.42.98 textViewDidChangeSelection	728

	111
• 8 Cocoa	271
– 8.11.1 class NSTimerMBS	381
* 8.11.3 Constructor(fireDate as date, timeInterval as Double, repeats as boolean)	382
* 8.11.4 Constructor(fireDate as date, timeInterval as Double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string)	383
* 8.11.5 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean)	383
* 8.11.6 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string)	384
* 8.11.7 Constructor(timeInterval as Double, repeats as boolean)	385
* 8.11.8 fire	385
* 8.11.9 invalidate	385
* 8.11.10 Timer(t as timer) as NSTimerMBS	386
* 8.11.12 fireDate as date	386
* 8.11.13 fireDateTime as DateTime	387
* 8.11.14 Handle as Integer	387
* 8.11.15 isValid as boolean	388
* 8.11.16 tag as Variant	388
* 8.11.17 timeInterval as Double	388
* 8.11.18 tolerance as Double	389
* 8.11.20 Action	389

- **11 Cocoa Networking** 775
 - 11.1.1 class `NSURLRequestCertificateFilterMBS` 775
 - * 11.1.3 `allowsAnyHTTPSCertificateForHost(host as string) as boolean` 775

	113
• 23 Speech	1097
– 23.3.1 class NSVoiceMBS	1123
* 23.3.3 Age as Integer	1123
* 23.3.4 Constructor	1123
* 23.3.5 Demotext as String	1123
* 23.3.6 Gender as String	1123
* 23.3.7 GenderFemale as String	1124
* 23.3.8 GenderMale as String	1124
* 23.3.9 GenderNeuter as String	1124
* 23.3.10 Identifier as String	1124
* 23.3.11 Language as String	1125
* 23.3.12 LocaleIdentifier as String	1125
* 23.3.13 Name as String	1125
* 23.3.14 NSVoiceAge as String	1126
* 23.3.15 NSVoiceDemoText as String	1126
* 23.3.16 NSVoiceGender as String	1126
* 23.3.17 NSVoiceIdentifier as String	1126
* 23.3.18 NSVoiceIndividuallySpokenCharacters as String	1127
* 23.3.19 NSVoiceLanguage as String	1127
* 23.3.20 NSVoiceLocaleIdentifier as String	1127
* 23.3.21 NSVoiceName as String	1128
* 23.3.22 NSVoiceSupportedCharacters as String	1128
* 23.3.23 Properties as Dictionary	1128

• 8 Cocoa	271
– 8.12.1 class <code>NSWindowDelegateMBS</code>	390
* 8.12.3 <code>Constructor(win as DesktopWindow)</code>	390
* 8.12.4 <code>Constructor(win as NSWindowMBS)</code>	390
* 8.12.5 <code>Constructor(win as window)</code>	391
* 8.12.6 <code>InstallRestoreEvents</code>	391
* 8.12.8 <code>concludeDragOperation(sender as NSDraggingInfoMBS)</code>	391
* 8.12.9 <code>customWindowsToEnterFullScreenForWindow(win as NSWindowMBS) as NSWindowMBS()</code>	392
* 8.12.10 <code>customWindowsToExitFullScreenForWindow(win as NSWindowMBS) as NSWindowMBS()</code>	392
* 8.12.11 <code>didDecodeRestorableState(win as NSWindowMBS, state as NSCoderMBS)</code>	393
* 8.12.12 <code>draggingEnded(sender as NSDraggingInfoMBS)</code>	393
* 8.12.13 <code>draggingEntered(sender as NSDraggingInfoMBS) as Integer</code>	393
* 8.12.14 <code>draggingExited(sender as NSDraggingInfoMBS)</code>	394
* 8.12.15 <code>draggingUpdated(sender as NSDraggingInfoMBS) as Integer</code>	394
* 8.12.16 <code>encodeRestorableStateWithCoder(win as NSWindowMBS, coder as NSCoderMBS)</code>	395
* 8.12.17 <code>performDragOperation(sender as NSDraggingInfoMBS) as boolean</code>	395
* 8.12.18 <code>prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean</code>	395
* 8.12.19 <code>restoreStateWithCoder(win as NSWindowMBS, coder as NSCoderMBS)</code>	396
* 8.12.20 <code>shouldDragDocumentWithEvent(win as NSWindowMBS, evt as NSEventMBS, dragImageLocation as NSPointMBS, pasteboard as Variant) as boolean</code>	396
* 8.12.21 <code>shouldPopUpDocumentPathMenu(win as NSWindowMBS, menu as NSMenuMBS) as boolean</code>	397
* 8.12.22 <code>startCustomAnimationToEnterFullScreenWithDuration(win as NSWindowMBS, duration as Double)</code>	397
* 8.12.23 <code>startCustomAnimationToExitFullScreenWithDuration(win as NSWindowMBS, duration as Double)</code>	397
* 8.12.24 <code>updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)</code>	398
* 8.12.25 <code>wantsPeriodicDraggingUpdates as boolean</code>	398
* 8.12.26 <code>willEncodeRestorableState(win as NSWindowMBS, state as NSCoderMBS)</code>	398
* 8.12.27 <code>willPositionSheet(win as NSWindowMBS, sheet as NSWindowMBS, rect as NSRectMBS) as NSRectMBS</code>	399
* 8.12.28 <code>willResizeForVersionBrowser(win as NSWindowMBS, maxPreferredFrameSize as NSSizeMBS, maxAllowedFrameSize as NSSizeMBS) as NSSizeMBS</code>	399
* 8.12.29 <code>willUseFullScreenContentSize(win as NSWindowMBS, proposedSize as NSSizeMBS) as NSSizeMBS</code>	400
* 8.12.30 <code>willUseFullScreenPresentationOptions(win as NSWindowMBS, proposedOptions as Integer) as Integer</code>	400
* 8.12.31 <code>windowDidBecomeKey(notification as NSNotificationMBS)</code>	401
* 8.12.32 <code>windowDidBecomeMain(notification as NSNotificationMBS)</code>	401
* 8.12.33 <code>windowDidChangeScreen(notification as NSNotificationMBS)</code>	401

	115
* 8.12.34 windowDidChangeScreenProfile(notification as NSNotificationMBS)	402
* 8.12.35 windowDidDeminiaturize(notification as NSNotificationMBS)	402
* 8.12.36 windowDidEndLiveResize(notification as NSNotificationMBS)	402
* 8.12.37 windowDidEndSheet(notification as NSNotificationMBS)	402
* 8.12.38 windowDidEnterFullScreen(notification as NSNotificationMBS)	403
* 8.12.39 windowDidEnterVersionBrowser(notification as NSNotificationMBS)	403
* 8.12.40 windowDidExitFullScreen(notification as NSNotificationMBS)	403
* 8.12.41 windowDidExitVersionBrowser(notification as NSNotificationMBS)	403
* 8.12.42 windowDidExpose(notification as NSNotificationMBS)	403
* 8.12.43 windowDidFailToEnterFullScreen(win as NSWindowMBS)	404
* 8.12.44 windowDidFailToExitFullScreen(win as NSWindowMBS)	404
* 8.12.45 windowDidMiniaturize(notification as NSNotificationMBS)	405
* 8.12.46 windowDidMove(notification as NSNotificationMBS)	405
* 8.12.47 windowDidResignKey(notification as NSNotificationMBS)	405
* 8.12.48 windowDidResignMain(notification as NSNotificationMBS)	405
* 8.12.49 windowDidResize(notification as NSNotificationMBS)	406
* 8.12.50 windowDidUpdate(notification as NSNotificationMBS)	406
* 8.12.51 windowShouldClose as boolean	406
* 8.12.52 windowShouldZoom(win as NSWindowMBS, newFrame as NSRectMBS) as boolean	406
* 8.12.53 windowWillBeginSheet(notification as NSNotificationMBS)	407
* 8.12.54 windowWillClose(notification as NSNotificationMBS)	407
* 8.12.55 windowWillEnterFullScreen(notification as NSNotificationMBS)	407
* 8.12.56 windowWillEnterVersionBrowser(notification as NSNotificationMBS)	407
* 8.12.57 windowWillExitFullScreen(notification as NSNotificationMBS)	408
* 8.12.58 windowWillExitVersionBrowser(notification as NSNotificationMBS)	408
* 8.12.59 windowWillMiniaturize(notification as NSNotificationMBS)	408
* 8.12.60 windowWillMove(notification as NSNotificationMBS)	408
* 8.12.61 windowWillResize(win as NSWindowMBS, newFrameSize as NSSizeMBS, newSize as NSSizeMBS) as NSSizeMBS	409
* 8.12.62 windowWillReturnUndoManager(win as NSWindowMBS) as NSUndoManagerMBS	409
* 8.12.63 windowWillStartLiveResize(notification as NSNotificationMBS)	410
* 8.12.64 windowWillUseStandardFrame(win as NSWindowMBS, newFrame as NSRectMBS) as NSRectMBS	410
– 8.13.1 class NSWorkspaceAuthorizationMBS	411
* 8.13.3 Constructor	411
* 8.13.5 Handle as Integer	411
– 8.14.1 class NSWorkspaceMBS	413
* 8.14.3 absolutePathForAppBundleWithIdentifier(bundleIdentifier as string) as string	414
* 8.14.4 activateFileViewerSelectingFiles(Files() as folderitem)	414
* 8.14.5 activateFileViewerSelectingURLs(URLs() as string)	414

* 8.14.6 desktopImageOptionsForScreen(screen as NSScreenMBS) as dictionary	415
* 8.14.7 desktopImageURLForScreen(screen as NSScreenMBS) as folderitem	415
* 8.14.8 fileLabelColors as NSColorMBS()	415
* 8.14.9 fileLabels as string()	416
* 8.14.10 findApplications	417
* 8.14.11 frontmostApplication as NSRunningApplicationMBS	417
* 8.14.12 fullPathForApplication(appname as string) as folderitem	417
* 8.14.13 hideOtherApplications	417
* 8.14.14 iconForFile(file as folderitem) as NSImageMBS	418
* 8.14.15 iconForFiles(files() as folderitem) as NSImageMBS	418
* 8.14.16 iconForFileType filetype as string) as NSImageMBS	419
* 8.14.17 isFilePackageAtPath(item as folderitem) as boolean	420
* 8.14.18 launchApplication(appname as string) as boolean	420
* 8.14.19 launchApplication(appname as string, showicon as boolean, autolaunch as boolean) as boolean	420
* 8.14.20 launchApplicationAtFile(file as folderitem, options as UInt32 = 0, configuration as dictionary = nil) as NSRunningApplicationMBS	421
* 8.14.21 launchApplicationAtFile(file as folderitem, options as UInt32, configuration as dictionary, byref error as NSErrorMBS) as NSRunningApplicationMBS	422
* 8.14.22 launchApplicationAtURL(URL as string, options as UInt32 = 0, configuration as dictionary = nil) as NSRunningApplicationMBS	423
* 8.14.23 launchApplicationAtURL(URL as string, options as UInt32, configuration as dictionary, byref error as NSErrorMBS) as NSRunningApplicationMBS	423
* 8.14.24 launchAppWithBundleIdentifier(bundleIdentifier as string, options as Integer = &h00030000, AppleEventDescriptor as Variant = nil) as Boolean	424
* 8.14.25 localizedDescriptionForType(typeName as string) as string	424
* 8.14.26 menuBarOwningApplication as NSRunningApplicationMBS	424
* 8.14.27 mountedLocalVolumePaths as string()	425
* 8.14.28 mountedRemovableMedia as string()	425
* 8.14.29 noteFileSystemChanged	426
* 8.14.30 noteFileSystemChanged(path as folderitem)	426
* 8.14.31 notificationCenter as NSNotificationCenterMBS	426
* 8.14.32 NSWorkspaceActiveSpaceDidChangeNotification as string	426
* 8.14.33 NSWorkspaceApplicationKey as string	427
* 8.14.34 NSWorkspaceCompressOperation as string	427
* 8.14.35 NSWorkspaceCopyOperation as string	427
* 8.14.36 NSWorkspaceDecompressOperation as string	428
* 8.14.37 NSWorkspaceDecryptOperation as string	428
* 8.14.38 NSWorkspaceDesktopImageAllowClippingKey as string	428
* 8.14.39 NSWorkspaceDesktopImageFillColorKey as string	428
* 8.14.40 NSWorkspaceDesktopImageScalingKey as string	429
* 8.14.41 NSWorkspaceDestroyOperation as string	429
* 8.14.42 NSWorkspaceDidActivateApplicationNotification as string	429

* 8.14.43	NSWorkspaceDidChangeFileLabelsNotification as string	429
* 8.14.44	NSWorkspaceDidDeactivateApplicationNotification as string	429
* 8.14.45	NSWorkspaceDidHideApplicationNotification as string	430
* 8.14.46	NSWorkspaceDidLaunchApplicationNotification as string	430
* 8.14.47	NSWorkspaceDidMountNotification as string	430
* 8.14.48	NSWorkspaceDidPerformFileOperationNotification as string	431
* 8.14.49	NSWorkspaceDidRenameVolumeNotification as string	431
* 8.14.50	NSWorkspaceDidTerminateApplicationNotification as string	431
* 8.14.51	NSWorkspaceDidUnhideApplicationNotification as string	431
* 8.14.52	NSWorkspaceDidUnmountNotification as string	432
* 8.14.53	NSWorkspaceDidWakeNotification as string	432
* 8.14.54	NSWorkspaceDuplicateOperation as string	432
* 8.14.55	NSWorkspaceEncryptOperation as string	432
* 8.14.56	NSWorkspaceLaunchConfigurationAppleEvent as string	432
* 8.14.57	NSWorkspaceLaunchConfigurationArchitecture as string	433
* 8.14.58	NSWorkspaceLaunchConfigurationArguments as string	433
* 8.14.59	NSWorkspaceLaunchConfigurationEnvironment as string	433
* 8.14.60	NSWorkspaceLinkOperation as string	433
* 8.14.61	NSWorkspaceMoveOperation as string	434
* 8.14.62	NSWorkspaceRecycleOperation as string	435
* 8.14.63	NSWorkspaceScreensDidSleepNotification as string	435
* 8.14.64	NSWorkspaceScreensDidWakeNotification as string	436
* 8.14.65	NSWorkspaceSessionDidBecomeActiveNotification as string	436
* 8.14.66	NSWorkspaceSessionDidResignActiveNotification as string	436
* 8.14.67	NSWorkspaceVolumeLocalizedNameKey as string	437
* 8.14.68	NSWorkspaceVolumeOldLocalizedNameKey as string	437
* 8.14.69	NSWorkspaceVolumeOldURLKey as string	437
* 8.14.70	NSWorkspaceVolumeURLKey as string	437
* 8.14.71	NSWorkspaceWillLaunchApplicationNotification as string	438
* 8.14.72	NSWorkspaceWillPowerOffNotification as string	438
* 8.14.73	NSWorkspaceWillSleepNotification as string	438
* 8.14.74	NSWorkspaceWillUnmountNotification as string	439
* 8.14.75	openFile(file as folderitem) as boolean	439
* 8.14.76	openFile(file as folderitem, appname as string) as boolean	439
* 8.14.77	openFile(file as folderitem, appname as string, Deactivate as boolean) as boolean	440
* 8.14.78	openURL(url as string) as boolean	441
* 8.14.79	openURL(url as string, bundleIdentifier as string, options as Integer = &h00030000, AppleEventDescriptor as Variant = nil) as Boolean	441
* 8.14.80	performFileOperation(operation as string, source as folderitem, destination as folderitem, files() as string, byref tag as Integer) as boolean	442
* 8.14.81	preferredFilenameExtensionForType(typeName as string) as string	442
* 8.14.82	requestAuthorization(type as integer, tag as variant = nil)	443

- * 8.14.83 `selectFile(file as folderitem)` as boolean 443
- * 8.14.84 `setDesktopImageURL(file as folderitem, screen as NSScreenMBS, options as dictionary, byref error as NSErrorMBS)` as boolean 443
- * 8.14.85 `setIcon(image as NSImageMBS, file as folderitem, flags as Integer = 0)` as boolean 444
- * 8.14.86 `setIcon(image as NSImageMBS, path as string, flags as Integer = 0)` as boolean 445
- * 8.14.87 `showSearchResultsForQueryString(queryString as string)` as boolean 446
- * 8.14.88 `typeOfFile(File as folderitem, byref error as NSErrorMBS)` as string 446
- * 8.14.89 `typeOfFile(Path as string, byref error as NSErrorMBS)` as string 446
- * 8.14.90 `unmountAndEjectDevice(item as folderitem, byref e as NSErrorMBS)` as boolean 447
- * 8.14.91 `URLForApplicationToOpenURL(url as string)` as string 447
- * 8.14.92 `URLForApplicationWithIdentifier(bundleIdentifier as string)` as string 448
- * 8.14.94 `accessibilityDisplayShouldDifferentiateWithoutColor` as Boolean 448
- * 8.14.95 `accessibilityDisplayShouldIncreaseContrast` as Boolean 448
- * 8.14.96 `accessibilityDisplayShouldInvertColors` as Boolean 448
- * 8.14.97 `accessibilityDisplayShouldReduceMotion` as Boolean 449
- * 8.14.98 `accessibilityDisplayShouldReduceTransparency` as Boolean 449
- * 8.14.99 `isSwitchControlEnabled` as Boolean 449
- * 8.14.100 `isVoiceOverEnabled` as Boolean 449
- * 8.14.102 `requestAuthorizationCompleted(type as Integer, authorization as NSWorkspaceAuthorizationMBS, error as NSErrorMBS, tag as variant)` 450

	119
• 9 Cocoa Controls	453
– 9.43.1 class SearchField	729
* 9.43.3 NSSearchFieldMBS as NSSearchFieldMBS	729
– 9.44.1 class SegmentedControl	730
* 9.44.3 NSSegmentedControlMBS as NSSegmentedControlMBS	730
– 9.45.1 class Statictext	731
* 9.45.3 NSTextFieldMBS as NSTextFieldMBS	731

• 9 Cocoa Controls	453
– 16.6.1 class TabPanel	944
* 16.6.3 NSTabViewMBS as NSTabViewMBS	944
– 9.46.1 class TextArea	732
* 9.46.3 NSTextFieldMBS as NSTextFieldMBS	732
* 9.46.4 NSTextViewMBS as NSTextViewMBS	733
* 9.46.6 RTFDataMBS as Memoryblock	733

	121
• 9 Cocoa Controls	453
– 16.7.1 class TextField	945
* 16.7.3 NSTextFieldMBS as NSTextFieldMBS	945
* 16.7.4 NSTextViewMBS as NSTextViewMBS	945

Chapter 2

List of all classes

• Application	1081
• ConsoleApplication	1082
• Control	939
• CustomNSSearchFieldMBS	453
• CustomNSTextFieldCellMBS	467
• CustomNSTextFieldMBS	470
• CustomNSTextViewMBS	484
• DesktopApplication	1083
• DesktopControl	940
• DesktopLabel	941
• DesktopSearchField	536
• DesktopSegmentedButton	537
• DesktopTabPanel	538
• DesktopTextArea	539
• DesktopTextField	942
• DragItem	947
• InstantMessageMBS	1034
• Label	943
• NSActionCellMBS	541

• NSAlertMBS	271
• NSAppleEventDescriptorMBS	223
• NSAppleEventHandlerMBS	249
• NSAppleEventManagerMBS	251
• NSAppleEventManagerSuspensionIDMBS	255
• NSAppleScriptMBS	256
• NSApplicationDelegateMBS	281
• NSApplicationMBS	294
• NSButtonCellMBS	543
• NSCellMBS	547
• NSColorPanelMBS	735
• NSControlMBS	568
• NSDataDetectorMBS	831
• NSDockTileMBS	329
• NSDraggingImageComponentMBS	948
• NSDraggingInfoMBS	951
• NSDraggingItemMBS	958
• NSDraggingSessionMBS	961
• NSFileHandleMBS	859
• NSFileManagerMBS	965
• NSFontManagerMBS	334
• NSFontPanelMBS	349
• NSGraphicsMBS	744
• NSImageCellMBS	580
• NSLayoutManagerMBS	354
• NSLinguisticTaggerMBS	1039
• NSLinguisticValueMBS	1055
• NSMenuItemCellMBS	584
• NSMutableParagraphStyleMBS	891

	125
• NSOpenPanelMBS	1061
• NSOrthographyMBS	1057
• NSPageLayoutMBS	777
• NSParagraphStyleMBS	899
• NSPathComponentCellMBS	586
• NSPathComponentMBS	594
• NSPipeMBS	876
• NSPopUpButtonCellMBS	600
• NSPrinterMBS	781
• NSPrintInfoMBS	788
• NSPrintOperationMBS	806
• NSPrintPanelMBS	824
• NSRegularExpressionMBS	835
• NSRunLoopMBS	370
• NSRunningApplicationMBS	1084
• NSSavePanelMBS	1069
• NSSavePanelObserverMBS	376
• NSSearchFieldCellMBS	610
• NSSearchFieldMBS	620
• NSSecureTextFieldMBS	631
• NSSegmentedControlMBS	638
• NSServiceProviderMBS	378
• NSSpeechRecognizerMBS	1097
• NSSpeechSynthesizerMBS	1100
• NSSpellCheckerMBS	1129
• NSSplitViewMBS	655
• NSStatusItemMBS	1153
• NSTabViewItemMBS	663
• NSTabViewMBS	667

• NSTaskMBS	878
• NSTextCheckingResultMBS	844
• NSTextContainerMBS	908
• NSTextFieldCellMBS	675
• NSTextFieldMBS	685
• NSTextFieldFinderMBS	691
• NSTextMBS	913
• NSTextStorageMBS	931
• NSTextTabMBS	935
• NSTextViewMBS	701
• NSTimerMBS	381
• NSURLRequestCertificateFilterMBS	775
• NSVoiceMBS	1123
• NSWindowDelegateMBS	390
• NSWorkspaceAuthorizationMBS	411
• NSWorkspaceMBS	413
• SearchField	729
• SegmentedControl	730
• Statictext	731
• TabPanel	944
• TextArea	732
• TextField	945

Chapter 3

List of all controls

• DesktopNSPathControlControlMBS	498
• DesktopNSSearchFieldControlMBS	503
• DesktopNSSecureTextFieldControlMBS	508
• DesktopNSSegmentedControlControlMBS	514
• DesktopNSSplitViewControlMBS	518
• DesktopNSTextFieldControlMBS	526
• DesktopNSTextViewControlMBS	531
• NSPathControlControlMBS	588
• NSSearchFieldControlMBS	614
• NSSecureTextFieldControlMBS	625
• NSSegmentedControlControlMBS	633
• NSSplitViewControlMBS	647
• NSTextFieldControlMBS	679
• NSTextViewControlMBS	695

Chapter 4

List of all global methods

- 5.1.1 InstallNSAccessibilityPatchMBS

131

Chapter 5

Accessibility

5.1 Globals

5.1.1 InstallNSAccessibilityPatchMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Patches crashes with [NSMenuItem accessibilityIsAttributeSettable:] .

Notes: The plugin adds a method [NSMenuItem accessibilityIsAttributeSettable:] if missing to make sure the app doesn't crash when some framework method from Apple calls it.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.5](#)
- [MBS Xojo Plugins, version 18.5pr6](#)
- [Fix for missing accessibilityIsAttributeSettable function](#)

Chapter 6

Addressbook

6.1 class ABAccountMBS

6.1.1 class ABAccountMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNContainerMBS class instead. **Function:** The class for an addressbook account.

Example:

```
dim a as new ABAddressBookMBS
dim accounts() as ABAccountMBS = a.allAccounts

for each c as ABAccountMBS in accounts
  MsgBox c.Name + EndOfLine + c.Identifier + EndOfLine + c.BaseURL
next
```

Notes: The ABAccount functions are not documented by Apple, but work well on OS X version 10.8 to 10.10.

They may work in newer versions if Apple does not change them. They may work in past OS X versions if Apple has the same features there, too.

If the functions are not available in a OS X version, you will see NSEExceptionMBS being raised.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 15.0](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr10](#)

6.1.2 Methods

6.1.3 Constructor

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

6.1.4 Properties

6.1.5 BaseURL as String

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: The base URL for this account.

Notes: (Read only property)

6.1.6 Handle as Integer

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.

Notes: (Read and Write property)

6.1.7 Identifier as String

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: The identifier.

Notes: (Read only property)

6.1.8 isMainAccount as Boolean

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Whether this account is the main account.

Notes: (Read only property)

6.1.9 Name as String

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Name of account.

Example:

```
dim a as new ABAddressBookMBS
dim accounts() as ABAccountMBS = a.allAccounts
```

```
dim c as ABAccountMBS = accounts(0)
MsgBox c.Name
```

Notes: (Read only property)

6.2 class ABAddressBookMBS

6.2.1 class ABAddressBookMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNContactStoreMBS class instead. **Function:** The Addressbook class for Mac OS X 10.2 and newer.

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.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [Multithreaded plugin functions can increase speed of Xojo application](#)
- [Problems with killing Xojo threads with plugin calls.](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr1](#)
- [Using dash if to reduce app size by referencing less plugins](#)
- [Multithreaded plugin functions can increase speed of Real Studio application](#)
- [MBS Real Studio Plugins, version 12.3pr15](#)
- [Addressbook Permission Dialog](#)
- [MBS Real Studio Plugins, version 12.3pr2](#)
- [Addressbook classes updated](#)

6.2.2 Methods

6.2.3 ABAddressBookErrorDomain as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The error domain for Addressbook.

6.2.4 ABMultiValueIdentifiersErrorKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the error keys for the addressbook.

6.2.5 accountWithIdentifier(Identifier as string) as ABAccountMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Finds account with matching identifier.

6.2.6 addRecord(record as ABRecordMBS) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Adds a record (ABPersonMBS or ABGroup) to the AddressBook Database

Example:

```

dim a as new ABAddressBookMBS
dim p as new ABPersonMBS

if not p.setValue("Miller",a.kABLastNameProperty) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABLastNameProperty)
end if

if not p.setValue("Ben",a.kABFirstNameProperty) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABFirstNameProperty)
end if

if a.addRecord(p) then
MsgBox "Record added"
else
MsgBox "Failed to add record"
end if

if a.save then
MsgBox "Changes saved"
else
MsgBox "Failed to save changes"
end if

```

Notes: Returns true if the addition was successful

See also:

- 6.2.7 addRecord(record as ABRecordMBS, Account as ABAccountMBS, byref error as NSErrorMBS) as boolean 138
- 6.2.8 addRecord(record as ABRecordMBS, byref error as NSErrorMBS) as boolean 138

6.2.7 addRecord(record as ABRecordMBS, Account as ABAccountMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Adds a record (ABPersonMBS or ABGroup) to the AddressBook Database with given account.

Notes: Returns true if the addition was successful.

On Mac OS X 10.7 the error parameter is set to describe the error.

See also:

- 6.2.6 addRecord(record as ABRecordMBS) as boolean 137
- 6.2.8 addRecord(record as ABRecordMBS, byref error as NSErrorMBS) as boolean 138

6.2.8 addRecord(record as ABRecordMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Adds a record (ABPersonMBS or ABGroup) to the AddressBook Database.

Notes: Returns true if the addition was successful.

On Mac OS X 10.7 the error parameter is set to describe the error.

See also:

- 6.2.6 addRecord(record as ABRecordMBS) as boolean 137
- 6.2.7 addRecord(record as ABRecordMBS, Account as ABAccountMBS, byref error as NSErrorMBS) as boolean 138

6.2.9 addressBook as ABAddressBookMBS

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Returns a new instance of ABAddressBook, or nil if the Address Book database can't be initialized.

Notes: If you're just making one-off lookups and edits, the sharedAddressBook method is probably more appropriate.

If the user denies your application access to the Address Book database, this method returns nil.

Available in OS X v10.5 and later.

You need to use this method if you want to get an addressbook for ABPersonViewMBS.

6.2.10 allAccounts as ABAccountMBS()

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Queries list of all accounts.

Example:

```
dim a as new ABAddressBookMBS
dim accounts() as ABAccountMBS = a.allAccounts
Break // look in debugger
```

6.2.11 Constructor

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The constructor.

6.2.12 enabledAccounts as ABAccountMBS()

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Queries list of enabled accounts.

Example:

```
dim a as new ABAddressBookMBS
dim accounts() as ABAccountMBS = a.enabledAccounts
Break // look in debugger
```

6.2.13 EnableEvent

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Activates the events in this class.

Notes: You only need to call this if you use AddHandler command in Xojo to add event handlers. The plugin automatically does that in the constructor, but that is too early for AddHandler. And plugin on enables events if you use them.

6.2.14 formattedAddressFromDictionary(address as Dictionary) as NSAttributedStringMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns an attributed string containing the formatted address.

Notes: The string's attributes match address dictionary keys (kABAddressStreetKey for example).

Each attribute value contains the localized description of the key. (For example, the value of a Canadian kABAddressZIPKey field would be Postal Code)

6.2.15 GotSharedAddressbook as boolean

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Whether some other part of your app queried the shared addressbook already.

Notes: sharedAddressbook asks for permissions the first time you call it.

So with this function you can check if some other application part already queried the sharedAddressbook function. If true, a call to sharedAddressbook should return quickly. Either with nil (no permissions) or the addressbook.

6.2.16 groupForName(name as string) as ABGroupMBS

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Searches the group with the given name.

Example:

```
dim a as new ABAddressBookMBS
dim name as string = "Some Group"
dim g as ABGroupMBS = a.groupForName(name)
MsgBox g.DisplayName+": "+str(g.members.Ubound+1)
```

6.2.17 groupForUniqueId(uniqueid as string) as ABGroupMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns a ABGroupMBS matching a given unique ID.

Example:

```
dim a as new ABAddressBookMBS

// you have some uid
dim groups() as ABGroupMBS = a.groups
dim uid as string = groups(0).valueForProperty(a.kABUIDProperty)

// later you want to find the group
dim g as ABGroupMBS = a.groupForUniqueId(uid)

// shows the name
MsgBox g.valueForProperty(a.kABGroupNameProperty)
```

Notes: Returns nil if the record could not be found or matches to a person.
 Available in Mac OS X 10.3 or newer.
 see also recordForUniqueId.
 See also:

- 6.2.18 groupForUniqueId(uniqueid as string, account as ABAccountMBS) as ABGroupMBS 141

6.2.18 groupForUniqueId(uniqueid as string, account as ABAccountMBS) as ABGroupMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Finds a group for given unique ID for given account.
 See also:

- 6.2.17 groupForUniqueId(uniqueid as string) as ABGroupMBS 140

6.2.19 groups as ABGroupMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns an array of all the groups in the AddressBook database

Example:

```
// list all email addresses in one group
```

```
Dim book as ABAddressBookMBS
dim groups() as ABGroupMBS
dim person as ABPersonMBS
dim data as ABMultiValueMBS
dim s as string
```

```
book=new ABAddressBookMBS
```

```
groups=book.groups
```

```
for each group as ABGroupMBS in groups
```

```
If group.valueForProperty(book.kABGroupNameProperty)= "test" then // or any valid group
```

```
dim members() as ABPersonMBS = group.members
```

```
for each member as ABPersonMBS in members
```

```
data=person.valueForProperty(book.kABEmailProperty)
```

```

if data<>nil then
for k as Integer=data.count-1 downto 0
s=s+data.valueAtIndex(k)+EndOfLine
next
end if
next
end if
Next
msgBox s

```

Notes: Returns an empty array in case the DB doesn't contain any groups.
Returns nil on any error.

6.2.20 groupsForAccount(account as ABAccountMBS) as ABGroupMBS()

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Returns an array of all the groups for this account.

Example:

```

dim a as new ABAddressBookMBS
dim c as ABAccountMBS = a.defaultAccount
dim groups() as ABGroupMBS = a.groupsForAccount(c)

```

Break // look in debugger

Notes: Returns an empty array in case the DB doesn't contain any body.

6.2.21 kABAddressCityKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the dictionary keys for the address.

6.2.22 kABAddressCountryCodeKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The Country Code of an address.

Notes: kABAddressCountryCodeKey code must be one of the following:

iso country codes

ae = United Arab Emirates
ar = Argentina
at = Austria
au = Australia
ba = Bosnia and Herzegovina
be = Belgium
bg = Bulgaria
bh = Bahrain
br = Brazil
ca = Canada
ch = Switzerland
cn = China
cs = Czech
de = Germany
dk = Denmark
eg = Egypt
es = Spain
fi = Finland
fr = France
gr = Greece
gl = Greenland
hk = Hong Kong
hr = Croatia
hu = Hungary
ie = Ireland
il = Israel
id = Indonesia
in = India
is = Iceland
it = Italy
ja = Japan
jo = Jordan
kr = South Korea
kw = Kuwait
lb = Lebanon
lu = Luxembourg
mk = Macedonia
mx = Mexico
nl = Netherlands
no = Norway
nz = New Zealand
om = Oman
pl = Poland
pt = Portugal
qa = Qatar

ro = Romania
ru = Russian Federation
sa = Saudi Arabia
se = Sweden
sg = Singapore
si = Slovenia
sk = Slovakia
sy = Syrian Arab Republic
tw = Taiwan
tr = Turkey
ua = Ukraine
uk = United Kingdom
us = United States
ye = Yemen
yu = Serbia and Montenegro
za = South Africa

6.2.23 kABAddressCountryKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the dictionary keys for the address.

Notes: kABAddressCountryCodeKey code must be one of the following:
iso country codes

6.2.24 kABAddressHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the home address.

6.2.25 kABAddressProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Street Addresses - kABMultiDictionaryProperty

Notes: This property is used for persons only.

6.2.26 kABAddressStateKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the dictionary keys for the address.

6.2.27 kABAddressStreetKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the dictionary keys for the address.

6.2.28 kABAddressWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the work address.

6.2.29 kABAddressZIPKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the dictionary keys for the address.

6.2.30 kABAIMHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the home AIM Instant Messaging account.

6.2.31 kABAIMInstantProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: AIM Instant Messaging - kABMultiStringProperty

Notes: This property is used for persons only.

Deprecated in Mac OS 10.7. You should use kABInstantMessageProperty.

6.2.32 kABAIMMobileMeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the multistring value for the `kABAIMInstantProperty` property.

Notes: Available on Mac OS X 10.7 or later.

none

6.2.33 `kABAIMWorkLabel` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the work AIM Instant Messaging account.

6.2.34 `kABAlternateBirthdayComponentsProperty` as string

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

Function: Alternate non-Gregorian birth date.

6.2.35 `kABAnniversaryLabel` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the anniversary date.

6.2.36 `kABAssistantLabel` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.37 `kABBirthdayComponentsProperty` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the property names for the addressbook records.

Notes: Available on Mac OS X 10.7 or later.

Birth date - `kABDateComponentsProperty`

6.2.38 kABBirthdayProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Birth date - kABDateProperty

Notes: This property is used for persons only.

6.2.39 kABBrotherLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.40 kABCalendarURIsProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The calendars associated to a person.

Notes: Calendar URIs - kABMultiStringProperty

6.2.41 kABChildLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.42 kABCreationDateProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Creation Date (when first saved) - kABDateProperty

Notes: A property for all records.

6.2.43 kABDatabaseChangedExternallyNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The notification name to use with NSNotificationObserverMBS class.

Notes: Posted when a process other than the current one has changed the Address Book database.

Depending on the operation performed on the address book, one or more of the following keys may be

included in the user-info dictionary: `kABInsertedRecords`, `kABUpdatedRecords`, and `kABDeletedRecords`. The values for each of the keys are the unique IDs of the records that were inserted, updated, or deleted, respectively. If the values for all the keys are nil, every record has changes. For example, this happens when the Address Book database is restored from a backup copy.

The plugin implements this notification for you and calls the `DatabaseChanged` event in `ABAddressBookMBS` class.

6.2.44 `kABDatabaseChangedNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The notification name to use with `NSNotificationObserverMBS` class.

Notes: Posted when this process has changed the Address Book database.

Depending on the operation performed on the address book, one or more of the following keys may be included in the user-info dictionary: `kABInsertedRecords`, `kABUpdatedRecords`, and `kABDeletedRecords`. The values for each of the keys are the unique IDs of the records that were inserted, updated, or deleted, respectively. If the values for all the keys are nil, every record has changes. For example, this happens when the Address Book database is restored from a backup copy.

The plugin implements this notification for you and calls the `DatabaseChanged` event in `ABAddressBookMBS` class.

6.2.45 `kABDeletedRecords` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys contained by the user-info dictionary of the notifications posted by the Address Book framework.

Notes: Records that have been deleted.

6.2.46 `kABDepartmentProperty` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Department name - (Person)

Notes: This property is used for persons only.

6.2.47 kABEmailHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the home email address.

6.2.48 kABEmailMobileMeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for emails.

Notes: Available on Mac OS X 10.7 or later.

MobileMe email

6.2.49 kABEmailProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Email(s) - kABMultiStringProperty

Notes: This property is used for persons only.

6.2.50 kABEmailWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the work email address.

6.2.51 kABFatherLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.52 kABFirstNamePhoneticProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: First name Phonetic - kABStringProperty

Notes: This property is used for persons only.

6.2.53 kABFirstNameProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: First name - kABStringProperty

Example:

```
dim a as new ABAddressBookMBS
dim p as new ABPersonMBS

if not p.setValue("Ben",a.kABFirstNameProperty) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABFirstNameProperty)
end if
```

Notes: This property is used for persons only.

6.2.54 kABFriendLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.55 kABGroupNameProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Name of the group - kABStringProperty

Example:

```
dim theAB as new ABAddressBookMBS
dim result as new ABGroupMBS
dim error as NSErrorMBS
dim b as Boolean=result.SetValue("test",TheAB.kABGroupNameProperty,error)
if not b then
Msgbox("Failed to name group test."+error.description)
else
MsgBox "OK"
end if
```

Notes: This property is used for groups only.

6.2.56 kABHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A generic label.

Notes: All kABXXXXHomeLabel are equivalent to this label.

6.2.57 kABHomePageLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Homepage URL label for the kABURLsProperty.

6.2.58 kABHomePageProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Home Web page - kABStringProperty

Notes: This property is used for persons only.

Deprecated in Mac OS 10.4. You should use kABURLsProperty.

6.2.59 kABICQHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the home ICQ Instant Messaging account.

6.2.60 kABICQInstantProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: ICQ Instant Messaging - kABMultiStringProperty

Notes: Deprecated in Mac OS 10.7. You should use kABInstantMessageProperty.

This property is used for persons only.

6.2.61 kABICQWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the work ICQ Instant Messaging account.

6.2.62 kABInsertedRecords as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys contained by the user-info dictionary of the notifications posted by the Address Book framework.

Notes: Records that have been inserted.

6.2.63 kABInstantMessageProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the addressbook record properties.

Notes: Available on Mac OS X 10.7 or later.

Instant Messaging - kABMultiDictionaryProperty

6.2.64 kABInstantMessageServiceAIM as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

AIM

6.2.65 kABInstantMessageServiceFacebook as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

Facebook

6.2.66 kABInstantMessageServiceGaduGadu as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

Gadu-Gadu

6.2.67 kABInstantMessageServiceGoogleTalk as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

Google Talk

6.2.68 kABInstantMessageServiceICQ as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

ICQ

6.2.69 kABInstantMessageServiceJabber as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

Jabber

6.2.70 kABInstantMessageServiceKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the dictionary for an instant message.

Notes: Available on Mac OS X 10.7 or later.

Dictionary key for the service type, not guaranteed to be present; possible values follow.

6.2.71 kABInstantMessageServiceMSN as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

MSN

6.2.72 kABInstantMessageServiceQQ as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

QQ

6.2.73 kABInstantMessageServiceSkype as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

Skype

6.2.74 kABInstantMessageServiceYahoo as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the kABInstantMessageUsernameKey key.

Notes: Available on Mac OS X 10.7 or later.

Yahoo!

6.2.75 kABInstantMessageUsernameKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the dictionary for an instant message.

Notes: Available on Mac OS X 10.7 or later.

Dictionary key for the instant messaging handle/username

6.2.76 kABJabberHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the home jabber Instant Messaging account.

6.2.77 kABJabberInstantProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Jabber Instant Messaging - kABMultiStringProperty

Notes: Deprecated in Mac OS 10.7. You should use kABInstantMessageProperty.
This property is used for persons only.

6.2.78 kABJabberWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the work jabber Instant Messaging account.

6.2.79 kABJobTitleProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Job Title - kABStringProperty

Notes: This property is used for persons only.

6.2.80 kABLastNamePhoneticProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Last name Phonetic - kABStringProperty

Notes: This property is used for persons only.

6.2.81 kABLastNameProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Last name - kABStringProperty

Example:

```
dim a as new ABAddressBookMBS
```

```
dim p as new ABPersonMBS
```

```
if not p.setValue("Miller",a.kABLastNameProperty) then  
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABLastNameProperty)  
end if
```

Notes: This property is used for persons only.

6.2.82 kABMaidenNameProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The maiden name of the person - kABStringProperty

Example:

```
dim a as new ABAddressBookMBS
dim p as ABPersonMBS = a.owner // get my card
MsgBox p.valueForProperty(a.kABMaidenNameProperty) // show my maiden name
```

Notes: This property is used for persons only.

6.2.83 kABManagerLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.84 kABMiddleNamePhoneticProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The middle name phonic - kABStringProperty

Notes: This property is used for persons only.

6.2.85 kABMiddleNameProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The middle name - kABStringProperty

Notes: This property is used for persons only.

6.2.86 kABMobileMeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the generic labels.

Notes: Available on Mac OS X 10.7 or later.

MobileMe - for AIM or email values

6.2.87 kABModificationDateProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Last saved date - kABDateProperty

Notes: A property for all records.

6.2.88 kABMotherLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.89 kABMSNHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the home MSN Instant Messaging account.

6.2.90 kABMSNInstantProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: MSN Instant Messaging - kABMultiStringProperty

Notes: This property is used for persons only.

Deprecated in Mac OS 10.7. You should use kABInstantMessageProperty.

6.2.91 kABMSNWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the work MSN Instant Messaging account.

6.2.92 kABNicknameProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The nick name of the person - kABStringProperty

Notes: This property is used for persons only.

6.2.93 kABNoteProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Note - kABStringProperty

Example:

```
dim a as new ABAddressBookMBS
dim p as ABPersonMBS = a.owner

// read
MsgBox p.valueForProperty(a.kABNoteProperty).StringValue

// write
if p.setValue("Hello World", a.kABNoteProperty) then
if a.save then
MsgBox "Changed."
end if
end if
```

Notes: This property is used for persons only.

6.2.94 kABOrganizationProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Company name - kABStringProperty

Example:

```
dim a as new ABAddressBookMBS
dim p as new ABPersonMBS

if not p.setValue("My Company",a.kABOrganizationProperty) then
```

```
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABOrganizationProperty)
end if
```

Notes: This property is used for persons only.

6.2.95 kABOtherDateComponentsProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the properties for a addressbook records.

Notes: Available on Mac OS X 10.7 or later.

Dates associated with this person - kABMultiDateComponentsProperty - (Person)

6.2.96 kABOtherDatesProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Dates associated with this person - kABMultiDateProperty - (Person)

Notes: This property is used for persons only.

6.2.97 kABOtherLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A generic label.

Notes: Can be used with any multi-value property.

6.2.98 kABParentLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.99 kABPartnerLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.100 kABPersonFlags as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Various flags - kABIntegerProperty

Example:

```
dim a as new ABAddressBookMBS
dim p as new ABPersonMBS

if not p.setValue("Miller",a.kABLastNameProperty) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABLastNameProperty)
end if

if not p.setValue("Ben",a.kABFirstNameProperty) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABFirstNameProperty)
end if

if not p.setValue("My Company",a.kABOrganizationProperty) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABOrganizationProperty)
end if

if not p.setValue(a.kABShowAsCompany,a.kABPersonFlags) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABPersonFlags)
end if

if a.addRecord(p) then
MsgBox "Record added"
else
MsgBox "Failed to add record"
end if

if a.save then
MsgBox "Changes saved"
else
MsgBox "Failed to save changes"
end if
```

Notes: This property is used for persons only.

6.2.101 kABPhoneHomeFAXLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the home fax number.

6.2.102 kABPhoneHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the home phone number.

ae	= United Arab Emirates
ar	= Argentina
at	= Austria
au	= Australia
ba	= Bosnia and Herzegovina
be	= Belgium
bg	= Bulgaria
bh	= Bahrain
br	= Brazil
ca	= Canada
ch	= Switzerland
cn	= China
cs	= Czech
de	= Germany
dk	= Denmark
eg	= Egypt
es	= Spain
fi	= Finland
fr	= France
gr	= Greece
gl	= Greenland
hk	= Hong Kong
hr	= Croatia
hu	= Hungary
ie	= Ireland
il	= Israel
id	= Indonesia
in	= India
is	= Iceland
it	= Italy
ja	= Japan
jo	= Jordan
kr	= South Korea
kw	= Kuwait
lb	= Lebanon
lu	= Luxembourg
mk	= Macedonia
mx	= Mexico
nl	= Netherlands
no	= Norway
nz	= New Zealand
om	= Oman
pl	= Poland
pt	= Portugal
qa	= Qatar
ro	= Romania
ru	= Russian Federation
sa	= Saudi Arabia

6.2.103 kABPhoneiPhoneLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the mobile phone number (for iPhone).

6.2.104 kABPhoneMainLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the main phone number.

6.2.105 kABPhoneMobileLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the mobile phone number.

6.2.106 kABPhonePagerLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the pager number.

6.2.107 kABPhoneProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Generic phone number - kABMultiStringProperty

Notes: This property is used for persons only.

6.2.108 kABPhoneWorkFAXLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the work fax number.

6.2.109 kABPhoneWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The label for the work phone number.

6.2.110 kABRelatedNamesProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: names related to this person - kABMultiStringProperty

Notes: This property is used for persons only.

6.2.111 kABSisterLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.112 kABSocialProfileProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Social Profiles - kABMultiDictionaryProperty

Notes: Available on Mac OS X 10.7 or later.

The multi dictionary contains dictionaries. Each has keys like kABSocialProfileURLKey, kABSocialProfileUsernameKey, kABSocialProfileUserIdentifierKey and kABSocialProfileServiceKey.

kABSocialProfileServiceKey has values like kABSocialProfileServiceTwitter, kABSocialProfileServiceFacebook, kABSocialProfileServiceLinkedIn, kABSocialProfileServiceFlickr and kABSocialProfileServiceMySpace.

6.2.113 kABSocialProfileServiceFacebook as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: Available on Mac OS X 10.7 or later.

Facebook

6.2.114 kABSocialProfileServiceFlickr as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: Available on Mac OS X 10.7 or later.

Flickr

6.2.115 kABSocialProfileServiceKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for addressbook record values.

Notes: Available on Mac OS X 10.7 or later.

The service for this social profile. Can be kABSocialProfileServiceTwitter, kABSocialProfileServiceFacebook, kABSocialProfileServiceLinkedIn, kABSocialProfileServiceFlickr or kABSocialProfileServiceMySpace.

6.2.116 kABSocialProfileServiceLinkedIn as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: Available on Mac OS X 10.7 or later.

LinkedIn

6.2.117 kABSocialProfileServiceMySpace as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: MySpace

Available on Mac OS X 10.7 or later.

6.2.118 kABSocialProfileServiceSinaWeibo as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: SinaWeibo

Available in Mac OS X 10.8 and newer.

6.2.119 kABSocialProfileServiceTencentWeibo as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: Available on Mac OS X 10.9 or later.

Tencent Weibo

6.2.120 kABSocialProfileServiceTwitter as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: Available on Mac OS X 10.7 or later.

Twitter

6.2.121 kABSocialProfileServiceYelp as string

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

Function: One of the values for the social profile dictionary's kABSocialProfileServiceKey key.

Notes: Available on Mac OS X 10.7 or later.

Yelp

6.2.122 kABSocialProfileURLKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for addressbook record values.

Notes: Available on Mac OS X 10.7 or later.

Service name. Possible values follow.

6.2.123 kABSocialProfileUserIdentifierKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for addressbook record values.

Notes: Available on Mac OS X 10.7 or later.

Service-specific identifier.

6.2.124 kABSocialProfileUsernameKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for addressbook record values.

Notes: Available on Mac OS X 10.7 or later.

User-visible profile name.

6.2.125 kABSpouseLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the labels for the related names.

6.2.126 kABSuffixProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The name suffix - kABStringProperty

Notes: e.g. "Sr." "Jr." "III"

This property is used for persons only.

6.2.127 kABTitleProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: the title of the person - kABStringProperty

Notes: e.g. "Sir" "Duke" "General" "Lord"

This property is used for persons only.

6.2.128 kABUIDProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The UID property - kABStringProperty

Notes: A property for all records.

6.2.129 kABUpdatedRecords as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys contained by the user-info dictionary of the notifications posted by the Address Book framework.

Notes: Records that have been updated.

6.2.130 kABURLsProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: URLs - kABMultiStringProperty

Example:

```
// shows all websites with labels

dim a as new ABAddressBookMBS
dim p as ABPersonMBS = a.owner
dim m as ABMultiValueMBS = p.valueForProperty(a.kABURLsProperty)

dim u as Integer = m.count-1
for i as Integer = 0 to u
dim label as string = m.labelAtIndex(i)
dim value as string = m.valueAtIndex(i)

MsgBox label+" ->" +value
next
```

Notes: This property is used for persons only.

6.2.131 kABWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A generic label.

Notes: All kABXXXXWorkLabel are equivalent to this label

6.2.132 kABYahooHomeLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the home yahoo Instant Messaging account.

6.2.133 kABYahooInstantProperty as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Yahoo Instant Messaging - kABMultiStringProperty

Notes: This property is used for persons only.

Deprecated in Mac OS 10.7. You should use kABInstantMessageProperty.

6.2.134 kABYahooWorkLabel as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A label for the work yahoo Instant Messaging account.

6.2.135 LocalizedPropertyOrLabel(propertyOrLabel as string) as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the localized version of built in properties, labels or keys

Notes: Returns propertyOrLabel if not found (e.g. if not built in).

6.2.136 NewPersonWithVCardRepresentation(data as memoryblock) as ABPersonMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Create a person from a vCard.

Notes: Returns nil on failure.

Convenience function which can be used instead of the ABPersonMBS constructor.

6.2.137 people as ABPersonMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns an array of all the people in the AddressBook database

Example:

```
dim a as new ABAddressBookMBS
```

```
// get all people
```

```
dim p(-1) as ABPersonMBS = a.people
```

```

// walk over people list
for each m as ABPersonMBS in p
try
// ask for image

dim j as NSImageMBS = m.image

// do something with image
if j<>nil then
Backdrop=j.CopyPictureWithMask
end if

catch x as NSEExceptionMBS
// raises exception if there is no image
end try
next

```

Notes: Returns an empty array in case the DB doesn't contain any body.
Returns nil on any error.

6.2.138 peopleForAccount(account as ABAccountMBS) as ABPersonMBS()

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Returns an array of all the people for this account.

Example:

```

dim a as new ABAddressBookMBS
dim c as ABAccountMBS = a.defaultAccount
dim people() as ABPersonMBS = a.peopleForAccount(c)

```

Break // look in debugger

Notes: Returns an empty array in case the DB doesn't contain any body.

6.2.139 peopleForEmail(email as string) as ABPersonMBS()

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Searches a contact for the given email.

Example:

```

dim a as new ABAddressBookMBS
dim persons() as ABPersonMBS = a.peopleForEmail("support@monkeybreadsoftware.de")

if UBound(persons) >= 0 then
MsgBox persons(0).DisplayName
else
MsgBox "nothing found."
end if

```

6.2.140 persistentAccounts as ABAccountMBS()

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Queries list of all persistent accounts.

Example:

```

dim a as new ABAddressBookMBS
dim accounts() as ABAccountMBS = a.persistentAccounts
Break // look in debugger

```

6.2.141 personForUniqueId(uniqueid as string) as ABPersonMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns a ABPersonMBS matching a given unique ID.

Example:

```

dim a as new ABAddressBookMBS

// you have some uid
dim uid as string = a.owner.valueForProperty(a.kABUIDProperty)

// later you want to find the person
dim p as ABPersonMBS = a.personForUniqueId(uid)

// shows the name
MsgBox p.valueForProperty(a.kABFirstNameProperty)

```

Notes: Returns nil if the record could not be found or matches to a group.

Available in Mac OS X 10.3 or newer.

see also recordForUniqueId.

See also:

- 6.2.142 `personForUniqueId(uniqueid as string, account as ABAccountMBS) as ABPersonMBS` 172

6.2.142 `personForUniqueId(uniqueid as string, account as ABAccountMBS) as ABPersonMBS`

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Finds a person for given unique ID for given account.

See also:

- 6.2.141 `personForUniqueId(uniqueid as string) as ABPersonMBS` 171

6.2.143 `recordClassFromUniqueId(uniqueid as string) as string`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Given a record `uniqueid` returns the record class name.

Notes: Return "ABPersonMBS" or "ABGroup" or "" for a given `uniqueid`.

6.2.144 `recordForUniqueId(uniqueid as string) as ABRecordMBS`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns a record (ABPersonMBS or ABGroup) matching a given unique ID.

Example:

```
dim a as new ABAddressBookMBS

// you have some uid
dim uid as string = a.owner.valueForProperty(a.kABUIDProperty)

// later you want to find the person
dim r as ABRecordMBS = a.recordForUniqueId(uid)
if r isa ABPersonMBS then
dim p as ABPersonMBS = ABPersonMBS(r)

// shows the name
MsgBox p.valueForProperty(a.kABFirstNameProperty)
end if
```

Notes: Returns nil if the record could not be found.

Available in Mac OS X 10.3 or newer.

See also:

6.2. CLASS ABADDRESSBOOKMBS 173

- 6.2.145 recordForUniqueId(uniqueid as string, account as ABAccountMBS) as ABRecordMBS 173

6.2.145 recordForUniqueId(uniqueid as string, account as ABAccountMBS) as ABRecordMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Finds a record for given unique ID for given account.

See also:

- 6.2.144 recordForUniqueId(uniqueid as string) as ABRecordMBS 172

6.2.146 recordsMatchingSearchElement(search as ABSearchElementMBS) as ABRecordMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns an array of records matching the given search element

Example:

```
dim ab as new ABAddressBookMBS
```

```
// search for people with birthday, by searching for dates starting 1901.
```

```
dim searchDate as new date( 1901, 1, 1 )
```

```
dim search as ABSearchElementMBS = ab.SearchElementForPersonProperty( ab.kABBirthdayProperty, "",  
"", searchDate, ab.kABGreaterThan )
```

```
// do the search
```

```
dim people() as ABRecordMBS = ab.RecordsMatchingSearchElement( search )
```

```
for each person as ABRecordMBS in people
```

```
dim p as ABPersonMBS = ABPersonMBS( person )
```

```
// now work on them
```

```
next
```

Notes: Returns an empty array if no matches or an error.

6.2.147 removeRecord(record as ABRecordMBS) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes a record (ABPersonMBS or ABGroup) from the AddressBook Database

Notes: Returns true if the removal was successful.

See also:

- 6.2.148 `removeRecord(record as ABRecordMBS, byref error as NSErrorMBS) as boolean` 174

6.2.148 `removeRecord(record as ABRecordMBS, byref error as NSErrorMBS) as boolean`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Removes a record (ABPersonMBS or ABGroup) from the AddressBook Database.

Notes: Returns true if the removal was successful.

On Mac OS X 10.7 the error parameter is set to describe the error.

See also:

- 6.2.147 `removeRecord(record as ABRecordMBS) as boolean` 173

6.2.149 `save as boolean`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Saves changes made since the last save.

Notes: Return true if successful (or there was no change).

See also:

- 6.2.150 `save(byref error as NSErrorMBS) as boolean` 174

6.2.150 `save(byref error as NSErrorMBS) as boolean`

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Saves changes made since the last save.

Notes: Return true if successful (or there was no change).

On Mac OS X 10.5 the error object is returned. On Mac OS X 10.4 this error property is nil and you only can use the result.

See also:

- 6.2.149 `save as boolean` 174

6.2.151 `searchElementForConjunction(conjunction as Integer, children() as ABSearchElementMBS) as ABSearchElementMBS`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Creates a search element combining several sub search elements.

Notes: Convenience function which can be used instead of `ABSearchElementMBS.searchElementForConjunction`.

6.2.152 searchElementForGroupProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer) as ABSearchElementMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns a search element that will search groups.

Notes: Convenience function to be used instead of the searchElementForProperty method in the ABGroupMBS class.

6.2.153 searchElementForPersonProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer) as ABSearchElementMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns a search element that will search people.

Example:

```
// search person by record's unique ID
// this is same as calling recordForUniqueId function directly
dim a as new ABAddressBookMBS
dim e as ABSearchElementMBS

dim PropertyName as string = a.kABUIDProperty
const Label = ""
const Key = ""
const value = "637FA922-7A2B-4F9A-BFA3-023253D4A3D5:ABPerson" // some person ID
const comparison = a.kABEqual

e = ABPersonMBS.searchElementForProperty(PropertyName, label, key, value, comparison)

dim records() as ABRecordMBS = a.recordsMatchingSearchElement(e)

for each r as ABRecordMBS in records
dim p as ABPersonMBS = ABPersonMBS(r)

MsgBox p.DisplayName
next
```

Notes: Convenience function to be used instead of the searchElementForProperty method in the ABPersonMBS class.

6.2.154 setMe(moi as ABPersonMBS)

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Sets "Me" to moi.

Notes: Pass nil to clear "Me".

6.2.155 sharedAddressbook as ABAddressBookMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Returns the shared addressbook object.

Example:

```
// quickly find the addressbook, locate me and display my name:  
MsgBox ABAddressBookMBS.sharedAddressbook.owner.DisplayName
```

Notes: If you call this method several times, the object is cached, so it's only created the first time (singleton).

Returns nil on Windows or Linux or low memory or missing permissions.

6.2.156 sharedAddressbookMT as ABAddressBookMBS

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Returns the shared addressbook object.

Notes: On Mac OS X 10.8, the user will be asked to allow access to the addressbook for your application. As the call to sharedAddressbook blocks in this case, this method can be called on a thread to avoid the blocking of your app.

If you call this method several times, the object is cached, so it's only created the first time (singleton).

Returns nil on Windows or Linux or low memory or missing permissions.

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.

6.2.157 Properties

6.2.158 defaultAccount as ABAccountMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Queries default account.

Example:

```
dim a as new ABAddressBookMBS
MsgBox a.defaultAccount.Name
```

Notes: (Read only property)

6.2.159 defaultCountryCode as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the default country code for records without specified codes.

Notes: Available in Mac OS X 10.3 or newer.

(Read only property)

6.2.160 defaultNameOrdering as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the default name ordering defined by the user in the Address Book preferences.

Notes: Possible values: kABFirstNameFirst or kABLastNameFirst

Available in Mac OS X 10.3 or newer.

(Read only property)

6.2.161 Handle as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: The handle to the used ABAddressbook object.

Notes: (Read and Write property)

6.2.162 hasUnsavedChanges as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns true if they are unsaved changes.

Notes: The unsaved changes flag is automatically set when changes are made.
(Read only property)

6.2.163 owner as ABPersonMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the person that represents the user.

Notes: Returns nil if "me" was never set.
(function is not named me as me is a reserved word in Xojo)
(Read only property)

6.2.164 Events

6.2.165 DatabaseChanged(Externally as boolean, InsertedRecords() as string, UpdatedRecords() as string, DeletedRecords() as string)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: The database has changed.

Notes: Externally: true if changes were made by another application.

InsertedRecords: Record UIDs of records changed. (can be empty)

UpdatedRecords: Record UIDs of records updated. (can be empty)

DeletedRecords: Record UIDs of records deleted. (can be empty)

6.2.166 Constants

Constants

Constant	Value	Description
kABArrayProperty	5	One of the property type constants. Array object.
kABBitsInBitFieldMatch	11	One of the search comparison modes. Supported in Mac OS X 10.3 and newer versions.
kABCcontainsSubString	7	One of the search comparison modes.
kABCcontainsSubStringCaseInsensitive	8	One of the search comparison modes.
kABDataProperty	7	One of the property type constants. Data object.
kABDateComponentsProperty	8	One of the property type constants. Available on Mac OS X 10.7 or later. Date component.
kABDateProperty	4	One of the property type constants. Date.
kABDefaultNameOrdering	0	One of the flags constants for the integer value stored in the property's <code>sonFlags</code> .
kABDictionaryProperty	6	One of the property type constants. Dictionary.
kABDoesNotContainSubString	12	One of the search comparison modes. Supported in Mac OS X 10.4 and newer versions.
kABDoesNotContainSubStringCaseInsensitive	13	One of the search comparison modes. Supported in Mac OS X 10.4 and newer versions.
kABEqual	0	One of the search comparison modes.
kABEqualCaseInsensitive	6	One of the search comparison modes.
kABErrorInProperty	0	One of the property type constants. Invalid property.
kABFirstNameFirst	&h40	One of the flags constants for the integer value stored in the property's <code>sonFlags</code> .
kABGreaterThan	4	One of the search comparison modes.
kABGreaterThanOrEqual	5	One of the search comparison modes.
kABIntegerProperty	2	One of the property type constants. Integer.
kABLastNameFirst	&h20	One of the flags constants for the integer value stored in the property's <code>sonFlags</code> .
kABLessThan	2	One of the search comparison modes.
kABLessThanOrEqual	3	One of the search comparison modes.
kABMultiArrayProperty	261	One of the property type constants. Multiple arrays.
kABMultiDataProperty	263	One of the property type constants. Multiple data values.
kABMultiDateComponentsProperty	264	One of the property type constants. Available on Mac OS X 10.7 or later. Date components.
kABMultiDateProperty	260	One of the property type constants. Multiple date values.
kABMultiDictionaryProperty	262	One of the property type constants.
kABMultiIntegerProperty	258	One of the property type constants. Multiple integer values.
kABMultiRealProperty	259	One of the property type constants. Multiple floating point values.
kABMultiStringProperty	257	One of the property type constants. Multiple strings
kABMultiValueMask	&h100	One of the property type constants. This value is combined with other values to define multi value constants.
kABNameOrderingMask	&h70	One of the flags constants for the integer value stored in the property's <code>sonFlags</code> .
kABNotEqual	1	One of the search comparison modes.
kABNotEqualCaseInsensitive	14	One of the search comparison modes. Supported in Mac OS X 10.4 and newer versions.

Error codes.

Constant	Value	Description
ABAddRecordsError	1001	
ABPropertyReadOnlyError	1014	
ABPropertyUnsupportedBySourceError	1013	
ABPropertyValueValidationError	1012	
ABRemoveRecordsError	1002	

6.3 class ABGroupMBS

6.3.1 class ABGroupMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNGroupMBS class instead. **Function:** ABGroup is a subclass of ABRecord.

Example:

```
dim a as new ABAddressBookMBS
dim g() as ABGroupMBS = a.groups
dim names() as string
for each gg as ABGroupMBS in g
names.append gg.DisplayName
next
MsgBox Join(names,EndOfLine)
```

Notes: It represents a group of people or other groups. No recursions allowed.

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.

Subclass of the ABRecordMBS class.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [MBS Real Studio Plugins, version 12.4pr1](#)
- [Addressbook classes updated](#)

6.3.2 Methods

6.3.3 addMember(group as ABPersonMBS) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Adds person to this group.

Example:

```
dim wbook As new ABAddressBookMBS
dim Group as new ABGroupMBS

if not Group.setValue("Test Group", wbook.kABGroupNameProperty) then
MsgBox "Failed to set group name."
```

```
Return
end if

if not wbook.addRecord(Group) then
MsgBox "Failed to add group to database."
return
end if

dim Person as new ABPersonMBS

if not person.setValue("John", wbook.kABFirstNameProperty) then
MsgBox "Failed to set first name."
Return
end if

if not person.setValue("Miller", wbook.kABLastNameProperty) then
MsgBox "Failed to set last name."
Return
end if

if not wbook.addRecord(person) then
MsgBox "Failed to add person to database."
return
end if

if not group.addMember(person) then
MsgBox "Failed to add person to group."
Return
end if

if not wbook.save then
MsgBox "Failed to save addressbook."
return
end if

MsgBox "Created test group with a person."
```

Notes: Does nothing if person is already part of this group (returns false)
Returns true if successful.

6.3.4 addProperty(propertyName as string, type as Integer) as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Adds property to all group records.

Example:

call `ABGroupMBS.addProperty "GroupWeight",1`

Notes: Property name must be unique.

For types see `typeOfProperty`.

Returns the number of properties successfully added.

6.3.5 addSubgroup(group as ABGroupMBS) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Add group to this group.

Notes: Does nothing if group is already part of this group (returns false)

Recursions are not allowed (returns false)

Returns true if successful

6.3.6 Constructor

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new ABGroup object.

Notes: Available in Mac OS X 10.5 or newer.

See also:

- 6.3.7 Constructor(addressBook as ABAddressBookMBS)

183

6.3.7 Constructor(addressBook as ABAddressBookMBS)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new ABGroup object in the given addressbook.

See also:

- 6.3.6 Constructor

183

6.3.8 members as ABPersonMBS()

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of ABPersonMBS.

Notes: Returns an empty array if this group doesn't contain any people.

Returns an empty array on any error.

6.3.9 parentGroups as ABGroupMBS()

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of ABGroup this group belongs to.

Notes: Returns an empty array if this group doesn't belong to any groups.
Returns an empty array on any error.

6.3.10 properties as string()

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of property names.

Notes: Returns nil on any error.

6.3.11 removeMember(group as ABPersonMBS) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes group from this group.

Notes: Does nothing if group is not part of this group (returns false)
Returns true if successful.

6.3.12 removeProperties(properties() as string) as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes properties from all groups.

Notes: Returns the number of properties successfully removed.

6.3.13 removeProperty(propertyName as string) as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes property from all groups

Notes: Returns the number of properties successfully removed.

6.3.14 removeSubgroup(group as ABGroupMBS) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes group from this group.

Notes: Does nothing if group is not part of this group (returns false).

Returns true if successful.

6.3.15 searchElementForProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer) as ABSearchElementMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns a search element object that searches for records of this type.

Notes: property: The name of the property to search on. It cannot be "".

label: The label name for a multivalue list. If property does not have multiple values, pass "". If property does have multiple values, pass "" to search all the values. By default, ABGroup records don't contain any multivalue list properties.

key: The key name for a dictionary. Pass "" if property is not a dictionary. If property is a dictionary, pass "" to search all keys. By default, ABGroup records don't contain any properties that are dictionaries.

value: What you're searching for. If "", the only supported value for comparison is kABEqual or kABNotEqual.

comparison: The type of comparison to perform and is an ABSearchComparison, such as kABEqual or kABPrefixMatchCaseInsensitive.

6.3.16 subgroups as ABGroupMBS()

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of ABGroup

Notes: Returns an empty array if this group doesn't contain any other groups.

Returns an empty array on any error.

6.3.17 typeOfProperty(propertyName as string) as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the type of a given property.

Notes: Possible values:

```
const kABMultiValueMask = &h100
```

```
const kABErrorInProperty = &h0
const kABStringProperty = &h1
const kABIntegerProperty = &h2
const kABRealProperty = &h3
const kABDateProperty = &h4
const kABArrayProperty = &h5
const kABDictionaryProperty = &h6
const kABDataProperty = &h7
const kABMultiStringProperty = kABMultiValueMask + kABStringProperty
const kABMultiIntegerProperty = kABMultiValueMask + kABIntegerProperty
const kABMultiRealProperty = kABMultiValueMask + kABRealProperty
const kABMultiDateProperty = kABMultiValueMask + kABDateProperty
const kABMultiArrayProperty = kABMultiValueMask + kABArrayProperty
const kABMultiDictionaryProperty = kABMultiValueMask + kABDictionaryProperty
const kABMultiDataProperty = kABMultiValueMask + kABDataProperty
```

6.3.18 Properties

6.3.19 `distributionIdentifierForProperty(propertyName as string, person as ABPersonMBS) as String`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the distribution identifier for a given property and person.

Notes: If not set then returns the property primary identifier.

Returns the distribution identifier or "" if not successful.

(Read and Write computed property)

6.4 class ABMultiValueMBS

6.4.1 class ABMultiValueMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNLabeledValueMBS class instead. **Function:** Represents values of type ABMultiXXXXXProperty.

Example:

```
// shows all websites with labels

dim a as new ABAddressBookMBS
dim p as ABPersonMBS = a.owner
dim m as ABMultiValueMBS = p.valueForProperty(a.kABURLsProperty)

dim u as Integer = m.count-1
for i as Integer = 0 to u
dim label as string = m.labelAtIndex(i)
dim value as string = m.valueAtIndex(i)

MsgBox label+" ->" +value
next
```

Notes: All values in an ABMultiValue must be of the same type (kABMultiStringProperty: all values must be strings....)

In case your application needs to store away a reference to a specific value/label pair, use the identifier. Index won't work in this case because any client can add/remove/reorder a multivalue making your index point to the wrong pair. Identifiers are unique Ids.

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.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [MBS Xojo Plugins, version 18.2pr8](#)
- [MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 15.0](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr7](#)
- [Using dash if to reduce app size by referencing less plugins](#)

- [MBS Real Studio Plugins, version 12.3pr2](#)
- [MBS Real Studio Plugins, version 11.3fc](#)

Xojo Developer Magazine

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

6.4.2 Methods

6.4.3 Constructor

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

Function: The private constructor.

6.4.4 copy as ABMultiValueMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Creates an immutable copy of the data.

Notes: Returns nil on any error.

6.4.5 edit as ABMutableMultiValueMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Creates a mutable copy of the data.

Notes: Returns nil on any error.

6.4.6 identifierAtIndex(index as UInt32) as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns an identifier at a given index

Notes: Returns "" on any error.

Index is zero based.

6.4.7 identifiers as string()

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Returns list of identifiers.

6.4.8 indexForIdentifier(identifier as string) as UInt32

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the index of a given identifier

Notes: Returns NotFound (&h7ffffff) on any error.

6.4.9 indexForLabel(label as string) as UInt32

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

Function: Returns the index of a given label

Notes: Returns NotFound (&h7ffffff) on any error.

6.4.10 labelAtIndex(index as UInt32) as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns a label at a given index.

Example:

```
// get an entry
dim a as new ABAddressBookMBS
dim p as ABPersonMBS = a.owner

// query all emails
dim e as ABMultiValueMBS = p.valueForProperty(a.kABEmailProperty)

// walk over all
dim u as Integer = e.count-1
for i as Integer = 0 to u
dim label as string = e.labelAtIndex(i)
dim value as string = e.valueAtIndex(i)

// show label and value for this entry
MsgBox str(i)+"": "+label+", "+value

// is it home?
if label = a.kABEmailHomeLabel then
MsgBox "Home: "+value
end if
```

[next](#)

Notes: Returns "" on any error.
Index is zero based.

6.4.11 labelForIdentifier(identifier as string) as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the label with the given identifier.

Example:

// same as the following code, but with index bound checking:

```
dim s as string
dim identifier as string = "fill the identifier here"
dim a as ABMultiValueMBS
// get multivalue somehow

s=a.labelAtIndex(a.indexForIdentifier(identifier))
```

Notes: Returns "" if identifier is not found.
A convenience method.

6.4.12 labels as string()

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Returns list of labels.

6.4.13 valueAtIndex(index as UInt32) as Variant

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns a value at a given index

Example:

// shows all websites with labels

```
dim a as new ABAddressBookMBS
dim p as ABPersonMBS = a.owner
```

```

dim m as ABMultiValueMBS = p.valueForProperty(a.kABURLsProperty)

dim u as Integer = m.count-1
for i as Integer = 0 to u
dim label as string = m.labelAtIndex(i)
dim value as string = m.valueAtIndex(i)

MsgBox label+" ->" +value
next

```

Notes: Returns nil on any error.
Index is zero based.

6.4.14 valueForIdentifier(identifier as string) as Variant

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the value with the given identifier.

Notes: Returns nil if identifier is not found.

A convenience method.

6.4.15 valueForLabel(label as string) as Variant

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the value for the value with the given label.

Example:

```

dim a as new ABAddressBookMBS
dim p as ABPersonMBS
dim m as ABMultiValueMBS

a=new ABAddressBookMBS
p=a.owner

m=p.valueForProperty(a.kABPhoneProperty)

MsgBox "Work phone: "+m.valueForLabel(a.kABPhoneWorkLabel)
MsgBox "Mobile phone: "+m.valueForLabel(a.kABPhoneMobileLabel)

```

Notes: Returns nil if not value exists for the label.
A convenience method.

6.4.16 values as Variant()

Plugin Version: 16.1, Platform: macOS, Targets: Desktop only.

Function: Returns list of all values.

Notes: Returns nil on any error.

Useful to get all email addresses as array.

6.4.17 Properties

6.4.18 Addressbook as ABAddressBookMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Reference to parent addressbook.

Notes: Plugin sets this for most objects to keep reference to addressbook and avoid this addressbook from being closed too early.

(Read and Write property)

6.4.19 Content as Dictionary

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Returns content of multi value as dictionary.

Notes: This is more for inspection in debugger.

The plugin will add key and value for all values.

Values will be twice in the dictionary, once with label and once with identifier as key.

So if two items have same label, one will overwrite other in dictionary.

(Read only property)

6.4.20 count as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the number of value/label pairs.

Notes: Returns 0 on any error.

(Read only property)

6.4.21 Description as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: The description for this multi value.

Example:

```
dim a as new ABAddressBookMBS
dim m as new ABMutableMultiValueMBS
call m.insertValue("Hello World", "Owner", 0)
MsgBox m.Description
```

Notes: (Read only property)

6.4.22 Handle as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: The handle of the used ABMultiValue object.

Notes: (Read and Write property)

6.4.23 primaryIdentifier as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Identifier for the primary value.

Notes: Returns "" on any error.

(Read only property)

6.4.24 propertyType as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Type of this multivalued (kABMultiXXXXProperty)

Notes: Returns kABErrorInProperty if this multi-value is empty or not all values have the same type.

Possible values:

```
const kABMultiValueMask = &h100
const kABErrorInProperty = &h0
const kABStringProperty = &h1
```

```
const kABIntegerProperty = &h2
const kABRealProperty = &h3
const kABDateProperty = &h4
const kABArrayProperty = &h5
const kABDictionaryProperty = &h6
const kABDataProperty = &h7
const kABMultiStringProperty = kABMultiValueMask + kABStringProperty
const kABMultiIntegerProperty = kABMultiValueMask + kABIntegerProperty
const kABMultiRealProperty = kABMultiValueMask + kABRealProperty
const kABMultiDateProperty = kABMultiValueMask + kABDateProperty
const kABMultiArrayProperty = kABMultiValueMask + kABArrayProperty
const kABMultiDictionaryProperty = kABMultiValueMask + kABDictionaryProperty
const kABMultiDataProperty = kABMultiValueMask + kABDataProperty
(Read only property)
```

6.5 class ABMutableMultiValueMBS

6.5.1 class ABMutableMultiValueMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNLabeledValueMBS class instead. **Function:** Mutable variant of ABMultiValueMBS.

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.

Subclass of the ABMultiValueMBS class.

Blog Entries

- [Saying goodbye to AddressBook framework](#)

6.5.2 Methods

6.5.3 addValue(value as Variant, label as string) as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Adds a value with its label.

Notes: Returns the identifier if successful, "" otherwise.

Note: No type checking is made when adding a value. But trying to set a multivalue property with a multivalue that doesn't have all its values of the same type will return an error.

Supported types: Date, Integer, String, Dictionary.

6.5.4 Constructor

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new ABMutableMultiValue object.

6.5.5 insertValue(value as Variant, label as string, index as UInt32) as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Insert a value/label pair at a given index.

Notes: Returns the identifier if successful. "" otherwise

Note: No type checking is made when adding a value. But trying to set a multivalue property with a multivalue that doesn't have all its values of the same type will return an error

Index is zero based.

6.5.6 `removeValueAndLabelAtIndex(index as UInt32)` as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes a value/label pair at a given index

Notes: Returns true if successful.

Index is zero based.

6.5.7 `replaceLabelAtIndex(index as UInt32, label as string)` as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Replaces a label at a given index.

Notes: Index is zero based.

Returns true on success.

6.5.8 `replaceValueAtIndex(index as UInt32, value as Variant)` as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Replaces a value at a given index

Notes: Index is zero based.

Returns true on success.

6.5.9 `setPrimaryIdentifier(identifier as string)` as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Sets the primary value given its identifier.

Notes: Returns true if successful.

6.6 class ABPersonMBS

6.6.1 class ABPersonMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNContactMBS class instead. **Function:** ABPersonMBS is a subclass of ABRecord and represents a person.

Example:

```
dim a as new ABAddressBookMBS // get addressbook
dim p as ABPersonMBS = a.owner // and find me

// read note
MsgBox p.valueForProperty(a.kABNoteProperty).StringValue

// write note
if p.setValue("Hello World", a.kABNoteProperty) then
if a.save then
MsgBox "Changed."
end if
end if
```

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. Subclass of the ABRecordMBS class.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [Using dash if to reduce app size by referencing less plugins](#)
- [MBS Real Studio Plugins, version 12.4pr1](#)
- [Tipp of the day: Jump to anchor in htmlviewer](#)

6.6.2 Methods

6.6.3 addProperty(propertyName as string, type as Integer) as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Adds property to all people records.

Example:

```

dim p as ABPersonMBS
// get a person

call p.addProperty "Distance",1

```

Notes: Property name must be unique.
 For types see typeOfProperty.
 Returns the number of properties successfully added.

6.6.4 Constructor

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new ABPersonMBS object.
 See also:

- 6.6.5 Constructor(addressBook as ABAddressBookMBS) 198
- 6.6.6 Constructor(vCardData as Memoryblock) 198

6.6.5 Constructor(addressBook as ABAddressBookMBS)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new ABPersonMBS object in the given addressbook.
Notes: Available in Mac OS X 10.5 or newer.
 See also:

- 6.6.4 Constructor 198
- 6.6.6 Constructor(vCardData as Memoryblock) 198

6.6.6 Constructor(vCardData as Memoryblock)

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Create a person from a vCard
Notes: Handle is 0 on failure after constructor finished.
 (e.g. because of invalid vCard data)

This gives a temporary ABPersonMBS object which is only useful in the same method.
 At least we observed problems and crashes when this person is stored in a property of a window and used later.
 See also:

6.6. CLASS ABPERSONMBS	199
• 6.6.4 Constructor	198
• 6.6.5 Constructor(addressBook as ABAddressBookMBS)	198

6.6.7 EditInAddressbook as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Opens addressbook entry in the addressbook for editing.

Notes: Returns true on success and false on failure.

6.6.8 linkedPeople as ABPersonMBS()

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Returns an array of all linked people, including this person.

Notes: Returns an array of only this person if this person is not linked.
Available in Mac OS X 10.8 and newer.

6.6.9 parentGroups as ABGroupMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns an array of ABGroup this group belongs to.

Notes: Returns an empty array if this person doesn't belong to any groups.
Returns an empty array on any error.

6.6.10 properties as string()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns an array of property names.

Notes: Returns an empty array on any error.

6.6.11 removeProperties(properties() as string) as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes properties from all people

Notes: Returns the number of properties successfully removed.

6.6.12 `removeProperty(propertyName as string)` as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Removes property from all people

Notes: Returns the number of properties successfully removed.

6.6.13 `searchElementForProperty(PropertyName as string, Label as string, Key as string, value as Variant, comparison as Integer)` as ABSearchElementMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns a search element object that specifies a query for records of this type.

Notes: `propertyName`: The name of the property to search on, such as `kABAddressProperty` or `kABLastNameProperty`. This name cannot be "".

`label`: The label name for a multivalue list, such as `kABAddressHomeLabel`, `kABPhoneWorkLabel`, or a user-specified label, such as Summer Home. If the specified property does not have multiple values, pass "". If the specified property does have multiple values, pass "" to search all the values.

`key`: The key name for a dictionary, such as `kABAddressCityKey` or `kABAddressStreetKey`. If the specified property is not a dictionary, pass "". If the specified property is a dictionary, pass nil to search all keys.

`value`: What you're searching for. If nil, then the only supported value for comparison is `kABEqual` or `kABNotEqual`.

`comparison`: The type of comparison to perform, such as `kABEqual` or `kABPrefixMatchCaseInsensitive`.

6.6.14 `setImageData(data as Memoryblock)` as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Set the image of a person to data. data should be in an NSImage/QuickTime compatible format.

Notes: Pass "" to clear the image.

6.6.15 `ShowInAddressbook` as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Shows addressbook entry in the addressbook.

Example:

```
// open addressbook
dim a as new ABAddressBookMBS
```

```
// pick a person, in this case me
```

```
dim p as ABPersonMBS = a.owner
```

```
// show in AddressBook
call p.ShowInAddressbook
```

Notes: Returns true on success and false on failure.

6.6.16 typeOfProperty(propertyName as string) as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the type of a given property.

Notes: Possible values:

```
const kABMultiValueMask = &h100
const kABErrorInProperty = &h0
const kABStringProperty = &h1
const kABIntegerProperty = &h2
const kABRealProperty = &h3
const kABDateProperty = &h4
const kABArrayProperty = &h5
const kABDictionaryProperty = &h6
const kABDataProperty = &h7
const kABMultiStringProperty = kABMultiValueMask + kABStringProperty
const kABMultiIntegerProperty = kABMultiValueMask + kABIntegerProperty
const kABMultiRealProperty = kABMultiValueMask + kABRealProperty
const kABMultiDateProperty = kABMultiValueMask + kABDateProperty
const kABMultiArrayProperty = kABMultiValueMask + kABArrayProperty
const kABMultiDictionaryProperty = kABMultiValueMask + kABDictionaryProperty
const kABMultiDataProperty = kABMultiValueMask + kABDataProperty
```

6.6.17 vCardRepresentation as Memoryblock

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the vCard representation of a person

Notes: Returns nil on any error.

6.6.18 Properties

6.6.19 image as NSImageMBS

Plugin Version: 7.5, Platform: macOS, Targets: Desktop only.

Function: Synchronously returns data containing an image for this person.

Example:

```
dim a as new ABAddressBookMBS
dim p as ABPersonMBS = a.owner
```

```
Backdrop=p.image.CopyPictureWithMask
```

Notes: Only does local file system searches.
Raises an exception if no image exists.

Convenience function which calls imageData and converts data to and from NSImage.
(Read and Write computed property)

6.6.20 imageData as Memoryblock

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Synchronously returns data containing an image for this person.

Example:

```
dim a as new ABAddressBookMBS
dim owner as ABPersonMBS = a.owner
dim s as string = owner.imageData
dim p as Picture
```

```
p=JPEGStringToPictureMBS(s) // try jpeg
if p<>Nil then
  Title="jpeg"
  Backdrop=p
  Return
end if
```

```
p=TIFFStringToPictureMBS(s) // try tiff
if p<>Nil then
  Title="tiff"
  Backdrop=p
  Return
end if
```

Notes: Only does local file system searches. Data will be in an NSImage/QuickTime compatible format.
Raises an exception if no image exists.
(Read and Write computed property)

6.7 class ABPickerMBS

6.7.1 class ABPickerMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNContactPickerMBS class instead. **Function:** A class to control a people picker window.

Example:

```
dim p as ABPickerMBS // a global property
```

```
p = new ABPickerMBS
p.Create
p.visible = True
```

Notes: Requires Mac OS X 10.3 or newer.

This class is for Carbon as event handling does not work on Cocoa.
For Cocoa better use ABPeoplePickerViewMBS.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [MBS Xojo Plugins, version 20.5pr9](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr6](#)
- [MBS Real Studio Plugins, version 12.3pr10](#)

6.7.2 Methods

6.7.3 AddProperty(propertyname as String)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Adds a property to the value column.

Notes: Requires Mac OS X 10.3 or newer.

6.7.4 ClearSearchField

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Clear the search field and reset the list of displayed names.

Notes: Requires Mac OS X 10.3 or newer.

6.7.5 Create

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new window.

Example:

```
dim p as ABPickerMBS // a global property
```

```
p = new ABPickerMBS  
p.Create
```

```
p.visible = True
```

Notes: The window is created invisible.

Requires Mac OS X 10.3 or newer.

The handle property is not 0 if this call was successful.

6.7.6 DeselectAll

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Remove selection.

6.7.7 DeselectGroup(group as ABGroupMBS)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Removes selection.

Notes: Requires Mac OS X 10.3 or newer.

6.7.8 DeselectIdentifier(person as ABPersonMBS, Identifier as String)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Removes selection.

Notes: Requires Mac OS X 10.3 or newer.

6.7.9 DeselectPerson(person as ABPersonMBS)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Removes selection.

Notes: Requires Mac OS X 10.3 or newer.

6.7.10 EditInAddressBook

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Launch AddressBook and edit the current selection

Notes: Requires Mac OS X 10.3 or newer.

6.7.11 InstallEvents(targetwindow as window)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Installs the event handler.

Notes: Requires Mac OS X 10.3 or newer.

The target window is only used as an anchor to send and receive events. You can use any window for that.

6.7.12 Properties as string()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A list of all the properties shown in the value columns.

Notes: Requires Mac OS X 10.3 or newer.

6.7.13 RemoveEvents

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Removes the event handler.

Notes: Requires Mac OS X 10.3 or newer.

6.7.14 RemoveProperty(propertyname as String)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Removes a property from the value column.

Notes: Requires Mac OS X 10.3 or newer.

6.7.15 SelectedDictionaries as Dictionary()

Plugin Version: 9.0, Platform: macOS, Targets: Desktop only.

Function: Returns an array containing dictionaries for each item selected in the values column.

Notes: Use this method if you select single addresses.

Requires Mac OS X 10.3 or newer.

6.7.16 SelectedGroups as ABGroupMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns group column selection as an array of ABGroup object handles.

Notes: Requires Mac OS X 10.3 or newer.

6.7.17 SelectedIdentifiers(person as ABPersonMBS) as string()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: This method returns an array of selected multi-value identifiers.

Notes: Requires Mac OS X 10.3 or newer.

Returns empty array if the displayed property is a single value type.

6.7.18 SelectedRecords as ABRecordMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns names column selection as an array of ABGroup or ABPersonMBS objects.

Notes: Requires Mac OS X 10.3 or newer.

You need to cast the objects from the array to ABGroupMBS or ABPersonMBS to use them better. And IsA can tell you whether an object is from the group or the person class.

6.7.19 SelectedStrings as String()

Plugin Version: 9.0, Platform: macOS, Targets: Desktop only.

Function: Returns an array containing strings for each item selected in the values column.

Notes: Use this method if you select single strings like a phone number.

Requires Mac OS X 10.3 or newer.

6.7.20 SelectedValues as Variant()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns an array containing variants for each item selected in the values column.

Notes: Use this method if you select single strings like a phone number.

Requires Mac OS X 10.3 or newer.

Changed from string to variant in plugin version 13.2, so you don't need to use SelectedDictionaries.

6.7.21 SelectGroup(group as ABGroupMBS, ExtendSelection as boolean)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Select group programatically.

Notes: Requires Mac OS X 10.3 or newer.

6.7.22 SelectIdentifier(person as ABPersonMBS, Identifier as String, ExtendSelection as boolean)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Individual values contained within an multi-value property can be selected with this method.

Notes: Requires Mac OS X 10.3 or newer.

6.7.23 SelectInAddressBook

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Launch AddressBook and select the current selection

Notes: Requires Mac OS X 10.3 or newer.

6.7.24 SelectPerson(person as ABPersonMBS, ExtendSelection as boolean)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Select person programatically.

Notes: Requires Mac OS X 10.3 or newer.

6.7.25 Properties

6.7.26 AllowGroupSelection as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Allow the user to select entire groups in the group column.

Notes: Requires Mac OS X 10.3 or newer.

If false, at least one person in the group will be selected. Defaults to false.

(Read and Write property)

6.7.27 AllowMultipleSelection as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Allow the user to select more than one group/record at a time.

Notes: Default is true.

Requires Mac OS X 10.3 or newer.

(Read and Write property)

6.7.28 AllowMultipleValueSelection as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Allow user to choose multiple values for a person.

Notes: Requires Mac OS X 10.3 or newer.

Choose the selection behavior for the value column. If multiple behaviors are selected, the most restrictive behavior will be used. Defaults to SingleValueSelection set.

(Read and Write property)

6.7.29 AllowSingleValueSelection as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Allow user to choose a single value for a person.

Notes: Requires Mac OS X 10.3 or newer.

Choose the selection behavior for the value column. If multiple behaviors are selected, the most restrictive behavior will be used. Defaults to SingleValueSelection set.

(Read and Write property)

6.7.30 Available as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: True if the picker is available.

Notes: Returns true on Mac OS X 10.3 or newer.

(Read only property)

6.7.31 DisplayedProperty as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Which property is displayed currently.

Notes: Returns nil on any error.

(Read and Write property)

6.7.32 Handle as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The handle of the ABPickerRef used internally.

Notes: (Read and Write property)

6.7.33 Height as Single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current height of the picker window.

Notes: Requires Mac OS X 10.3 or newer.

(Read and Write property)

6.7.34 Left as Single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current position of the picker window.

Notes: Requires Mac OS X 10.3 or newer.

(Read and Write property)

6.7.35 Top as Single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current position of the picker window.

Notes: Requires Mac OS X 10.3 or newer.

top=0 is on the bottom of the screen as this is the Cocoa coordinate system.

(Read and Write property)

6.7.36 Visible as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Whether the picker window is visible.

Notes: The window is created invisible, so you must make it visible to show it to the user.

Requires Mac OS X 10.3 or newer.

(Read and Write property)

6.7.37 Width as Single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current width of the picker window.

Notes: Requires Mac OS X 10.3 or newer.

(Read and Write property)

6.7.38 ColumnTitle(columntitle as String) as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Localized titles for third party properties.

Notes: Requires Mac OS X 10.3 or newer.

(Read and Write computed property)

6.7.39 Events

6.7.40 DisplayedPropertyChanged

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: One of the events of a People Picker.

Notes: Requires Mac OS X 10.3 or newer.

6.7.41 GroupDoubleClicked

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: One of the events of a People Picker.

Notes: Requires Mac OS X 10.3 or newer.

6.7.42 GroupSelectionChanged

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: One of the events of a People Picker.

Notes: Requires Mac OS X 10.3 or newer.

6.7.43 NameDoubleClicked

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: One of the events of a People Picker.

Notes: Requires Mac OS X 10.3 or newer.

6.7.44 NameSelectionChanged

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: One of the events of a People Picker.

Notes: Requires Mac OS X 10.3 or newer.

6.7.45 ValueSelectionChanged

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: One of the events of a People Picker.

Notes: Requires Mac OS X 10.3 or newer.

6.8 class ABRecordMBS

6.8.1 class ABRecordMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CNContactMBS class instead. **Function:** A class to hold an Addressbook Record.

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.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [MBS Real Studio Plugins, version 12.3pr2](#)
- [MBS Real Studio Plugins, version 11.4pr2](#)

6.8.2 Methods

6.8.3 Constructor

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

Function: The private constructor.

6.8.4 removeValueForProperty(propertyName as string) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Remove the value of a given property.

Notes: Subsequent calls to valueForProperty on the same property will return nil.

Returns true if the value was removed successfully and false on any error.

6.8.5 setValue(value as Variant, propertyName as string) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Set the value of a given property. The type of the value must match the property type.

Example:

```
dim a as ABAddressBookMBS
dim p as ABPersonMBS
```

```
// get a somehow
// get p somehow

if not p.setValue("My Company",a.kABOrganizationProperty) then
MsgBox "Failed to set field "+a.LocalizedPropertyOrLabel(a.kABOrganizationProperty)
end if
```

Notes: Value can be Date, Integer, Double, Dictionary, MultiValueMBS/MutableMultiValueMBS or String.
Returns true if the value was set successfully
See also:

- 6.8.6 setValue(value as Variant, propertyName as string, byref error as NSErrorMBS) as boolean 215

6.8.6 setValue(value as Variant, propertyName as string, byref error as NSErrorMBS) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Set the value of a given property. The type of the value must match the property type.
Notes: Value can be Date, Integer, Double, Dictionary, MultiValueMBS/MutableMultiValueMBS or String.
Returns true if the value was set successfully

On Mac OS X 10.7 or later, we set the error property on any error.
See also:

- 6.8.5 setValue(value as Variant, propertyName as string) as boolean 214

6.8.7 valueForProperty(PropertyName as string) as Variant

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the value of a given property.

Example:

```
dim a as new ABAddressBookMBS // get addressbook
dim p as ABPersonMBS = a.owner // and find me

// read note
MsgBox p.valueForProperty(a.kABNoteProperty).StringValue
```

Notes: The type of the value depends on the property type.
Returns nil on any error.

Tip: Put the result in a variant, so you can see the type in the debugger. Emails for example can be a ABMultiValueMBS object while name is normally a string.

6.8.8 Properties

6.8.9 account as ABAccountMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Queries account for this record.

Notes: (Read only property)

6.8.10 Addressbook as ABAddressBookMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Reference to parent addressbook.

Notes: Plugin sets this for most objects to keep reference to addressbook and avoid this addressbook from being closed too early.

(Read and Write property)

6.8.11 Description as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: The description for this record.

Example:

```
dim a as new ABAddressBookMBS
dim m as ABPersonMBS = a.owner
MsgBox m.Description
```

Notes: (Read only property)

6.8.12 DisplayName as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The display name.

Example:

```
// quickly find the addressbook, locate me and display my name:
MsgBox ABAddressBookMBS.sharedAddressbook.owner.DisplayName
```

Notes: For a group, the group name, for an organization the organization name and for a normal person the first name, last name, prefix/suffix and middle name. Name order depends on the settings for person or addressbook.

(Read only property)

6.8.13 Handle as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: The handle of the ABGroup or ABPersonMBS object being used.

Notes: (Read and Write property)

6.8.14 isReadOnly as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns whether or not the record is read only.

Notes: Available on Mac OS X 10.4.

Returns false on older systems or other errors.

(Read only property)

6.8.15 uniqueId as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Convenience method to return the unique ID of a record.

Example:

```
dim a as new ABAddressBookMBS
dim o as ABPersonMBS = a.owner

// the unique ID for this record including the type
dim u as string = o.uniqueId

// the raw ID as Apple stores it
dim i as string = o.valueForProperty("com.apple.uuid")

// show it
MsgBox u+EndOfLine+i
```

Notes: Equivalent to `valueForProperty(kABUIDProperty)`.
(Read only property)

6.9 class ABSearchElementMBS

6.9.1 class ABSearchElementMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use `CNContactFetchRequestMBS` class instead. **Function:** A class for a search element.

Example:

```
// search and display all entries with a given name in a 2 column listbox with address

// Save reference to caller.
dim addr as new ABAddressBookMBS // Initialise the Address Book plugin
dim searchName as string = "Schmitz" // search on.

// Do the search
dim srch as ABSearchElementMBS = addr.searchElementForPersonProperty(addr.kABLastNameProperty,
"", "", searchName, addr.kABContainsSubStringCaseInsensitive)
dim srchRes() as ABRecordMBS = addr.recordsMatchingSearchElement(srch) // Get the results into an
array

for each rc as ABRecordMBS in srchRes
if rc isa ABPersonMBS then // Is it a person record?
dim pers as ABPersonMBS = ABPersonMBS(rc) // Get it into a personnel record

// Now get out the names and addresses.
listbox1.AddRow(pers.valueForProperty(addr.kABFirstNameProperty) + " " + pers.valueForProperty(addr.kABLast-
NameProperty))
listbox1.RowTag(listbox1.LastIndex) = pers.valueForProperty(addr.kABUIDProperty)

// Need to find the home address.
dim mlv as ABMultiValueMBS = pers.valueForProperty(addr.kABAddressProperty)
if mlv <> nil Then
// get home address
dim d as Dictionary = mlv.valueForLabel(addr.kABHomeLabel)

if d = nil then
// get primary
d = mlv.valueForIdentifier(mlv.primaryIdentifier)
end if

if d<>Nil then
// show address with street and city
listbox1.Cell(listbox1.LastIndex,1) = d.Lookup(addr.kABAddressStreetKey,"")+" "+d.Lookup(addr.kABAd-
dressCityKey,"")
end if
end if
```

```
end if
next
```

Notes: Use `searchElementForProperty` in `ABPersonMBS` and `ABGroupMBS` classes to create objects.

All methods in this class will catch exceptions from Cocoa and raise a `NSEExceptionMBS` 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.

Blog Entries

- [Saying goodbye to AddressBook framework](#)
- [MBS Real Studio Plugins, version 12.3pr2](#)

6.9.2 Methods

6.9.3 Constructor

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

Function: The private constructor.

6.9.4 `matchesRecord(record as ABRecordMBS)` as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Whether a record matches the search element.

Notes: Returns false if `handle=0` or `record=nil` or record does not match. Else yes.

6.9.5 `searchElementForConjunction(conjunction as Integer, children() as ABSearchElementMBS)` as `ABSearchElementMBS`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Creates a search element combining several sub search elements.

Notes: `conjunction` can be `kABSearchAnd` or `kABSearchOr`.

Returns nil on any error.

Pass `kABSearchOr` or `kABSearchAnd` for `conjunction`.

This is a class method. No need to have a valid handle.

6.9.6 Properties

6.9.7 Addressbook as ABAddressBookMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Reference to parent addressbook.

Notes: Plugin sets this for most objects to keep reference to addressbook and avoid this addressbook from being closed too early.

(Read and Write property)

6.9.8 Description as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: The description for this search element.

Example:

```
dim a as new ABAddressBookMBS
dim m as ABSearchElementMBS = a.searchElementForPersonProperty(a.kABFirstNameProperty, "", "",
"John", a.kABContainsSubStringCaseInsensitive)
MsgBox m.Description
```

Notes: (Read only property)

6.9.9 Handle as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: The handle to the Cocoa object being used.

Notes: (Read and Write property)

6.9.10 Constants

Constants

Constant	Value	Description
kABSearchAnd	0	A search conjunction.
kABSearchOr	1	A search conjunction.

Chapter 7

Apple Script

7.1 class NSAppleEventDescriptorMBS

7.1.1 class NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An instance of NSAppleEventDescriptor represents a descriptor—the basic building block for Apple events.

Notes: This class is a wrapper for the underlying Apple event descriptor data type, AEDesc. Scriptable Cocoa applications frequently work with instances of NSAppleEventDescriptor, but should rarely need to work directly with the AEDesc data structure.

A descriptor is a data structure that stores data and an accompanying four-character code. A descriptor can store a value, or it can store a list of other descriptors (which may also be lists). All the information in an Apple event is stored in descriptors and lists of descriptors, and every Apple event is itself a descriptor list that matches certain criteria.

Important: An instance of NSAppleEventDescriptor can represent any kind of descriptor, from a simple value descriptor, to a descriptor list, to a full-fledged Apple event.

Descriptors can be used to build arbitrarily complex containers, so that one Apple event can represent a script statement such as tell application "TextEdit" to get word 3 of paragraph 6 of document 3.

In working with Apple event descriptors, it can be useful to understand some of the underlying data types. You'll find terms such as descriptor, descriptor list, Apple event record, and Apple event defined in Building an Apple Event in Apple Events Programming Guide. You'll also find information on the four-character codes used to identify information within a descriptor. Apple event data types are defined in Apple Event Manager Reference. The values of many four-character codes used by Apple (and in some cases reused by developers) can be found in AppleScript Terminology and Apple Event Codes.

The most common reason to construct an Apple event with an instance of `NSAppleEventDescriptor` is to supply information in a return Apple event. The most common situation where you might need to extract information from an Apple event (as an instance of `NSAppleEventDescriptor`) is when an Apple event handler installed by your application is invoked, as described in "Installing an Apple Event Handler" in *How Cocoa Applications Handle Apple Events*. In addition, if you execute an AppleScript script using the `NSAppleScript` class, you get an instance of `NSAppleEventDescriptor` as the return value, from which you can extract any required information.

When you work with an instance of `NSAppleEventDescriptor`, you can access the underlying descriptor directly, if necessary, with the `aeDesc` method. Other methods, including `descriptorWithType` make it possible to create and initialize instances of `NSAppleEventDescriptor` without creating temporary instances of `memoryblock`.

Cocoa doesn't currently provide a mechanism for applications to directly send raw Apple events (though compiling and executing an AppleScript script with `NSAppleScript` may result in Apple events being sent). However, Cocoa applications have full access to the Apple Event Manager C APIs for working with Apple events. So, for example, you might use an instance of `NSAppleEventDescriptor` to assemble an Apple event and call the Apple Event Manager function `AESend` to send it.

If you need to send Apple events, or if you need more information on some of the Apple event concepts described here, see *Apple Events Programming Guide* and *Apple Event Manager Reference*.

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 19.0pr6](#)
- [MBS Xojo / Real Studio Plugins, version 16.2pr4](#)
- [MBS Xojo / Real Studio Plugins, version 15.5pr1](#)
- [MBS Real Studio Plugins, version 12.1pr5](#)
- [MBS Real Studio Plugins, version 11.3pr7](#)
- [MBS Real Studio Plugins, version 11.2pr11](#)
- [MBS Real Studio Plugins, version 11.2pr9](#)

7.1.2 Methods

7.1.3 `appleEventWithEventClass(eventClass as string, eventID as string, targetDescriptor as NSAppleEventDescriptorMBS, returnID as Int16, transactionID as UInt32) as NSAppleEventDescriptorMBS`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor that represents an Apple event, initialized according to the specified information.

Notes: eventClass: The event class to be set in the returned descriptor.

eventID: The event ID to be set in the returned descriptor.

addressDescriptor: A pointer to a descriptor that identifies the target application for the Apple event. Passing nil results in an Apple event descriptor that has no keyAddressAttr attribute (it is valid for an Apple event to have no target address attribute).

returnID: The return ID to be set in the returned descriptor. If you pass a value of kAutoGenerateReturnID, the Apple Event Manager assigns the created Apple event a return ID that is unique to the current session. If you pass any other value, the Apple Event Manager assigns that value for the ID.

transactionID: The transaction ID to be set in the returned descriptor. A transaction is a sequence of Apple events that are sent back and forth between client and server applications, beginning with the client's initial request for a service. All Apple events that are part of a transaction must have the same transaction ID. You can specify kAnyTransactionID if the Apple event is not one of a series of interdependent Apple events.

Returns a descriptor for an Apple event, initialized according to the specified parameter values, or nil if an error occurs.

Constants such as kAutoGenerateReturnID and kAnyTransactionID are defined in AE.framework, a sub-framework of ApplicationServices.framework.

7.1.4 attributeDescriptorForKeyword(keyword as string) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a descriptor for the receiver's Apple event attribute identified by the specified keyword.

Notes: keyword: A keyword (a four-character code) that identifies the descriptor to obtain.

Returns the attribute descriptor for the specified keyword, or nil if an error occurs.

7.1.5 coerceToDescriptorType(descriptorType as string) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a descriptor obtained by coercing the receiver to the specified type.

Notes: descriptorType: The descriptor type to coerce the receiver to.

Returns a descriptor of the specified type, or nil if an error occurs.

7.1.6 Constructor

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

Function: The private constructor.

7.1.7 copy as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a copy of the apple event descriptor.

7.1.8 currentProcessDescriptor as NSAppleEventDescriptorMBS

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Create and return an application address descriptor using the current process.

Example:

```
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.currentProcessDescriptor
MsgBox "Process ID: "+str(d.processIDValue)
```

Notes: The result is suitable for use as the "targetDescriptor" parameter of appleEventWithEventClass.

7.1.9 descriptorAtIndex(index as Integer) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the descriptor at the specified (one-based) position in the receiving descriptor list.

Example:

```
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.listDescriptor
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithString("Hello")
dim e as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithString("World")
```

```
n.insertDescriptor(d,1)
n.insertDescriptor(e,2)
```

```
MsgBox str(n.numberOfItems)
```

```
dim x1 as NSAppleEventDescriptorMBS = n.descriptorAtIndex(1)
dim x2 as NSAppleEventDescriptorMBS = n.descriptorAtIndex(2)
```

```
MsgBox x1.stringValue+" "+x2.stringValue
```

Notes: Index: The one-based descriptor list position of the descriptor to return.

Returns the descriptor from the specified position (one-based) in the descriptor list, or nil if the specified descriptor cannot be obtained.

7.1.10 descriptorForKeyword(keyword as string) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's descriptor for the specified keyword.

Example:

```
dim n as new NSAppleScriptMBS("return system info")
dim r as NSAppleEventDescriptorMBS = n.execute

dim lines(-1) as string

dim u as Integer = r.numberOfItems
for i as Integer = 1 to u
dim keyword as string = r.keywordForDescriptorAtIndex(i)
dim value as string
dim p as NSAppleEventDescriptorMBS = r.descriptorForKeyword(keyword)
if p<>Nil then value = p.stringValue
lines.Append keyword+": "+value
next

MsgBox Join(lines,EndOfLine)
```

Notes: keyword: A keyword (a four-character code) that identifies the descriptor to obtain.

Returns a descriptor for the specified keyword, or nil if an error occurs.

7.1.11 descriptorWithAlias(item as folderitem) as NSAppleEventDescriptorMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with type typeAlias that stores the specified folderitem reference.

Example:

```

// pick a folderitem
dim folder as FolderItem = SpecialFolder.Desktop

// create value with file reference
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithAlias(folder)

// convert back to FolderItem
dim file as FolderItem = d.FSRefValue

// show path
MsgBox file.NativePath

```

Notes: This type can be converted internally to FSRef descriptor.

7.1.12 descriptorWithApplicationURL(fileURL as string) as NSAppleEventDescriptorMBS

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Create and return an application address descriptor using the file URL for an application.

Example:

```

dim f as FolderItem = SpecialFolder.Applications.Child("Stickies.app")
dim u as string = f.URLPath
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithApplicationURL(u)
MsgBox d.applicationURLValue

```

Notes: The result is suitable for use as the "targetDescriptor" parameter of appleEventWithEventClass. See also:

- 7.1.13 descriptorWithApplicationURL(item as folderitem) as NSAppleEventDescriptorMBS 228

7.1.13 descriptorWithApplicationURL(item as folderitem) as NSAppleEventDescriptorMBS

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Create and return an application address descriptor using the folderitem for an application.

Example:

```

dim f as FolderItem = SpecialFolder.Applications.Child("Stickies.app")
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithApplicationURL(f)
MsgBox d.applicationURLValue

```

Notes: The result is suitable for use as the "targetDescriptor" parameter of `appleEventWithEventClass`.
See also:

- 7.1.12 `descriptorWithApplicationURL(fileURL as string)` as `NSAppleEventDescriptorMBS` 228

7.1.14 `descriptorWithBoolean(value as Boolean)` as `NSAppleEventDescriptorMBS`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with type `typeBoolean` that stores the specified Boolean value.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithBoolean(true)
MsgBox a.stringValue // shows true
```

Notes: Returns a descriptor with the specified Boolean value, or nil if an error occurs.

7.1.15 `descriptorWithBundleIdentifier(BundleID as String)` as `NSAppleEventDescriptorMBS`

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Create and return an application address descriptor using the bundle identifier.

Example:

```
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithBundleIdentifier("com.apple.iCal")
MsgBox d.bundleIDValue
```

Notes: The result is suitable for use as the "targetDescriptor" parameter of `appleEventWithEventClass`.

7.1.16 `descriptorWithCurrentProcessSerialNumber` as `NSAppleEventDescriptorMBS`

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates descriptor with current process serial number.

Example:

```
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithCurrentProcessSerialNumber
MsgBox n.stringValue // shows app name
```

7.1.17 descriptorWithDate(value as date) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor with a date value.

Example:

```
dim d as new date
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithDate(d)

dim x as date = n.dateValue
MsgBox x.LongDate+" "+x.LongTime // shows today
```

7.1.18 descriptorWithDateTime(value as dateTime) as NSAppleEventDescriptorMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor with a date value.

7.1.19 descriptorWithDescriptorType(descriptorType as string, data as memoryblock) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with the specified event type that stores the specified data.

Notes: descriptorType: The descriptor type to be set in the returned descriptor.

data: The data, as a memoryblock, to be set in the returned descriptor.

Returns a descriptor with the specified type and data, or nil if an error occurs.

You can use this method to create a descriptor that you can build into a complete Apple event by calling methods such as setAttributeDescriptor, setDescription, and setParamDescriptor.

See also:

- 7.1.20 descriptorWithDescriptorType(descriptorType as string, data as memoryblock, offset as UInt32,

7.1. CLASS NSAPPLLEEVENTDESCRIPTORMBS	231
length as UInt32) as NSAppleEventDescriptorMBS	231

7.1.20 descriptorWithType(descriptorType as string, data as memoryblock, offset as UInt32, length as UInt32) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with the specified event type that stores the specified data (from a series of bytes).

Notes: descriptorType: The descriptor type to be set in the returned descriptor.

bytes: The data, as a sequence of bytes, to be set in the returned descriptor.

offset: offset in memoryblock.

length: The length, in bytes, of the data to be set in the returned descriptor.

Returns a descriptor with the specified type and data, or nil if an error occurs.

See also:

- 7.1.19 descriptorWithType(descriptorType as string, data as memoryblock) as NSAppleEventDescriptorMBS 230

7.1.21 descriptorWithDouble(value as Double) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor with a double value.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithDouble(5)
MsgBox a.stringValue // shows 5
```

7.1.22 descriptorWithEnumCode(enumerator as string) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with type typeEnumerated that stores the specified enumerator data type value.

Notes: enumerator: A type code that identifies the type of enumerated data to be stored in the returned descriptor.

Returns a descriptor with the specified enumerator data type value, or nil if an error occurs.

7.1.23 `descriptorWithURL(fileURL as string) as NSAppleEventDescriptorMBS`

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor for a file URL.

Example:

```
dim f as FolderItem = SpecialFolder.Applications.Child("Stickies.app")
dim u as string = f.URLPath
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithURL(u)
MsgBox d.fileURLValue
```

See also:

- 7.1.24 `descriptorWithURL(item as folderitem) as NSAppleEventDescriptorMBS` 232

7.1.24 `descriptorWithURL(item as folderitem) as NSAppleEventDescriptorMBS`

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor for a file URL based on the folderitem.

Example:

```
dim f as FolderItem = SpecialFolder.Applications.Child("Stickies.app")
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithURL(f)
MsgBox d.fileURLValue
```

See also:

- 7.1.23 `descriptorWithURL(fileURL as string) as NSAppleEventDescriptorMBS` 232

7.1.25 `descriptorWithInt16(value as Int16) as NSAppleEventDescriptorMBS`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with Apple event type typeSInt16 that stores the specified integer value.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithInt16(5)
MsgBox a.stringValue // shows 5
```

7.1.26 descriptorWithInt32(value as Int32) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with Apple event type typeSInt32 that stores the specified integer value.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithInt32(5)
MsgBox a.stringValue // shows 5
```

Notes: Returns a descriptor containing the specified integer value, or nil if an error occurs.

7.1.27 descriptorWithProcessIdentifier(PID as Integer) as NSAppleEventDescriptorMBS

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Create and return an application address descriptor using the process identifier.

Example:

```
// get my PID
dim p as new ProcessMBS
p.GetCurrentProcess
dim pid as Integer = p.ProcessID

// make process ID descriptor
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithProcessIdentifier(pid)

// show it
MsgBox "ProcessID: "+str(n.processIDValue) + EndOfLine + n.stringValue
```

Notes: The result is suitable for use as the "targetDescriptor" parameter of appleEventWithEventClass.

7.1.28 `descriptorWithSingle(value as single)` as `NSAppleEventDescriptorMBS`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor with a single value.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithSingle(5)
MsgBox a.stringValue // shows 5
```

7.1.29 `descriptorWithString(text as string)` as `NSAppleEventDescriptorMBS`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with type `typeUnicodeText` that stores the text from the specified string.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithString("Hello World")

MsgBox a.stringValue
```

Notes: Returns a descriptor that contains the text from the specified string, or nil if an error occurs.

7.1.30 `descriptorWithTypeCode(typeCode as string)` as `NSAppleEventDescriptorMBS`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor initialized with type `typeType` that stores the specified type value.

Notes: `typeCode`: The type value to be set in the returned descriptor.

Returns a descriptor with the specified type, or nil if an error occurs.

7.1.31 descriptorWithUInt32(value as UInt32) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a descriptor with an unsigned integer value.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithUInt32(5)
MsgBox a.stringValue // shows 5
```

7.1.32 insertDescriptor(descriptor as NSAppleEventDescriptorMBS, index as Integer)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Inserts a descriptor at the specified (one-based) position in the receiving descriptor list, replacing the existing descriptor, if any, at that position.

Notes: descriptor: The descriptor to insert in the receiver. Specifying an index of 0 or count + 1 causes appending to the end of the list.

Index: The one-based descriptor list position at which to insert the descriptor.

Because it actually replaces the descriptor, if any, at the specified position, this method might better be called `replaceDescriptor`. The receiver must be a list descriptor. The indices are one-based. Currently provides no indication if an error occurs.

7.1.33 keywordForDescriptorAtIndex(index as Integer) as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the keyword for the descriptor at the specified (one-based) position in the receiver.

Example:

```
dim n as new NSAppleScriptMBS("return system info")
dim r as NSAppleEventDescriptorMBS = n.execute

dim lines(-1) as string

dim u as Integer = r.numberOfItems
for i as Integer = 1 to u
dim keyword as string = r.keywordForDescriptorAtIndex(i)
dim value as string
dim p as NSAppleEventDescriptorMBS = r.descriptorForKeyword(keyword)
if p<>Nil then value = p.stringValue
```

```
lines.Append keyword+": "+value  
next
```

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

Notes: Index: The one-based descriptor list position of the descriptor to get the keyword for.

Returns the keyword (a four-character code) for the descriptor at the one-based location specified by anIndex, or 0 if an error occurs.

7.1.34 listDescriptor as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and initializes an empty list descriptor.

Notes: Returns an empty list descriptor, or nil if an error occurs.

A list descriptor is a descriptor whose data consists of one or more descriptors. You can add items to the list by calling insertDescriptor or remove them with removeDescriptorAtIndex.

7.1.35 nullDescriptor as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and initializes a descriptor with no parameter or attribute values set.

Notes: Returns a descriptor with no parameter or attribute values set, or nil if an error occurs.

You don't typically call this method, as most NSAppleEventDescriptor instance methods can't be safely called on the returned empty descriptor.

7.1.36 paramDescriptorForKeyword(keyword as string) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a descriptor for the receiver's Apple event parameter identified by the specified keyword.

Notes: keyword: A keyword (a four-character code) that identifies the parameter descriptor to obtain.

Returns a descriptor for the specified keyword, or nil if an error occurs.

7.1.37 print

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

Function: Writes description for this event descriptor to the console.

Notes: You can see result in Console.app.

7.1.38 recordDescriptor as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and initializes a descriptor for an Apple event record whose data has yet to be set.

Notes: Returns an Apple event descriptor whose data has yet to be set, or nil if an error occurs.

An Apple event record is a descriptor whose data is a set of descriptors keyed by four-character codes. You can add information to the descriptor with methods such as `setAttributeDescriptor`, `setDescription`, and `setParameterDescriptor`.

7.1.39 removeDescriptorAtIndex(index as Integer)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Removes the descriptor at the specified (one-based) position in the receiving descriptor list.

Notes: Index: The one-based position of the descriptor to remove.

The receiver must be a list descriptor. The indices are one-based. Currently provides no indication if an error occurs.

7.1.40 removeDescriptorWithKeyword(keyword as string)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Removes the receiver's descriptor identified by the specified keyword.

Notes: keyword: A keyword (a four-character code) that identifies the descriptor to remove.

The receiver must be an Apple event or Apple event record. Currently provides no indication if an error occurs.

7.1.41 removeParamDescriptorWithKeyword(keyword as string)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Removes the receiver's parameter descriptor identified by the specified keyword.

Notes: keyword: A keyword (a four-character code) that identifies the parameter descriptor to remove. Currently provides no indication if an error occurs.

The receiver must be an Apple event or Apple event record, both of which can contain parameters.

7.1.42 send(options as Integer, timeoutInSeconds as Double, byref error as NSErrorMBS) as NSAppleEventDescriptorMBS

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sends an Apple event.

Example:

```
// pick a file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
// make a descriptor for file
dim fd as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithFSRef(f)
// make a descriptor for target app. here by bundle id
dim bd as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithBundleIdentifier("com.apple.finder")
// make a descriptor for apple event, here OpenDocument event
dim ad as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.appleEventWithEventClass("aevt",
"odoc", bd, fd.kAutoGenerateReturnID, fd.kAnyTransactionID)

// assign parameter
ad.setParamDescriptor(fd, "—")

// now run
dim e as NSErrorMBS
dim rd as NSAppleEventDescriptorMBS = ad.send(ad.NSAppleEventSendDefaultOptions, 0.1, e)

// error sending?
if e <> nil then
MsgBox e.LocalizedDescription
end if

Break // inspect rd for success or failure of event?
```

7.1.43 setAttributeDescriptor(descriptor as NSAppleEventDescriptorMBS, keyword as string)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds a descriptor to the receiver as an attribute identified by the specified keyword.

Notes: descriptor: The attribute descriptor to add to the receiver.

keyword: A keyword (a four-character code) that identifies the attribute descriptor to add. If a descriptor with that keyword already exists in the receiver, it is replaced.

The receiver must be an Apple event. Currently provides no indication if an error occurs.

7.1.44 setDescription(descriptor as NSAppleEventDescriptorMBS, keyword as string)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds a descriptor, identified by a keyword, to the receiver.

Notes: descriptor: The descriptor to add to the receiver.

keyword: A keyword (a four-character code) that identifies the descriptor to add. If a descriptor with that keyword already exists in the receiver, it is replaced.

The receiver must be an Apple event or Apple event record. Currently provides no indication if an error occurs.

7.1.45 setParamDescriptor(descriptor as NSAppleEventDescriptorMBS, keyword as string)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds a descriptor to the receiver as an Apple event parameter identified by the specified keyword.

Notes: descriptor: The parameter descriptor to add to the receiver.

keyword: A keyword (a four-character code) that identifies the parameter descriptor to add. If a descriptor with that keyword already exists in the receiver, it is replaced.

The receiver must be an Apple event or Apple event record, both of which can contain parameters.

7.1.46 Properties

7.1.47 aeDesc as Ptr

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a pointer to the AEDesc structure that is encapsulated by the receiver, if it has one.

Notes: If the receiver has a valid AEDesc structure, returns a pointer to it; otherwise returns nil.

(Read only property)

7.1.48 applicationURLValue as String

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The application URL.

Example:

```
dim f as FolderItem = SpecialFolder.Applications.Child("Stickies.app")
dim u as string = f.URLPath
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithApplicationURL(u)
MsgBox d.applicationURLValue
```

Notes: (Read only property)

7.1.49 booleanValue as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as a Boolean value, coercing (to typeBoolean) if necessary.

Example:

```
dim lines(-1) as string

lines.Append "set a to 1"
lines.Append "set b to 1"
lines.Append "return a = b" // return a boolean result

// compile, run and show value
dim source as string = Join(lines,EndOfLine)
dim n as new NSAppleScriptMBS(source)

dim error as Dictionary
dim d as NSAppleEventDescriptorMBS = n.execute(Error)
```

```
MsgBox str(d.booleanValue)
```

Notes: Returns the contents of the descriptor, as a Boolean value, or false if an error occurs.
(Read only property)

7.1.50 bundleIDValue as String

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries the bundle identifier.

Example:

```
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithBundleIdentifier("com.apple.iCal")
MsgBox d.bundleIDValue
```

Notes: (Read only property)

7.1.51 data as Memoryblock

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's data as a memoryblock.

Notes: Returns an instance of memoryblock containing the receiver's data, or nil if an error occurs.
(Read only property)

7.1.52 dateTimeValue as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as a date value.

Notes: (Read only property)

7.1.53 dateValue as date

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as a date value.

Example:

```
dim d as new date
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithDate(d)

dim x as date = n.dateValue
MsgBox x.LongDate+" " +x.LongTime // shows today
```

Notes: (Read only property)

7.1.54 description as string

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

Function: The descriptor for this event.

Example:

```
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithString("Hello")
MsgBox n.description
```

Notes: This is a text representation for debugging.
(Read only property)

7.1.55 descriptorType as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the descriptor type of the receiver.

Notes: (Read only property)

7.1.56 doubleValue as Double

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as a double.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithString("5")
MsgBox str(a.doubleValue) // shows 5
```

Notes: (Read only property)

7.1.57 enumCodeValue as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as an enumeration type, coercing (to typeEnumerated) if necessary.

Notes: Returns the contents of the descriptor, as an enumeration type, or 0 if an error occurs. (Read only property)

7.1.58 eventClass as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the event class for the receiver.

Notes: Returns the event class (a four-character code) for the receiver, or 0 if an error occurs.

The receiver must be an Apple event. An Apple event is identified by its event class and event ID, a pair of four-character codes stored as 32-bit integers. For example, most events in the Standard suite have the four-character code 'core' (defined as the constant kAECoreSuite in AE.framework, a subframework of ApplicationServices.framework). For more information on event classes and event IDs, see Building an Apple Event in Apple Events in Apple Events Programming Guide.

(Read only property)

7.1.59 eventID as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the event ID for the receiver.

Notes: The event ID (a four-character code) for the receiver, or 0 if an error occurs.

The receiver must be an Apple event. An Apple event is identified by its event class and event ID, a pair of four-character codes stored as 32-bit integers. For example, the open Apple event from the Standard suite has the four-character code 'odoc' (defined as the constant kAEOpen in AE.framework, a subframework of ApplicationServices.framework).

(Read only property)

7.1.60 fileURLValue as String

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The file URL.

Example:

```
dim f as FolderItem = SpecialFolder.Applications.Child("Stickies.app")
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithURL(f)
MsgBox d.fileURLValue
```

Notes: (Read only property)

7.1.61 Handle as Integer

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the internal reference to the NSAppleEventDescriptor object.

Notes: (Read and Write property)

7.1.62 int16Value as Int16

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as an int16.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithString("5")
MsgBox str(a.int16Value) // shows 5
```

Notes: (Read only property)

7.1.63 int32Value as Int32

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as an integer, coercing (to typeSInt32) if necessary.

Example:

```
dim a as NSAppleEventDescriptorMBS
```

```
a = NSAppleEventDescriptorMBS.descriptorWithString("5")
MsgBox str(a.int32Value) // shows 5
```

Notes: Returns the contents of the descriptor, as an integer value, or 0 if an error occurs.
(Read only property)

7.1.64 isRecordDescriptor as Boolean

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Return whether or not a descriptor is a record-like descriptor.

Notes: Record-like descriptors function as records, but may have a descriptorType other than AERecord, such as ObjectSpecifier.

(Read only property)

7.1.65 numberOfItems as Integer

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of descriptors in the receiver's descriptor list.

Example:

```
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.listDescriptor
dim d as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithString("Hello")
dim e as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithString("World")
```

```
n.insertDescriptor(d,1)
```

```
n.insertDescriptor(e,2)
```

```
MsgBox str(n.numberOfItems)
```

```
dim x1 as NSAppleEventDescriptorMBS = n.descriptorAtIndex(1)
```

```
dim x2 as NSAppleEventDescriptorMBS = n.descriptorAtIndex(2)
```

```
MsgBox x1.stringValue+" "+x2.stringValue
```

Notes: Returns the number of descriptors in the receiver's descriptor list (possibly 0); returns 0 if an error occurs.

(Read only property)

7.1.66 processIDValue as Integer

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries the process ID.

Example:

```
// get my PID
dim p as new ProcessMBS
p.GetCurrentProcess
dim pid as Integer = p.ProcessID

// make process ID descriptor
dim n as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithIdentifier(pid)

// show it
MsgBox "ProcessID: "+str(n.processIDValue) + EndOfLine + n.stringValue
```

Notes: (Read only property)

7.1.67 returnID as Int16

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's return ID (the ID for a reply Apple event).

Notes: Returns the receiver's return ID (an integer value), or 0 if an error occurs.

(Read only property)

7.1.68 singleValue as single

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as a single value.

Example:

```
dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithString("5")
MsgBox str(a.singleValue) // shows 5
```

Notes: (Read only property)

7.1.69 stringValue as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as a Unicode text string, coercing (to typeUnicodeText) if necessary.

Example:

```
dim a as NSAppleEventDescriptorMBS
```

```
a = NSAppleEventDescriptorMBS.descriptorWithString("Hello World")
```

```
MsgBox a.stringValue
```

Notes: Returns the contents of the descriptor, as a string, or "" if an error occurs.
(Read only property)

7.1.70 transactionID as Int32

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's transaction ID, if any.

Notes: Returns the receiver's transaction ID (an integer value), or 0 if an error occurs.

The receiver must be an Apple event. Currently provides no indication if an error occurs. For more information on transactions, see the description for appleEventWithEventClass.

(Read only property)

7.1.71 typeCodeValue as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as a type, coercing (to typeType) if necessary.

Notes: (Read only property)

7.1.72 UInt32Value as UInt32

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the contents of the receiver as an UInt32.

Example:

```

dim a as NSAppleEventDescriptorMBS
a = NSAppleEventDescriptorMBS.descriptorWithString("5")
MsgBox str(a.UInt32Value) // shows 5

```

Notes: (Read only property)

7.1.73 Constants

Constants

Constant	Value	Description
kAnyTransactionID	0	Special constant for transaction ID. no transaction is in use
kAutoGenerateReturnID	-1	Special constant for return ID. AECreatAppleEvent will generate a session-unique ID internally.

Send Options

Constant	Value	Description
NSAppleEventSendAlwaysInteract	&h30	Server should always interact with user where appropriate.
NSAppleEventSendCanInteract	&h20	Server may try to interact with user.
NSAppleEventSendCanSwitchLayer	&h40	Interaction may switch layer.
NSAppleEventSendDefaultOptions	&h23	Default options: WaitForReply with CanInteract.
NSAppleEventSendDontAnnotate	&h10000	Don't automatically add any sandbox or other annotations to the event.
NSAppleEventSendDontExecute	&h2000	Don't execute this event; used for recording.
NSAppleEventSendDontRecord	&H1000	Don't record this event.
NSAppleEventSendNeverInteract	&h10	Server should not interact with user.
NSAppleEventSendNoReply	1	Sender doesn't want a reply to event.
NSAppleEventSendQueueReply	2	Sender wants a reply but won't wait.
NSAppleEventSendWaitForReply	3	Sender wants a reply and will wait.

7.2 class NSAppleEventHandlerMBS

7.2.1 class NSAppleEventHandlerMBS

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

Function: The class for an apple event handler.

Blog Entries

- [MBS Real Studio Plugins, version 12.5pr3](#)

7.2.2 Methods

7.2.3 Constructor

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

Function: The constructor.

7.2.4 Destructor

Plugin Version: 16.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The destructor.

7.2.5 Properties

7.2.6 Handle as Integer

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

Function: Internal object reference.

Notes: (Read and Write property)

7.2.7 Events

7.2.8 handleAppleEvent(theEvent as NSAppleEventDescriptorMBS, replyEvent as NSAppleEventDescriptorMBS)

Plugin Version: 12.5, Platform: macOS, Targets: .

Function: The event called when an event needs to be handled.

7.3 class NSAppleEventManagerMBS

7.3.1 class NSAppleEventManagerMBS

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

Function: The class for the Apple Event Manager.

Notes: Provides a mechanism for registering handler routines for specific types of Apple events and dispatching events to those handlers.

Cocoa provides built-in scriptability support that uses scriptability information supplied by an application to automatically convert Apple events into script command objects that perform the desired operation. However, some applications may want to perform more basic Apple event handling, in which an application registers handlers for the Apple events it can process, then calls on the Apple Event Manager to dispatch received Apple events to the appropriate handler. NSAppleEventManager supports these mechanisms by providing methods to register and remove handlers and to dispatch Apple events to the appropriate handler, if one exists. For related information, see *How Cocoa Applications Handle Apple Events* (on Apple Developer Website).

For information about the Apple Event Manager, see *Apple Event Manager Reference* and *Apple Events Programming Guide* (on Apple Developer Website).

Blog Entries

- [MBS Real Studio Plugins, version 12.5pr3](#)

7.3.2 Methods

7.3.3 `appleEventForSuspensionID(id as NSAppleEventManagerSuspensionIDMBS) as NSAppleEventDescriptorMBS`

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

Function: Given a suspensionID returned by an invocation of `suspendCurrentAppleEvent`, returns the descriptor for the event whose handling was suspended.

7.3.4 Constructor

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

Function: The constructor.

7.3.5 `currentAppleEvent` as `NSAppleEventDescriptorMBS`

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

Function: Returns the descriptor for `currentAppleEvent` if an Apple event is being handled on the current thread.

Notes: An Apple event is being handled on the current thread if a handler that was registered with `setEventHandler` is being messaged at this instant or `setCurrentAppleEventAndReplyEventWithSuspensionID` has just been invoked. Returns `nil` otherwise. The effects of mutating or retaining the returned descriptor are undefined, although it may be copied.

7.3.6 `currentReplyAppleEvent` as `NSAppleEventDescriptorMBS`

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

Function: Returns the corresponding reply event descriptor if an Apple event is being handled on the current thread.

Notes: An Apple event is being handled on the current thread if `currentAppleEvent` does not return `nil`. Returns `nil` otherwise. This descriptor, including any mutations, will be returned to the sender of the current event when all handling of the event has been completed, if the sender has requested a reply. The effects of retaining the descriptor are undefined; it may be copied, but mutations of the copy are not returned to the sender of the current event.

7.3.7 `NSAppleEventManagerWillProcessFirstEventNotification` as `string`

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

Function: One of the notification names you can use with `NSNotificationObserverMBS`.

Notes: Posted by `NSAppleEventManager` before it first dispatches an Apple event. Your application can use this notification to avoid registering any Apple event handlers until the first time at which they may be needed. The notification object is the `NSAppleEventManager`. This notification does not contain a `userInfo` dictionary.

7.3.8 `removeEventHandlerForEventClass(eventClass as string, eventID as string)`

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

Function: If an Apple event handler has been registered for the event specified by `eventClass` and `eventID`, removes it.

7.3.9 replyAppleEventForSuspensionID(id as NSAppleEventManagerSuspensionIDMBS) as NSAppleEventDescriptorMBS

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

Function: Given a nonzero suspensionID returned by an invocation of suspendCurrentAppleEvent, returns the corresponding reply event descriptor.

Notes: This descriptor, including any mutations, will be returned to the sender of the suspended event when handling of the event is resumed, if the sender has requested a reply. The effects of retaining the descriptor are undefined; it may be copied, but mutations of the copy are returned to the sender of the suspended event. replyAppleEventForSuspensionID may be invoked in any thread, not just the one in which the corresponding invocation of suspendCurrentAppleEvent occurred.

7.3.10 resumeWithSuspensionID(id as NSAppleEventManagerSuspensionIDMBS)

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

Function: Given a suspensionID returned by an invocation of suspendCurrentAppleEvent, signal that handling of the suspended event may now continue.

Notes: This may result in the immediate sending of the reply event to the sender of the suspended event, if the sender has requested a reply. If suspensionID has been used in a previous invocation of setCurrentAppleEventAndReplyEventWithSuspensionID the effects of that invocation are completely undone. Redundant invocations of resumeWithSuspensionID are ignored. Subsequent invocations of other NSAppleEventManager methods using the same suspension ID are invalid. resumeWithSuspensionID may be invoked in any thread, not just the one in which the corresponding invocation of suspendCurrentAppleEvent occurred.

7.3.11 setCurrentAppleEventAndReplyEventWithSuspensionID(id as NSAppleEventManagerSuspensionIDMBS)

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

Function: Given a nonzero suspensionID returned by an invocation of suspendCurrentAppleEvent, sets the values that will be returned by subsequent invocations of currentAppleEvent and currentReplyAppleEvent to be the event whose handling was suspended and its corresponding reply event, respectively.

Notes: Redundant invocations of setCurrentAppleEventAndReplyEventWithSuspensionID are ignored.

7.3.12 setEventHandler(handler as NSAppleEventHandlerMBS, eventClass as string, eventID as string)

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

Function: Registers the Apple event handler specified by handler for the event specified by eventClass and eventID.

7.3.13 suspendCurrentAppleEvent as NSAppleEventManagerSuspensionIDMBS

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

Function: Suspends the handling of the current event and returns an ID that must be used to resume the handling of the event if an Apple event is being handled on the current thread.

Notes: An Apple event is being handled on the current thread if currentAppleEvent does not return nil. Returns zero otherwise. The suspended event is no longer the current event after this method returns.

7.3.14 Properties

7.3.15 Handle as Integer

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

Function: Internal object reference.

Notes: (Read and Write property)

7.4 class NSAppleEventManagerSuspensionIDMBS

7.4.1 class NSAppleEventManagerSuspensionIDMBS

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

Function: Identifies an Apple event whose handling has been suspended.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Real Studio Plugins, version 12.5pr3](#)

7.4.2 Methods

7.4.3 Constructor

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

Function: The private constructor.

7.4.4 Properties

7.4.5 Handle as Integer

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

Function: Internal object reference.

Notes: (Read and Write property)

7.5 class NSAppleScriptMBS

7.5.1 class NSAppleScriptMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSAppleScriptMBS class provides the ability to load, compile, and execute scripts.

Example:

```
dim source as string = "tell Application ""iTunes"" to pause"
dim n as new NSAppleScriptMBS(source)
call n.execute
```

Notes: Your application may need the NSAppleEventsUsageDescription entry in info.plist to do any AppleScript usage:

https://developer.apple.com/documentation/bundleresources/information_property_list/nsappleeventsusagedescription

If you use hardened runtime, which you need for notarization, you may need the entitlement for Apple Events, too:

https://developer.apple.com/documentation/bundleresources/entitlements/com_apple_security_automation_apple-events

Important: You should access NSAppleScriptMBS only from the main thread.

This class provides applications with the ability to

- load a script from a URL or from a text string
- compile or execute a script or an individual Apple event
- obtain an NSAppleEventDescriptorMBS containing the reply from an executed script or event
- obtain an attributed string for a compiled script, suitable for display in a script editor
- obtain various kinds of information about any errors that may occur

Important: NSAppleScriptMBS provides the execute method so that you can send an Apple event to invoke a handler in a script. (In an AppleScript script, a handler is the equivalent of a function.) However, you cannot use this method to send Apple events to other applications.

When you create an instance of NSAppleScriptMBS object, you can use a URL or a folderitem to specify a script that can be in either text or compiled form, or you can supply the script as a string. Should an error occur when compiling or executing the script, several of the methods return a dictionary containing error

information. The keys for obtaining error information, such as `NSAppleScriptErrorMessage`, are described in the Constants section.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.4](#)
- [MBS Xojo Plugins, version 18.4pr7](#)
- [MBS Xojo / Real Studio Plugins, version 15.5pr1](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr7](#)
- [MBS Real Studio Plugins, version 13.0pr5](#)
- [MBS Real Studio Plugins, version 12.3pr16](#)
- [MBS Real Studio Plugins, version 12.1pr10](#)
- [MBS Real Studio Plugins, version 11.3pr4](#)
- [MBS Real Studio Plugins, version 11.2pr11](#)
- [MBS Real Studio Plugins, version 11.2pr9](#)

Xojo Developer Magazine

- [17.1, page 11: News](#)
- [16.6, page 9: News](#)

7.5.2 Methods

7.5.3 compile as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Compiles the receiver, if it is not already compiled.

Example:

```
dim source as string = "tell application ""iTunes"""+EndOfLine+"pause"+EndOfLine+"end tell"
dim n as new NSAppleScriptMBS(source)
```

```
dim error as Dictionary
if n.compile then
  MsgBox "OK"
else
  MsgBox error.Value(n.NSAppleScriptErrorMessage)
end if
```

Notes: error: Optional, on return, if an error occurs, an error information dictionary.

Return Value

Returns true for success or if the script was already compiled, false otherwise.

See also:

- 7.5.4 compile(byref error as dictionary) as boolean

258

7.5.4 compile(byref error as dictionary) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Compiles the receiver, if it is not already compiled.

Example:

```
dim source as string = "tell application ""iTunes"""+EndOfLine+"pause"+EndOfLine+"end tell"
dim n as new NSAppleScriptMBS(source)
```

```
dim error as Dictionary
if n.compile(error) then
  MsgBox "OK"
else
  MsgBox error.Value(n.NSAppleScriptErrorMessage)
end if
```

Notes: error: Optional, on return, if an error occurs, an error information dictionary.

Return Value

Returns true for success or if the script was already compiled, false otherwise.

See also:

- 7.5.3 compile as boolean

257

7.5.5 Constructor(file as folderitem, byref error as Dictionary)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a newly allocated script instance from the source identified by the passed folderitem.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("test.scpt")
dim error as Dictionary
dim n as new NSAppleScriptMBS(file, error)
```

```
if n.handle = 0 then
  dim err as Integer = error.lookup(n.NSAppleScriptErrorNumber,0)
  if err = -43 then
    MsgBox "File not found."
```

```

else
MsgBox "Some other error. "+str(n)
end if
else
MsgBox n.source
end if

```

Notes: file: A folderitem that locates a script, in either text or compiled form.
error: On return, if an error occurs, the error information dictionary.

Handle is zero in case of error.

See also:

- 7.5.6 Constructor(source as string) 259
- 7.5.7 Constructor(sourceLines() as string) 259
- 7.5.8 Constructor(URL as string, byref error as Dictionary) 260

7.5.6 Constructor(source as string)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a newly allocated script instance from the passed source.

Example:

```

dim source as string = "beep"
dim n as new NSAppleScriptMBS(source)

```

Notes: Handle is zero in case of error.

See also:

- 7.5.5 Constructor(file as folderitem, byref error as Dictionary) 258
- 7.5.7 Constructor(sourceLines() as string) 259
- 7.5.8 Constructor(URL as string, byref error as Dictionary) 260

7.5.7 Constructor(sourceLines() as string)

Plugin Version: 13.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a newly allocated script instance from the passed source.

Example:

```

dim lines() as string

lines.Append "property hello : ""Hallo Leute""
lines.Append "property just : ""Just a test""
lines.Append "display dialog hello"
lines.Append "return just"

dim a as new NSAppleScriptMBS(lines)

// compile
call a.Compile

// show names
dim names() as string = a.properties
MsgBox Join(names,EndOfLine)

// query value
MsgBox a.valueDescriptorForProperty("hello").stringValue

// change value
dim o as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithString("just a test")
call a.setValueDescriptorForProperty("hello", o)

// and query again
MsgBox a.valueDescriptorForProperty("hello").stringValue

```

Notes: Handle is zero in case of error.
See also:

- 7.5.5 Constructor(file as folderitem, byref error as Dictionary) 258
- 7.5.6 Constructor(source as string) 259
- 7.5.8 Constructor(URL as string, byref error as Dictionary) 260

7.5.8 Constructor(URL as string, byref error as Dictionary)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a newly allocated script instance from the source identified by the passed URL.

Notes: file: A folderitem that locates a script, in either text or compiled form.

error: On return, if an error occurs, the error information dictionary.

Handle is zero in case of error.
See also:

7.5. CLASS NSAPPLESCRIPTMBS	261
• 7.5.5 Constructor(file as folderitem, byref error as Dictionary)	258
• 7.5.6 Constructor(source as string)	259
• 7.5.7 Constructor(sourceLines() as string)	259

7.5.9 copy as NSAppleScriptMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a copy of this object.

Example:

```
dim source as string = "tell application ""iTunes""+EndOfLine+"pause"+EndOfLine+"end tell"
dim n as new NSAppleScriptMBS(source)
```

```
dim copy as NSAppleScriptMBS = n.copy
call copy.execute
```

7.5.10 DeterminePermissionToAutomateTarget(target as NSAppleEventDescriptorMBS, theAEEEventClass as string = "****", theAEEEventID as String = "****", askUserIfNeeded as boolean) as Integer

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Determines whether the current application is able to send an AppleEvent with the given event-Class and eventID to the application described as target.

Example:

```
dim target as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithBundleIdentifier("com.apple.Safari")
```

```
// we ask for all apple events
```

```
dim e as integer = NSAppleScriptMBS.DeterminePermissionToAutomateTarget(target, "****", "****", true)
```

```
MsgBox str(e)
```

Notes: Mac OS 10.14 and later impose additional requirements on applications when they send AppleEvents to other applications in order to insure that users are aware of and consent to allowing such control or information exchange. Generally this involves the user being prompted in a secure fashion the first time an application attempts to send an AppleEvent to another application. If the user consents then this application can send events to the target. If the user does not consent then any future attempts to send AppleEvents will result in a failure with errAEEEventNotPermitted (-1743) being returned.

Certain AppleEvents are allowed to be sent without prompting the user. Pass `typeWildcard` for the `eventClass` and `eventID` to determine if every event is allowed to be sent from this application to the target.

Applications can determine, without sending an AppleEvent to a target application, whether they are allowed to send AppleEvents to the target with this function. If `askUserIfNeeded` is true, and this application does not yet have permission to send AppleEvents to the target, then the user will be asked if permission can be granted; if `askUserIfNeeded` is false and permission has not been granted, then `errAEEEventWouldRequireUserConsent` (-1744) will be returned.

The target must refer to an already running application.

If the current application is permitted to send the given AppleEvent to the target, then `noErr` (0) will be returned. If the current application is not permitted to send the event, `errAEEEventNotPermitted` (-1743) will be returned. If the target application is not running, then `procNotFound` (-600) will be returned. If `askUserIfNeeded` is false, and this application is not yet permitted to send AppleEvents to the target, then `errAEEEventWouldRequireUserConsent` (-1744) will be returned.

Mac OS X threading: Thread safe since version 10.14. Do not call this function on your main thread because it may take arbitrarily long to return if the user needs to be prompted for consent.

target: The address descriptor. Before calling `AEDeterminePermissionToAutomateTarget`, you set the descriptor to identify the target application for the Apple event. The target address descriptor must refer to a running application. If the target application is on another machine, then Remote AppleEvents must be enabled on that machine for the user.

theAEEEventClass: The event class of the Apple event to determine permission for.

theAEEEventID: The event ID of the Apple event to determine permission for.

askUserIfNeeded: a Boolean; if true, and if this application does not yet have permission to send events to the target application, then prompt the user to obtain permission. If false, do not prompt the user.

Plugin returns -1 if the function is not available, e.g. older MacOS versions.

7.5.11 execute as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Executes the receiver, compiling it first if it is not already compiled.

Example:

```
dim source as string = "tell Application ""iTunes"" to pause"
dim n as new NSAppleScriptMBS(source)
call n.execute
```

Notes: error: Optional, on return, if an error occurs, an error information dictionary.

Returns the result of executing the event, or nil if an error occurs.
Any changes to property values caused by executing the script do not persist.
See also:

- 7.5.12 execute(byref error as dictionary) as NSAppleEventDescriptorMBS

7.5.12 execute(byref error as dictionary) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Executes the receiver, compiling it first if it is not already compiled.

Example:

```
dim source as string = "tello Application ""iTunes"" to play"
dim error as dictionary
dim n as new NSAppleScriptMBS(source)
dim p as NSAppleEventDescriptorMBS = n.execute(Error)
if p <> nil then
  MsgBox "OK"
else
  MsgBox error.Lookup(n.NSAppleScriptErrorMessage,"unknown error")
end if
```

Notes: error: Optional, on return, if an error occurs, an error information dictionary.
Returns the result of executing the event, or nil if an error occurs.
Any changes to property values caused by executing the script do not persist.
See also:

- 7.5.11 execute as NSAppleEventDescriptorMBS

7.5.13 executeAppleEvent(event as NSAppleEventDescriptorMBS, byref error as dictionary) as NSAppleEventDescriptorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Executes an Apple event in the context of the receiver, as a means of allowing the application to invoke a handler in the script.

Notes: event: The Apple event to execute.

error: On return, if an error occurs, an error information dictionary.

Returns the result of executing the event, or nil if an error occurs.

Compiles the receiver before executing it if it is not already compiled.

Important: You cannot use this method to send Apple events to other applications.

7.5.14 executeSubroutine(Name as String, parameters() as NSAppleEventDescriptorMBS, byref error as dictionary) as NSAppleEventDescriptorMBS

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Calls a subroutine in a script.

Example:

```
// our script

Dim CodeLines() as string

CodeLines.Append "on Add(Name1, Name2)"
CodeLines.Append "return Name1 & "" "" & Name2"
CodeLines.Append "end Add"

// now compile it
Dim a as new NSAppleScriptMBS(CodeLines)
Dim error as dictionary
Dim CompileOkay As Boolean = a.Compile(error)

if CompileOkay then

// script name and parameters
Dim ScriptFuncName As String = "Add"

Dim ScriptParams() As NSAppleEventDescriptorMBS
ScriptParams.Append NSAppleEventDescriptorMBS.descriptorWithString("Hello")
ScriptParams.Append NSAppleEventDescriptorMBS.descriptorWithString("World")

// bow run it
Dim p as NSAppleEventDescriptorMBS

p = a.executeSubroutine(ScriptFuncName, ScriptParams, Error)
if error = nil then
// show result
Dim ScriptResult As String = p.stringValue

MsgBox "ScriptResult:" + EndOfLine + EndOfLine + ScriptResult
else
MsgBox "Error running script." + _
EndOfLine + EndOfLine + _
error.Lookup(a.NSAppleScriptErrorMessage, "Unknown error") + _
EndOfLine + EndOfLine + _
```

```

error.Lookup(a.NSAppleScriptErrorBriefMessage, "Unknown error") + _
EndOfLine + EndOfLine + _
error.Lookup(a.NSAppleScriptErrorNumber, "Unknown error")
end if
else
// Compile Error
MsgBox "Error loading script." + _
EndOfLine + EndOfLine + _
error.Lookup(a.NSAppleScriptErrorMessage, "Unknown error") + _
EndOfLine + EndOfLine + _
error.Lookup(a.NSAppleScriptErrorBriefMessage, "Unknown error") + _
EndOfLine + EndOfLine + _
error.Lookup(a.NSAppleScriptErrorNumber, "Unknown error")
end if

```

Notes: Similar to `executeAppleEvent`, but creates the apple event for you.

Name: The name of the subroutine to execute.

parameters: The parameters for the subroutine.

error: On return, if an error occurs, an error information dictionary.

Returns the result of executing the event, or nil if an error occurs.

Compiles the receiver before executing it if it is not already compiled.

Important: You cannot use this method to send Apple events to other applications.

7.5.15 NSAppleScriptErrorAppName as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys in the error dictionary.

Notes: Value for this key is a string that specifies the name of the application that generated the error.

7.5.16 NSAppleScriptErrorBriefMessage as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys in the error dictionary.

Example:

```

dim source as string = "tell application ""iTunes"" + EndOfLine + "pause" + EndOfLine + "end if"
dim n as new NSAppleScriptMBS(source)

```

```

dim error as Dictionary
if not n.compile(error) then
// shows error about missing tell where it found an if.
MsgBox error.Value(n.NSAppleScriptErrorBriefMessage)
end if

```

Notes: Value for this key is a string that provides a brief description of the error.

7.5.17 NSAppleScriptErrorMessage as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys in the error dictionary.

Example:

```

dim source as string = "tell application ""iTunes"""+EndOfLine+"pause"+EndOfLine+"end if"
dim n as new NSAppleScriptMBS(source)

```

```

dim error as Dictionary
if n.compile(error) then
MsgBox "OK"
else
// shows error about missing tell where it found an if.
MsgBox error.Value(n.NSAppleScriptErrorMessage)
end if

```

Notes: Value for this key is a NSRangeMBS object.

7.5.18 NSAppleScriptErrorNumber as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys in the error dictionary.

Example:

```

dim source as string = "tell application ""iTunes"""+EndOfLine+"pause"+EndOfLine+"end if"
dim n as new NSAppleScriptMBS(source)

```

```

dim error as Dictionary
if not n.compile(error) then
// error about missing tell where it found an if.
MsgBox error.Value(n.NSAppleScriptErrorNumber) // shows -2741

```

end if

Notes: Value for this key is a number that specifies the error number.

7.5.19 NSAppleScriptErrorRange as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys in the error dictionary.

Example:

```
dim source as string = "tell application ""iTunes"""+EndOfLine+"pause"+EndOfLine+"end if"
dim n as new NSAppleScriptMBS(source)

dim error as Dictionary
if not n.compile(error) then
  // error about missing tell where it found an if.
  dim r as NSRangeMBS = error.Value(n.NSAppleScriptErrorRange)
  MsgBox r.String // { 36,2 } , the position of the if
end if
```

Notes: Value for this key is a string that supplies a detailed description of the error condition.

7.5.20 properties as string()

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries the names of all properties in the script.

Example:

```
dim s as String
dim a as NSAppleScriptMBS
dim i,c,cc as Integer
dim z,t as String

s=s+"property hello : ""Hallo Leute"""+chr(13)
s=s+"property just : ""Just a test"""+chr(13)
s=s+"display dialog hello"+chr(13)
s=s+"return just"+chr(13)

MsgBox "The script:"+chr(13)+s
a=new NSAppleScriptMBS(s)
```

```
// compile
call a.Compile

// show names
dim names() as string = a.properties
MsgBox Join(names,EndOfLine)

// query value
MsgBox a.valueDescriptorForProperty("hello").stringValue

// change value
dim o as NSAppleEventDescriptorMBS = NSAppleEventDescriptorMBS.descriptorWithString("just a test")
call a.setValueDescriptorForProperty("hello", o)

// and query again
MsgBox a.valueDescriptorForProperty("hello").stringValue
```

7.5.21 setValueDescriptorForProperty(propertyName as string, value as NSAppleEventDescriptorMBS) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets a property value.

7.5.22 valueDescriptorForProperty(propertyName as string) as NSAppleEventDescriptorMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries a property value.

7.5.23 Properties

7.5.24 Handle as Integer

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal reference to the NSAppleScript object.

Notes: (Read and Write property)

7.5.25 isCompiled as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value that indicates whether the receiver's script has been compiled.

Example:

```
dim source as string = "tell Application ""iTunes"" to play"
dim n as new NSAppleScriptMBS(source)
```

```
MsgBox "isCompiled: "+str(n.isCompiled)
```

```
call n.compile
```

```
MsgBox "isCompiled: "+str(n.isCompiled)
```

Notes: (Read only property)

7.5.26 richTextSource as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the syntax-highlighted source code of the receiver if the receiver has been compiled and its source code is available.

Example:

```
// init with some source
dim source as string = "tell Application ""iTunes"" to play"
dim n as new NSAppleScriptMBS(source)

// compile
call n.compile

// format text
dim richtext as NSAttributedStringMBS = n.richTextSource

if richtext = nil then
  MsgBox "Failed to format source."
else
  // write to RTF file
  dim file as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
  dim b as BinaryStream = file.CreateBinaryFile("")

  b.Write richtext.RTF
  b.close
```

```
file.Launch  
end if
```

Notes: Returns nil otherwise. It is possible for an instance of NSAppleScript that has been instantiated with Constructor to be a script for which the source code is not available, but is nonetheless executable. (Read only property)

7.5.27 source as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the script source for the receiver.

Notes: Returns the script source code of the receiver if it is available, "" otherwise.

It is possible for an NSAppleScript that has been instantiated with Constructor to be a script for which the source code is not available but is nonetheless executable. (Read only property)

Chapter 8

Cocoa

8.1 class NSAlertMBS

8.1.1 class NSAlertMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The class for a standard Cocoa alert.

Example:

```
Dim a As New NSAlertMBS
```

```
a.alertStyle = NSAlertMBS.NSCriticalAlertStyle  
a.messageText = "Should be do it?"
```

```
Dim actionButton As NSButtonMBS = a.addButtonWithTitle("Do it!")  
actionButton.hasDestructiveAction = True  
actionButton.keyEquivalent = encodings.UTF8.Chr(13) // enter
```

```
Dim cancelButton As NSButtonMBS = a.addButtonWithTitle("Cancel")  
cancelButton.keyEquivalent = encodings.UTF8.Chr(27) // esc
```

```
Dim e As Integer = a.runModal
```

```
If e = a.NSAlertFirstButtonReturn Then  
  MessageBox "Do it!"  
End If
```

Blog Entries

- [MBS Xojo Plugins, version 22.5pr1](#)

- [News from the MBS Xojo Plugins Version 22.1](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.1](#)
- [MBS Xojo Plugins, version 22.1pr6](#)
- [Text alignment for alerts](#)
- [MBS Xojo / Real Studio plug-ins version 16.2](#)
- [MBS Xojo / Real Studio Plugins, version 16.2pr2](#)
- [MBS Xojo / Real Studio plug-ins in version 14.2](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr1](#)

Xojo Developer Magazine

- [20.3, page 10: News](#)
- [12.4, page 9: News](#)

8.1.2 Methods

8.1.3 addButtonWithTitle(title as string) as Variant

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Customize the buttons in the alert panel.

Example:

```
// make dialog
dim a as NSAlertMBS = NSAlertMBS.alertWithMessageText("Hello World", "First Button", "Second Button")

// add button
dim thirdButton as NSButtonMBS = a.addButtonWithTitle("Third Button")

// and show dialog
call a.runModal
```

Notes: Buttons are added from right to left (for left to right languages). Returns NSButtonMBS object.

8.1.4 alertWithError(error as NSErrorMBS) as NSAlertMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Given an NSError, create an NSAlert that can be used to present the error to the user.

Notes: The error's localized description, recovery suggestion, and recovery options will be used to set the alert's message text, informative text, and button titles, respectively.

8.1.5 alertWithMessageText(MessageText as string, defaultButton as string = "", alternateButton as string = "", otherButton as string = "", informativeText as string = "") as NSAlertMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new alert with given property values.

8.1.6 beginSheetModalForWindow(win as DesktopWindow)

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

Function: Begins a sheet on the document window.

Notes: If the alert has an alertStyle of NSCriticalAlertStyle, it will be shown as a "critical" sheet; it will otherwise be presented as a normal sheet.

Calls later SheetDidEnd event with the result.

Please keep a reference to the dialog object alive to avoid trouble.
e.g. store reference in parent window, global property or app property.
See also:

- 8.1.7 beginSheetModalForWindow(win as NSWindowMBS) 273
- 8.1.8 beginSheetModalForWindow(win as window) 274

8.1.7 beginSheetModalForWindow(win as NSWindowMBS)

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Begins a sheet on the document window.

Notes: If the alert has an alertStyle of NSCriticalAlertStyle, it will be shown as a "critical" sheet; it will otherwise be presented as a normal sheet.

Calls later SheetDidEnd event with the result.

Please keep a reference to the dialog object alive to avoid trouble.
e.g. store reference in parent window, global property or app property.
See also:

- 8.1.6 `beginSheetModalForWindow(win as DesktopWindow)` 273
- 8.1.8 `beginSheetModalForWindow(win as window)` 274

8.1.8 `beginSheetModalForWindow(win as window)`

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Begins a sheet on the document window.

Notes: If the alert has an `alertStyle` of `NSCriticalAlertStyle`, it will be shown as a "critical" sheet; it will otherwise be presented as a normal sheet.

Calls later `SheetDidEnd` event with the result.

Please keep a reference to the dialog object alive to avoid trouble.
e.g. store reference in parent window, global property or app property.
See also:

- 8.1.6 `beginSheetModalForWindow(win as DesktopWindow)` 273
- 8.1.7 `beginSheetModalForWindow(win as NSWindowMBS)` 273

8.1.9 `buttons as Variant()`

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Get the buttons, where the rightmost button is at index 0.

Example:

```
dim a as NSAlertMBS = NSAlertMBS.alertWithMessageText("Hello World", "First Button", "Second Button")
dim buttons() as Variant = a.buttons

for each b as NSButtonMBS in buttons
  MsgBox b.title
next
```

Notes: Returns `NSButtonMBS` array.

8.1.10 close

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Closes the alert sheet.

Notes: The SheetDidEnd event will not run.

8.1.11 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The constructor for an empty alert.

Notes: Use properties to configure the dialog.

8.1.12 Destructor

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The destructor.

8.1.13 layout

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Performs the layout for the dialog.

Notes: Can be used to indicate that the alert panel should do immediate layout, overriding the default behavior of laying out lazily just before showing panel. You should only call this method if you want to do your own custom layout after it returns. You should call this method only after you have finished with NSAlert customization, including setting message and informative text, and adding buttons and an accessory view if needed. You can make layout changes after this method returns, in particular to adjust the frame of an accessory view. Note that the standard layout of the alert may change in the future, so layout customization should be done with caution.

8.1.14 runModal as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Run the alert as an application-modal panel and return the result.

Example:

[Dim a As New NSAlertMBS](#)

```
a.messageText = "Should be do it?"
```

```
// OK button is added by default
Dim e As Integer = a.runModal
```

8.1.15 SetTextAlignment(value as Integer)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Changes text alignment for dialog to left text alignment.

Notes: on macOS Big Sur and later, the default alignment is center.

If you show multiple lines of text, it may be easier to read with left alignment.

This calls layout method to layout the dialog.

See NSParagraphStyleMBS or NSTextMBS class for the alignment constants.

8.1.16 Properties

8.1.17 accessoryView as NSViewMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The accessory view displayed in the alert panel.

Notes: By default, the accessory view is positioned below the informative text and the suppression button (if any) and above the alert buttons, left-aligned with the informative text. If you want to customize the location of the accessory view, you must first call layout method.

(Read and Write property)

8.1.18 alertStyle as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The alert style.

Notes: Value can be:

NSWarningAlertStyle	0	Warning
NSInformationalAlertStyle	1	Information
NSCriticalAlertStyle	2	Critical Error

(Read and Write property)

8.1.19 helpAnchor as String

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The help anchor to use.

Notes: (Read and Write property)

8.1.20 icon as NSImageMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Custom icon for dialog.

Notes: By default uses the image named NSApplicationIcon.

(Read and Write property)

8.1.21 informativeText as String

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The informative text.

Notes: (Read and Write property)

8.1.22 messageText as String

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The message text.

Notes: (Read and Write property)

8.1.23 showsHelp as Boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Whether to show help button.

Notes: True adds a help button to the alert panel. When the help button is pressed, the delegate is first consulted. If the event does not implement ShowHelp event or returns false, then NSHelpManager.openHelpAnchor is called with a nil book and the anchor specified by HelpAnchor, if any. An exception will be raised if the delegate returns false and there is no help anchor set.

(Read and Write property)

8.1.24 ShowsSuppressionButton as Boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Indicates whether or not the alert should contain a suppression checkbox.

Notes: The default is false. This checkbox is typically used to give the user an option to not show this alert again. If shown, the suppression button will have a default localized title similar to @"Do not show this message again". You can customize this title using alert.suppressionButton.Title. When the alert is dismissed, you can get the state of the suppression button, using [[alert suppressionButton] state] and store the result in user defaults, for example. This setting can then be checked before showing the alert again. By default, the suppression button is positioned below the informative text, and above the accessory view (if any) and the alert buttons, and left-aligned with the informative text. However do not count on the placement of this button, since it might be moved if the alert panel user interface is changed in the future. If you need a checkbox for purposes other than suppression text, it is recommended you create your own using an accessory view.

(Read and Write property)

8.1.25 suppressionButton as Variant

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Returns a suppression button which may be customized, including the title and the initial state.

Example:

```
// make dialog
dim a as NSAlertMBS = NSAlertMBS.alertWithMessageText("Hello World", "First Button", "Second Button")

// get button
dim suppressionButton as NSButtonMBS = a.suppressionButton

// change title
suppressionButton.title = "Hello World Button"

// want to show it
a.ShowsSuppressionButton = true

// and show dialog
call a.runModal
```

Notes: You can also use this method to get the state of the button after the alert is dismissed, which may

be stored in user defaults and checked before showing the alert again. In order to show the suppression button in the alert panel, you must set ShowsSuppressionButton to true.

Returns NSButtonMBS object.

(Read only property)

8.1.26 TimedOut as Boolean

Plugin Version: 16.2, Platform: macOS, Targets: Desktop only.

Function: Whether we got a timeout.

Notes: This is set to true when a timeout occurred.

(Read only property)

8.1.27 timeOut as Double

Plugin Version: 16.2, Platform: macOS, Targets: Desktop only.

Function: Timeout for the dialog.

Notes: Set this to the number of seconds after which the dialog should close.

(Read and Write property)

8.1.28 window as Variant

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Returns the application-modal panel or the document-modal sheet corresponding to this alert.

Notes: (Read only property)

8.1.29 Events

8.1.30 SheetDidEnd(returnCode as Integer)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The sheet did finish.

8.1.31 ShowHelp as boolean

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Show custom help.

Notes: See ShowHelp property.

8.1.32 Constants

Return codes

Constant	Value	Description
NSAlertFirstButtonReturn	1000	First (rightmost) button
NSAlertSecondButtonReturn	1001	Second button.
NSAlertThirdButtonReturn	1002	Third button.

Alert styles.

Constant	Value	Description
NSCriticalAlertStyle	2	Critical Error
NSInformationalAlertStyle	1	Informational Alert
NSWarningAlertStyle	0	Warning Alert (Default style)

8.2 class NSApplicationDelegateMBS

8.2.1 class NSApplicationDelegateMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The class for an Cocoa application delegate.

Notes: Please install in app Constructor as App.Open event may be too late.

Using this class you can get application related events before (!) the app class gets it. And of course more events than just the ones the app class have.

Only for Cocoa desktop targets.

In general the plugin calls first the event. Depending on the result it may pass the event to the Xojo application delegate. If you have no code in the plugin event, everything just passes through and you should not see a difference.

The plugin application delegate is installed with the Constructor and uninstalled in the Destructor.

The original delegate from Xojo is preserved and all messages are forwarded to it. Also when this object is destroyed, the old delegate is restored.

Blog Entries

- [The order of events in Xojo](#)
- [MBS Xojo Plugins, version 21.2pr2](#)
- [MBS Xojo / Real Studio Plugins, version 14.3pr9](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr1](#)
- [MonkeyBread Software Releases the MBS Real Studio plug-ins in version 11.3](#)
- [MBS Real Studio Plugins, version 11.2pr12](#)
- [Lion features for Real Studio](#)
- [Lion arrived](#)
- [MBS Real Studio Plugins, version 11.2pr9](#)
- [Dock Menu for Cocoa in Real Studio](#)

8.2.2 Events

8.2.3 `applicationDidBecomeActive(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately after the application becomes active.

8.2.4 `applicationDidChangeScreenParameters(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center when the configuration of the displays attached to the computer is changed (either programmatically or when the user changes settings in the Displays control panel).

8.2.5 `applicationDidDecodeRestorableState(coder as NSCoderMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: This event gives you the chance to restore your own state.

8.2.6 `applicationDidFailToRegisterForRemoteNotificationsWithError(error as NSErrorMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Application failed to register for remote notifications.

Notes: Check your provisioning profile for entitlements.

Available on Mac OS X 10.7 or newer.

8.2.7 `applicationDidFinishLaunching(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center after the application has been launched and initialized but before it has received its first event.

Notes: Delegates can implement this method to perform further initialization. This method is called after

the application's main run loop has been started but before it has processed any events. If the application was launched by the user opening a file, the delegate's `applicationOpenFile` method is called before this method. If you want to perform initialization before any files are opened, implement the `applicationWillFinishLaunching` method in your delegate, which is called before `applicationOpenFile`.)

8.2.8 `applicationDidHide(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately after the application is hidden.

8.2.9 `applicationDidReceiveRemoteNotification(userInfo as Dictionary)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: A notification was received.

Notes: The dictionary contains payload like this:

key "aps" contains another dictionary. This second dictionary contains keys like "alert", "badge" and "sound".

Available on Mac OS X 10.7 or newer.

8.2.10 `applicationDidRegisterForRemoteNotificationsWithDeviceToken(deviceToken as memoryblock)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Application registered for remote notifications.

Notes: Tell your server the device token ID and use it in your push notifications.

Distinguish between iOS and Mac versions of your app!

Available on Mac OS X 10.7 or newer.

8.2.11 `applicationDidResignActive(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately after the application is deactivated.

8.2.12 applicationDidUnhide(Notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately after the application is made visible.

8.2.13 applicationDidUpdate(Notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately after the application object updates its windows.

8.2.14 applicationDockMenu as NSMenuMBS

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Allows the delegate to supply a dock menu for the application dynamically.

Notes: Return the menu to display in the dock. Or nil for having no/default menu.

8.2.15 applicationOpenFile(filename as string) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate to open a single file.

Notes: filename: The path of the file to open.

Return yes if the file was successfully opened or false if it was not.

Sent directly by application to the delegate. The method should open the file filename, returning true if the file is successfully opened, and false otherwise. If the user started up the application by double-clicking a file, the delegate receives the applicationOpenFile message before receiving applicationDidFinishLaunching. (applicationWillFinishLaunching is sent before applicationOpenFile.)

If you return false, the plugin will pass the event to the default Xojo runtime application delegate, so the OpenDocument event can fire.

8.2.16 applicationOpenFiles(filenamees() as string) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate to open multiple files.

Notes: sender: The application object associated with the delegate.

filenamees: An array of strings containing the names of the files to open..

Identical to applicationOpenFile except that the receiver opens multiple files corresponding to the file names in the filenamees array. Delegates should invoke the replyToOpenOrPrint method upon success or failure, or when the user cancels the operation.

If you add code to this event, it is possible that OpenDocument event in Xojo does not fire.

8.2.17 applicationOpenFileWithoutUI(filename as string) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate to open a file programmatically.

Notes: filename: The name of the file to open.

Return true if the file was successfully opened or false if it was not.

Sent directly by sender to the delegate to request that the file filename be opened as a linked file. The method should open the file without bringing up its application's user interface—that is, work with the file is under programmatic control of sender, rather than under keyboard control of the user.

If you add code to this event, it is possible that OpenDocument event in Xojo does not fire.

8.2.18 applicationOpenTempFile(filename as string) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate to open a temporary file.

Notes: filename: The name of the temporary file to open.

True if the file was successfully opened or false if it was not.

Sent directly by application to the delegate. The method should attempt to open the file filename, returning true if the file is successfully opened, and false otherwise.

By design, a file opened through this method is assumed to be temporary—it's the application's responsibility to remove the file at the appropriate time.

If you add code to this event, it is possible that OpenDocument event in Xojo does not fire.

8.2.19 `applicationOpenUntitledFile` as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate to open an untitled file.

Notes: Return true if the file was successfully opened or false if it was not.

Sent directly by application to the delegate to request that a new, untitled file be opened.

If you return false, the plugin will pass the event to the default Xojo runtime application delegate.

If you add code to this event, it is possible that OpenDocument event in Xojo does not fire.

8.2.20 `applicationPrintFile(filename as string)` as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent when the user starts up the application on the command line with the `-NSPrint` option.

Notes: filename: The name of the file to print.

Returns true if the file was successfully printed or false if it was not.

This message is sent directly by application to the delegate. The application terminates (using the `terminate` method) after this method returns.

If at all possible, this method should print the file without displaying the user interface. For example, if you pass the `-NSPrint` option to the TextEdit application, TextEdit assumes you want to print the entire contents of the specified file. However, if the application opens more complex documents, you may want to display a panel that lets the user choose exactly what they want to print.

8.2.21 `applicationPrintFiles(fileName() as string, printSettings as dictionary, showPrintPanels as boolean)` as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Prints a group of files.

Notes: fileNames: An array of strings, each of which contains the name of a file to print.

printSettings: A dictionary containing NSPrintInfo-compatible print job attributes.

showPrintPanels: A Boolean that specifies whether the print panel should be displayed for each file printed. Print progress indicators will be presented even if this value is false.

Return a constant indicating whether printing was successful. For a list of possible values, see NSPrinting* constants.

Return NSPrintingReplyLater if the result of printing cannot be returned immediately, for example, if printing will cause the presentation of a sheet. If your method returns NSPrintingReplyLater it must always invoke the NSApplicationMBS method replyToOpenOrPrint when the entire print operation has been completed, successfully or not.

8.2.22 applicationShouldHandleReopen(hasVisibleWindows as boolean) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the application to the delegate prior to default behavior to reopen (rapp) AppleEvents.

Notes: flag: Indicates whether the NSApplication object found any visible windows in your application. You can use this value as an indication of whether the application would do anything if you return true.

Return true if you want the application to perform its normal tasks or false if you want the application to do nothing.

These events are sent whenever the Finder reactivates an already running application because someone double-clicked it again or used the dock to activate it.

By default the Application Kit will handle this event by checking whether there are any visible NSWindow (not NSPanel) objects, and, if there are none, it goes through the standard untitled document creation (the same as it does if application is launched without any document to open). For most document-based applications, an untitled document will be created.

The application delegate will also get a chance to respond to the normal untitled document delegate methods. If you implement this method in your application delegate, it will be called before any of the default behavior happens. If you return true, then NSApplication will proceed as normal. If you return false, then NSApplication will do nothing. So, you can either implement this method with a version that does nothing, and return false if you do not want anything to happen at all (not recommended), or you can implement this method, handle the event yourself in some custom way, and return false.

Miniaturized windows, windows in the Dock, are considered visible by this method, and cause flag to return true, despite the fact that miniaturized windows return false when sent an isVisible message.

Having no code in the event will tell the plugin to return true.

8.2.23 applicationShouldOpenUntitledFile as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked immediately before opening an untitled file.

Notes: Return true if the application should open a new untitled file or false if it should not.

Use this method to decide whether the application should open a new, untitled file. Note that applicationOpenUntitledFile is invoked if this method returns true.

If you return false here, the NewDocument event in Xojo may not fire.

Having no code in this event is the same as returning true.

8.2.24 applicationShouldTerminate as Integer

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent to notify the delegate that the application is about to terminate..

Notes: One of the values defined in NSTerminate* constants indicating whether the application should terminate.

This method is called after the application's Quit menu item has been selected, or after the terminate method has been called. Generally, you should return NSTerminateNow to allow the termination to complete, but you can cancel the termination process or delay it somewhat as needed. For example, you might delay termination to finish processing some critical data but then terminate the application as soon as you are done by calling the replyToApplicationShouldTerminate method.

8.2.25 applicationShouldTerminateAfterLastWindowClosed as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked when the user closes the last window the application has open.

Notes: Return false if the application should not be terminated when its last window is closed; otherwise,

true to terminate the application.

The application sends this message to your delegate when the application's last window is closed. It sends this message regardless of whether there are still panels open. (A panel in this case is defined as being an instance of NSPanel or one of its subclasses.)

If your implementation returns false, control returns to the main event loop and the application is not terminated. If you return true, your delegate's `applicationShouldTerminate` method is subsequently invoked to confirm that the application should be terminated.

Having no code in this event is the same as returning false.

8.2.26 `applicationWillBecomeActive(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately before the application becomes active.

8.2.27 `applicationWillEncodeRestorableState(coder as NSCoderMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Event called to give you chance to encode any additional state into the NSCoder.

Notes: If the restorable state managed by the delegate changes, you must call `NSApplicationMBS.invalidateRestorableState` so that it will be re-encoded.

8.2.28 `applicationWillFinishLaunching(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately before the application object is initialized.

8.2.29 `applicationWillHide(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately before the application is hidden.

8.2.30 `applicationWillPresentError(error as NSErrorMBS) as NSErrorMBS`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent to the delegate before the specified application presents an error message to the user.

Notes: error: The error object that is used to construct the error message. Your implementation of this method can return a new NSError object or the same one in this parameter.

Return the error object to display.

You can implement this delegate method to customize the presentation of any error presented by your application, as long as no code in your application overrides either of the NSResponder methods `presentError` in a way that prevents errors from being passed down the responder chain to the application object.

Your implementation of this delegate method can examine error and, if its localized description or recovery information is unhelpfully generic, return an error object with specific localized text that is more suitable for presentation in alert sheets and dialogs. If you do this, always use the domain and error code of the NSError object to distinguish between errors whose presentation you want to customize and those you do not. Don't make decisions based on the localized description, recovery suggestion, or recovery options because parsing localized text is problematic. If you decide not to customize the error presentation, just return the passed-in error object.

If you have no code in this event or you return nil, the plugin passes the given error back to the Cocoa runtime.

8.2.31 `applicationWillResignActive(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately before the application is deactivated.

8.2.32 `applicationWillTerminate(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately before the application terminates.

8.2.33 `applicationWillUnhide(Notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately after the application is unhidden.

8.2.34 applicationWillUpdate(Notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Sent by the default notification center immediately before the application object updates its windows.

8.2.35 restoreWindowWithIdentifier(identifier as string, state as NSCoderMBS, byref resultWindow as Variant, byref error as NSErrorMBS) as boolean

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

Function: Invoked to request that a window be restored.

Notes: If you plan to use this event, please initialize the NSApplicationDelegateMBS subclass in app.constructor.

identifier: The unique interface item identifier string that was previously associated with the window. Use this string to determine which window to create.

state: A NSCoderMBS object containing the window state information. This coder object contains the combined restorable state of the window, which can include the state of the window, its delegate, window controller, and document object. You can use this state to determine which window to create.

Pass back with the parameters:

The window that was created or nil if the window could not be created.

An error object if the window was not recognized or could not be created for whatever reason; otherwise, specify nil. In OS X 10.7, the error parameter is ignored.

Return true if the window was restored; otherwise false.

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 use 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.

The receiver is application is passed the identifier of the window, which allows it 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 instance 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.

Available in OS X v10.7 and later.

The plugin implements this method for `NSApplication` and forwards the message to this event. If you return true, please set either `error` or `resultWindow` values. `resultWindow` must be an `NSWindowMBS` or a window object.

8.2.36 `userDidAcceptCloudKitShareWithMetadata(metadata as Variant)`

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

Function: Tells you that the app has access to shared information in CloudKit.

Example:

// sample implementation of the event:

```
Sub userDidAcceptCloudKitShareWithMetadata(metadata as Variant) Handles userDidAcceptCloudKit-
ShareWithMetadata
Dim c As CKShareMetadataMBS = metadata

Message "User did accept CloudKit share: "+c.containerIdentifier
End Sub
```

Notes: `metadata` is a `CKShareMetadataMBS` class.

`metadata`: Information about the CloudKit data that is available to the app. Use this object to retrieve information about the `CKShare` object and the associated records that are available.

Use this method to respond to a CloudKit Sharing invitation. In your implementation, accept the share by scheduling a `CKAcceptSharesOperationMBS` object that contains the provided `metadata` object. After the user accepts the share, you can begin fetching records and incorporating the resulting data into your app. The system launches the app, as necessary, before calling this method.

8.2.37 Constants

Print Reply Constants

Constant	Value	Description
NSPrintingCancelled	0	Printing was cancelled.
NSPrintingFailure	3	Printing failed.
NSPrintingReplyLater	2	The result of printing cannot be returned immediately, for example, if printing will cause the presentation of a sheet. If your method returns <code>NSPrintingReplyLater</code> it must always invoke <code>replyToOpenOrPrint</code> when the entire print operation has been completed, successfully or not.
NSPrintingSuccess	1	Printing was successful.

Terminate Reply Constants

Constant	Value	Description
NSTerminateCancel	0	The application should not be terminated.
NSTerminateLater	2	It may be OK to proceed with termination later. Returning this value causes Cocoa to run the run loop in the <code>NSModalPanelRunLoopMode</code> until your application subsequently calls <code>replyToApplicationShouldTerminate</code> with the value <code>true</code> or <code>false</code> . This return value is for delegates that need to provide document modal alerts (sheets) in order to decide whether to quit.
NSTerminateNow	1	It is OK to proceed with termination.

8.3 class NSApplicationMBS

8.3.1 class NSApplicationMBS

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: The cocoa application class.

Example:

```
dim a as NSApplicationMBS = NSApplicationMBS.sharedApplication
```

```
// set a custom picture
```

```
dim p as Picture = LogoMBS(500)
```

```
dim n as new NSImageMBS(p)
```

```
a.applicationIconImage = n
```

```
// restore
```

```
'a.applicationIconImage = nil
```

Notes: The plugin only implements a small subset of what's available in Cocoa. If you miss something, please send us an email.

You can get an instance of this class using one of three ways:

- `app.NSApplicationMBS` function
- `new NSApplicationMBS`
- `NSApplicationMBS.sharedInstance`

Blog Entries

- [MBS Xojo Plugins, version 20.5pr1](#)
- [MBS Xojo Plugins, version 18.2pr3](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 17.5](#)
- [MBS Xojo Plugins, version 17.5pr9](#)
- [MBS Xojo / Real Studio Plugins, version 16.5pr4](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr7](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr1](#)
- [Working on Mountain Lion Support](#)

- [Lion features for Real Studio](#)
- [Lion arrived](#)

Xojo Developer Magazine

- [16.1, page 10: News](#)

8.3.2 Methods

8.3.3 activateIgnoringOtherApps(flag as boolean)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Makes the receiver the active application.

Notes: flag: If false, the application is activated only if no other application is currently active. If true, the application activates regardless.

The flag parameter is normally set to false. When the Finder launches an application, using a value of false for flag allows the application to become active if the user waits for it to launch, but the application remains unobtrusive if the user activates another application. Regardless of the setting of flag, there may be a time lag before the application activates—you should not assume the application will be active immediately after sending this message.

You rarely need to invoke this method. Under most circumstances, the Application Kit takes care of proper activation. However, you might find this method useful if you implement your own methods for inter-application communication.

You don't need to send this message to make one of the application's NSWindows key. When you send a makeKeyWindow message to an NSWindow object, you ensure that it is the key window when the application is active.

8.3.4 addWindowsItem(win as NSWindowMBS, title as string, isFilename as boolean)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Adds an item to the Window menu for a given window.

Notes: win: The window being added to the menu. If this window object already exists in the Window menu, this method has no effect.

Title: The string to display for the window's menu item. How the string is interpreted is dependent on the value in the isFilename parameter.

isFilename: If false, title appears literally in the menu; otherwise, title is assumed to be a converted pathname

with the name of the file preceding the path (the way the `NSWindow` method `setTitleWithRepresentedFilename` shows a title)

You rarely need to invoke this method directly because Cocoa places an item in the Window menu automatically whenever you set the title of an `NSWindow` object.

8.3.5 `arrangeInFront`

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Arranges windows listed in the Window menu in front of all other windows.

Example:

```
dim a as new NSApplicationMBS
a.arrangeInFront
```

Notes: Windows associated with the application but not listed in the Window menu are not ordered to the front.

8.3.6 `cancelUserAttentionRequest(request as Integer)`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Cancels a previous user attention request.

Notes: request: The request identifier returned by the `requestUserAttention` method.

A request is also canceled automatically by user activation of the application.

8.3.7 `changeWindowsItem(win as NSWindowMBS, title as string, isFilename as boolean)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Changes the item for a given window in the Window menu to a given string.

Notes: win: The window whose title you want to change in the Window menu. If win is not in the Window menu, this method adds it.

title: The string to display for the window's menu item. How the string is interpreted is dependent on the value in the `isFilename` parameter.

isFilename: If false, title appears literally in the menu; otherwise, title is assumed to be a converted pathname with the name of the file preceding the path (the way the `NSWindow` method `setTitleWithRepresentedFile-`

name shows a title)

8.3.8 completeStateRestoration

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

Function: Completes the extended state restoration.

Notes: This method informs the application that the extended state restoration is completed for the balancing .

If a window has some state that may take a long time to restore, such as a web page, you may use this method and methods to completeStateRestoration to extend the period of this crash protection beyond the default.

You call extendStateRestoration within your implementation of restoreWindowWithIdentifier. You would then call completeStateRestoration some time after the window is fully restored. If the app crashes in the interim, then it may offer to discard restorable state on the next launch.

The extendStateRestoration and completeStateRestoration method act as a counter. Each call to extendStateRestoration increments the counter, and must be matched with a corresponding call to completeStateRestoration which decrements it. When the counter reaches zero, the app is considered to have been fully restored, and any further calls are silently ignored.

This method is thread safe.

Available in OS X v10.7 and later.

8.3.9 Constructor

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: The constructor.

Example:

```
dim n as new NSApplicationMBS

n.dockTile.badgeLabel = "Hello"
n.dockTile.showsApplicationBadge = true
```

Notes: Creates an object which points to the shared NSApplication instance.

8.3.10 deactivate

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Deactivates the receiver.

Example:

```
dim a as new NSApplicationMBS
```

```
a.deactivate
```

Notes: Normally, you shouldn't invoke this method—the Application Kit is responsible for proper deactivation.

8.3.11 disableRelaunchOnLogin

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Disable relaunching this app on login, if the app was running at the time the user logged out.

Notes: These methods increment and decrement a counter respectively; if the counter is 0 at the time the user logs out, then the app may be relaunched when the user logs back in. The counter is initially zero, so by default apps are relaunched.

If your app should not be relaunched because it launches via some other mechanism (e.g. `launchd`), then the recommended usage is to call `disableRelaunchOnLogin` once, and never pair it with an `enable` call.

If your app should not be relaunched because it triggers a restart (e.g. an installer), then the recommended usage is to call `disableRelaunchOnLogin` immediately before you attempt to trigger a restart, and `enableRelaunchOnLogin` immediately after. This is because the user may cancel restarting; if the user later restarts for another reason, then your app should be brought back.

These methods are thread safe.

Available on Mac OS X 10.7 or newer.

8.3.12 discardEventsMatchingMask(mask as Integer, beforeEvent as NSEventMBS)

Plugin Version: 17.5, Platform: macOS, Targets: Desktop only.

Function: Removes all events matching the given mask and generated before the specified event.

Notes: `mask`: Contains one or more flags indicating the types of events to discard. The constants section of the `NSEvent` class defines the constants you can add together to create this mask. The discussion section also lists some of the constants that are typically used.

`lastEvent`: A marker event that you use to indicate which events should be discarded. Events that occurred before this event are discarded but those that occurred after it are not.

Use this method to ignore any events that occurred before a specific event. For example, suppose your app has a tracking loop that you exit when the user releases the mouse button. You could use this method, specifying `NSAnyEventMask` as the mask argument and the ending mouse-up event as the `lastEvent` argument, to discard all events that occurred while you were tracking mouse movements in your loop. Passing the mouse-up event as `lastEvent` ensures that any events that might have occurred after the mouse-up event (that is, that appear in the queue after the mouse-up event) are not discarded.

Typically, you send this message to an `NSWindow` object, rather than to the app object. Discarding events for a window clears out all of the events for that window only, leaving events for other windows in place.

For the mask parameter, you can add together event type constants such as the following:

- `NSLeftMouseDownMask`
- `NSLeftMouseUpMask`
- `NSRightMouseDownMask`
- `NSRightMouseUpMask`
- `NSMouseMovedMask`
- `NSLeftMouseDraggedMask`
- `NSRightMouseDraggedMask`
- `NSMouseEnteredMask`
- `NSMouseExitedMask`
- `NSKeyDownMask`
- `NSKeyUpMask`
- `NSFlagsChangedMask`
- `NSPeriodicMask`
- `NSCursorUpdateMask`
- `NSAnyEventMask`

This method can also be called in subthreads. Events posted in subthreads bubble up in the main thread event queue.

8.3.13 `enabledRemoteNotificationTypes` as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns what badges you enabled.

Notes: Available on Mac OS X 10.7 or newer.

8.3.14 `enableRelaunchOnLogin`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Enable relaunching this app on login, if the app was running at the time the user logged out.

Notes: These methods increment and decrement a counter respectively; if the counter is 0 at the time the user logs out, then the app may be relaunched when the user logs back in. The counter is initially zero, so by default apps are relaunched.

If your app should not be relaunched because it launches via some other mechanism (e.g. `launchd`), then the recommended usage is to call `disableRelaunchOnLogin` once, and never pair it with an `enable` call.

If your app should not be relaunched because it triggers a restart (e.g. an installer), then the recommended usage is to call `disableRelaunchOnLogin` immediately before you attempt to trigger a restart, and `enableRelaunchOnLogin` immediately after. This is because the user may cancel restarting; if the user later restarts for another reason, then your app should be brought back.

These methods are thread safe.

Available on Mac OS X 10.7 or newer.

8.3.15 `extendStateRestoration`

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

Function: Allows an application to extend its state restoration period.

Notes: This method allows an application to extend the state restoration period beyond the usual. For example, the app crashes before state restoration is complete, then it may offer to discard restorable state on the next launch.

If a window has some state that may take a long time to restore, such as a web page, you may use this method and methods to `completeStateRestoration` to extend the period of this crash protection beyond the default.

You call `extendStateRestoration` within your implementation of `restoreWindowWithIdentifier`. You would then call `completeStateRestoration` some time after the window is fully restored. If the app crashes in the

interim, then it may offer to discard restorable state on the next launch.

The `extendStateRestoration` and `completeStateRestoration` method act as a counter. Each call to `extendStateRestoration` increments the counter, and must be matched with a corresponding call to `completeStateRestoration` which decrements it. When the counter reaches zero, the app is considered to have been fully restored, and any further calls are silently ignored.

This method is thread safe.
Available in OS X v10.7 and later.

8.3.16 `hide`

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Hides all the receiver's windows, and the next application in line is activated.

Example:

```
NSApplicationMBS.sharedApplication.hide
```

Notes: This method is usually invoked when the user chooses `Hide` in the application's main menu. When this method begins, it posts an `NSApplicationWillHideNotification` to the default notification center. When it completes successfully, it posts an `NSApplicationDidHideNotification`.

8.3.17 `hideOtherApplications`

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Hides all applications, except the receiver.

Example:

```
NSApplicationMBS.sharedApplication.hideOtherApplications
```

8.3.18 `invalidateRestorableState`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Invalidates the restorable state.

Notes: `applicationWillEncodeRestorableState` event will be called soon in your `NSApplicationDelegateMBS` subclass to get a new state encoded.

8.3.19 miniaturizeAll

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Miniaturizes all the receiver's windows.

Example:

```
NSApplicationMBS.sharedApplication.miniaturizeAll
```

8.3.20 modalWindow as NSWindowMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns the modal window that the receiver is displaying.

Example:

```
// show title of current dialog
MsgBox NSApplicationMBS.sharedApplication.modalWindow.Title
```

Notes: Returns the modal window being displayed or nil if no modal window is being displayed.

This method returns the current standalone modal window. It does not return sheets that are attached to other windows. If you need to retrieve a sheet window, use the attachedSheet method of NSWindow.

8.3.21 nextEventMatchingMask(mask as Integer, untilDate as date, mode as String, dequeueFlag as boolean) as NSEventMBS

Plugin Version: 17.5, Platform: macOS, Targets: Desktop only.

Function: Returns the next event matching a given mask, or nil if no such event is found before a specified expiration date.

Example:

```
Function MouseDown(X As Integer, Y As Integer) Handles MouseDown as Boolean
// in mousedown event
```

```
dim a as NSApplicationMBS = NSApplicationMBS.sharedApplication
```

```
// check current event
dim e as NSEventMBS = a.currentEvent
```

```
// check next mouse up event
dim d as new date
```

```
d.Second = d.Second + 2 // maximum 2 seconds wait time
dim n as NSEventMBS = a.nextEventMatchingMask(NSEventMBS.NSLeftMouseDownMask, d, NSRunLoopMBS.NSDefaultRunLoopMode, false)
```

```
// e->mouseDown
// e->mouseUp, if already available
```

```
Break
End Function
```

Notes: mask: Contains one or more flags indicating the types of events to return. The constants section of the NSEvent class defines the constants you can add together to create this mask. The discardEventsMatchingMask method also lists several of these constants.

expiration: The expiration date for the current event request. Specifying nil for this parameter is equivalent to returning a date object using the distantPast method.

mode: The run loop mode in which to run while looking for events. The mode you specify also determines which timers and run-loop observers may fire while the app waits for the event.

flag: Specify true if you want the event removed from the queue.

Returns the event object whose type matches one of the event types specified by the mask parameter.

You can use this method to short circuit normal event dispatching and get your own events. For example, you may want to do this in response to a mouse-down event in order to track the mouse while its button is down. (In such an example, you would pass the appropriate event types for mouse-dragged and mouse-up events to the mask parameter and specify the NSEventTrackingRunLoopMode run loop mode.) Events that do not match one of the specified event types are left in the queue.

You can specify one of the run loop modes defined by AppKit or a custom run loop mode used specifically by your app. AppKit defines the following run-loop modes:

- NSDefaultRunLoopMode
- NSEventTrackingRunLoopMode
- NSModalPanelRunLoopMode
- NSConnectionReplyMode

See also:

- 8.3.22 nextEventMatchingMask(mask as Integer, untilDate as dateTime, mode as String, dequeueFlag as boolean) as NSEventMBS

8.3.22 nextEventMatchingMask(mask as Integer, untilDate as dateTime, mode as String, dequeueFlag as boolean) as NSEventMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Returns the next event matching a given mask, or nil if no such event is found before a specified expiration date.

See also:

- 8.3.21 nextEventMatchingMask(mask as Integer, untilDate as date, mode as String, dequeueFlag as boolean) as NSEventMBS 302

8.3.23 NSAppKitVersionNumber as Double

Plugin Version: 11.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: This constant identifies the installed version of the Application Kit framework.

Example:

```
const NSAppKitVersionNumber10_0 = 577
const NSAppKitVersionNumber10_1 = 620
const NSAppKitVersionNumber10_2 = 663
const NSAppKitVersionNumber10_2_3 = 663.6
const NSAppKitVersionNumber10_3 = 743
const NSAppKitVersionNumber10_3_2 = 743.14
const NSAppKitVersionNumber10_3_3 = 743.2
const NSAppKitVersionNumber10_3_5 = 743.24
const NSAppKitVersionNumber10_3_7 = 743.33
const NSAppKitVersionNumber10_3_9 = 743.36
const NSAppKitVersionNumber10_4 = 824
const NSAppKitVersionNumber10_4_1 = 824.1
const NSAppKitVersionNumber10_4_3 = 824.23
const NSAppKitVersionNumber10_4_4 = 824.33
const NSAppKitVersionNumber10_4_7 = 824.41
const NSAppKitVersionNumber10_5 = 949
const NSAppKitVersionNumber10_5_2 = 949.27
const NSAppKitVersionNumber10_5_3 = 949.33
const NSAppKitVersionNumber10_6 = 1038
const NSAppKitVersionNumber10_7 = 1138
const NSAppKitVersionNumber10_7_2 = 1138.23
```

```
if NSApplicationMBS.NSAppKitVersionNumber >= NSAppKitVersionNumber10_5 then
  MsgBox "This is Mac OS X 10.5 or newer."
end if
```

Notes: See NSAppKitVersionNumber* constants.

8.3.24 NSApplicationDidBecomeActiveNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted immediately after the application becomes active.

The notification object is sharedApplication. This notification does not contain a userInfo dictionary.

8.3.25 NSApplicationDidChangeScreenParametersNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted when the configuration of the displays attached to the computer is changed.

The configuration change can be made either programmatically or when the user changes settings in the Displays control panel. The notification object is sharedApplication. This notification does not contain a userInfo dictionary.

8.3.26 NSApplicationDidFinishLaunchingNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the end of the finishLaunching method to indicate that the application has completed launching and is ready to run.

The notification object is sharedApplication. This notification does not contain a userInfo dictionary.

8.3.27 NSApplicationDidFinishRestoringWindowsNotification as string

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

Function: Posted when the application is finished restoring windows.

Notes: The notification is posted when the application is finished restoring windows, that is, when all the completion handlers from restoreWindowWithIdentifier have been called. This is always posted after NSApplicationWillFinishLaunchingNotification, but may be posted before or after NSApplicationDidFinishLaunchingNotification, depending on whether clients copy the completion handlers and invoke them later. If there were no windows to restore, then this notification is still posted at the corresponding point in app launch (between NSApplicationWillFinishLaunchingNotification and NSApplicationDidFinishLaunchingNotification).

The notification object is `sharedApplication`. This notification does not contain a `userInfo` dictionary. Available in OS X v10.7 and later.

8.3.28 `NSApplicationDidHideNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the end of the `hide` method to indicate that the application is now hidden. The notification object is `NSApp`. This notification does not contain a `userInfo` dictionary.

8.3.29 `NSApplicationDidResignActiveNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted immediately after the application gives up its active status to another application. The notification object is `NSApp`. This notification does not contain a `userInfo` dictionary.

8.3.30 `NSApplicationDidUnhideNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the end of the `unhideWithoutActivation` method to indicate that the application is now visible. The notification object is `NSApp`. This notification does not contain a `userInfo` dictionary.

8.3.31 `NSApplicationDidUpdateNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the end of the `updateWindows` method to indicate that the application has finished updating its windows. The notification object is `NSApp`. This notification does not contain a `userInfo` dictionary.

8.3.32 NSApplicationLaunchIsDefaultLaunchKey as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the keys for the userInfo dictionary on didFinishLaunching.

Notes: The following key is present in the userInfo of NSApplicationDidFinishLaunchingNotification or in the didFinishLaunching event. Its value is a number containing a bool. It will be false if the app was launched to open or print a file, to perform a Service, if the app had saved state that will be restored, or if the app launch was in some other sense not a "default" launch. Otherwise its value will be true.

Available on Mac OS X 10.7 or newer.

8.3.33 NSApplicationLaunchRemoteNotificationKey as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the keys for the userInfo dictionary on didFinishLaunching.

Notes: User info keys for NSApplicationDidFinishLaunchingNotification.

Available on Mac OS X 10.7 or newer.

8.3.34 NSApplicationLaunchUserNotificationKey as string

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

Function: One of the user info keys for NSApplicationDidFinishLaunchingNotification.

Example:

```
Sub applicationDidFinishLaunching(Notification as NSNotificationMBS)
dim userInfo as Dictionary = Notification.userInfo
dim key as string = NSApplicationMBS.NSApplicationLaunchUserNotificationKey
dim UserNotification as NSUserNotificationMBS = userInfo.Lookup(key, nil)

if UserNotification <>nil then
MsgBox UserNotification.identifier+": "+UserNotification.informativeText
end if
End Sub
```

Notes: This key is present in the userInfo of NSApplicationDidFinishLaunchingNotification. It will be present if your application was launched because a user activated a notification in the Notification Center. Its value is an NSUserNotification object.

Available in Mac OS X 10.8.

8.3.35 `NSApplicationWillBecomeActiveNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted immediately after the application becomes active.

The notification object is `NSApp`. This notification does not contain a `userInfo` dictionary.

8.3.36 `NSApplicationWillFinishLaunchingNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the start of the `finishLaunching` method to indicate that the application has completed its initialization process and is about to finish launching.

The notification object is `NSApp`. This notification does not contain a `userInfo` dictionary.

8.3.37 `NSApplicationWillHideNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the start of the `hide` method to indicate that the application is about to be hidden.

The notification object is `NSApp`. This notification does not contain a `userInfo` dictionary.

8.3.38 `NSApplicationWillResignActiveNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted immediately before the application gives up its active status to another application.

The notification object is `sharedApplication`. This notification does not contain a `userInfo` dictionary.

8.3.39 `NSApplicationWillTerminateNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted by the `terminate` method to indicate that the application will terminate.

Posted only if the delegate method `applicationShouldTerminate` returns true. The notification object is `sharedApplication`. This notification does not contain a `userInfo` dictionary.

8.3.40 NSApplicationWillUnhideNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the start of the `unhideWithoutActivation` method to indicate that the application is about to become visible.

The notification object is `sharedApplication`. This notification does not contain a `userInfo` dictionary.

8.3.41 NSApplicationWillUpdateNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: One of the notification names.

Notes: Posted at the start of the `updateWindows` method to indicate that the application is about to update its windows.

The notification object is `sharedApplication`. This notification does not contain a `userInfo` dictionary.

8.3.42 orderFrontCharacterPalette

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Opens the character palette.

Example:

```
dim a as new NSApplicationMBS
a.orderFrontCharacterPalette
```

Notes: Available in Mac OS X v10.3 and later.

This shows the special characters palette.

8.3.43 orderFrontStandardAboutPanel

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Displays a standard About window.

Example:

```
dim a as new NSApplicationMBS
a.orderFrontStandardAboutPanel
```

Notes: This method calls `orderFrontStandardAboutPanelWithOptions` with a `nil` argument. See `orderFrontStandardAboutPanelWithOptions` for a description of what's displayed.

8.3.44 `orderFrontStandardAboutPanelWithOptions(options as dictionary)`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Displays a standard About window with information from a given options dictionary.

Example:

```
dim a as new NSApplicationMBS

// this image has no mask, so you'll see a white border
dim p as Picture = LogoMBS(500)
dim n as new NSImageMBS(p)
dim options as new Dictionary

// show window with default values
'a.orderFrontStandardAboutPanel

// we overwrite default values with new values
options.Value("Credits") = NSAttributedStringMBS.attributedStringWithString("Written by Christian Schmitz")
options.Value("ApplicationName") = "MyCoolApp"
options.Value("Version") = "1.2.3"
options.Value("Copyright") = "©2011 Monkeybread Software"
options.Value("ApplicationIcon") = n

a.orderFrontStandardAboutPanelWithOptions(options)

Exception ex as NSErrorMBS
MsgBox ex.message
```

Notes: `options`: A dictionary whose keys define the contents of the About window. See the discussion for a description of the available keys.

The following strings are keys that can occur in `optionsDictionary`:

8.3.45 `OverlayApplicationIconImage(image as NSImageMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

” ”	An NSAttributedStringMBS displayed in the info area of the panel. If not specified, this method then looks for a file named ”Credits.html”, ”Credits.rtf”, and ”Credits.rtf”, in that order, in the bundle returned by the NSBundle class method mainBundle. The first file found is used. If none is found, the info area is left blank.
”ApplicationName”	A string displayed as the application’s name. If not specified, this method then uses the value of CFBundleName (localizable). If neither is found, this method uses the process name.
”ApplicationIcon”	A NSImageMBS object displayed as the application’s icon. If not specified, this method then looks for an image named ”NSApplicationIcon”, using NSImageMBS.imageNamed(”NSApplicationIcon”). If neither is available, this method uses the generic application icon.
”Version”	A string with the build version number of the application (”58.4”), displayed as ”(v58.4)”. If not specified, obtain from the CFBundleVersion key in infoDictionary; if not specified, leave blank (the ”(v)” is not displayed).
”Copyright”	A string with a line of copyright information. If not specified, this method then looks for the value of NSHumanReadableCopyright in the localized version infoDictionary. If neither is available, this method leaves the space blank.
”ApplicationVersion”	A string with the application version (”Mac OS X”, ”3”, ”WebObjects 4.5”, ”AppleWorks 6”,...). If not specified, obtain from the CFBundleShortVersionString key in infoDictionary. If neither is available, the build version, if available, is printed alone, as ”Version x.x”.

Function: Overlays the image.

Example:

```
// draw a picture with a red dot on the top left
```

```
dim p as new Picture(512,512,32)
```

```
dim g as Graphics = p.Graphics
```

```
g.ForeColor = &cFF0000
```

```
g.FillRect 0,0,128,128
```

```
g = p.mask.Graphics
```

```
g.ForeColor = &cFFFFFF
```

```
g.Fillrect 0,0,512,512
```

```
g.ForeColor = &c000000
```

```
g.Filloval 0,0,128,128
```

```
Backdrop = p
```

```
// create nsimage
```

```
dim n as new NSImageMBS(p,p.mask)
```

```
// and overlay over original image
NSApplicationMBS.sharedApplication.OverlayApplicationIconImage n
```

Notes: Same as setting `applicationIconImage`, but instead overlays the original image with the new image. This way you can show custom badges. For normal text badges, use `NSDockTileMBS` class.

Pass `image = nil` to reset application dock icon.

Size of the image seems to be 128 Pixel by default, but could go up to 1024 in the future. Plugin scales up or down as needed for you.

8.3.46 `postEvent(e as NSEventMBS, atStart as boolean)`

Plugin Version: 17.5, Platform: macOS, Targets: Desktop only.

Function: Adds a given event to the receiver,Äôs event queue.

Notes: `anEvent`: The event object to post to the queue.

`flag`: Specify true to add the event to the front of the queue; otherwise, specify false to add the event to the back of the queue.

This method can also be called in subthreads. Events posted in subthreads bubble up in the main thread event queue.

8.3.47 `preventWindowOrdering`

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

Function: Suppresses the usual window ordering in handling the most recent mouse-down event.

Notes: This method is only useful for mouse-down events when you want to prevent the window that receives the event from being ordered to the front.

8.3.48 `registerForRemoteNotificationTypes(type as Integer)`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Registers for remote notifications.

Example:

```
NSApplicationMBS.sharedApplication.registerForRemoteNotificationTypes NSApplicationMBS.NSRemoteNo-
tificationTypeBadge
```

Notes: Only applications distributed through the Mac App Store may use Push notifications.

Type can only be NSRemoteNotificationTypeBadge currently for Mac OS X 10.7.0.

Available on Mac OS X 10.7 or newer.

8.3.49 removeWindowsItem(win as NSWindowMBS)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Removes the Window menu item for a given window.

Notes: Win: The window whose menu item is to be removed.

This method doesn't prevent the item from being automatically added again. Use the `excludedFromWindowsMenu` method of `NSWindow` if you want the item to remain excluded from the Window menu.

8.3.50 replyToApplicationShouldTerminate(reply as boolean)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Responds to `NSTerminateLater` once the application knows whether it can terminate.

Notes: `souldTerminate`: Specify true if you want the application to terminate; otherwise, specify false.

If your application delegate returns `NSTerminateLater` from its `applicationShouldTerminate` event, your code must subsequently call this method to let the `NSApplication` object know whether it can actually terminate itself.

8.3.51 replyToOpenOrPrint(reply as Integer)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Handles errors that might occur when the user attempts to open or print files.

Notes: `reply`: The error that occurred. For a list of possible values, see "Constants."

Delegates should invoke this method if an error is encountered in the `applicationOpenFiles` or `applicationPrintFiles` delegate methods.

8.3.52 requestUserAttention(type as Integer) as Integer

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Starts a user attention request.

Example:

```
dim a as NSApplicationMBS = NSApplicationMBS.sharedApplication

if not a.isActive then // only when we are in background
call a.requestUserAttention(a.NSInformationalRequest) // dock bounces
end if
```

Notes: requestType: The severity of the request. Can be NSInformationalRequest or NSCriticalRequest.

Returns the identifier for the request. You can use this value to cancel the request later using the cancelUserAttentionRequest method.

Activating the application cancels the user attention request. A spoken notification will occur if spoken notifications are enabled. Sending requestUserAttention to an application that is already active has no effect.

If the inactive application presents a modal panel, this method will be invoked with NSCriticalRequest automatically. The modal panel is not brought to the front for an inactive application.

8.3.53 runModalForWindow(win as NSWindowMBS) as Integer

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

Function: Starts a modal event loop for the specified window.

Notes: win: The window to be displayed modally. If it is not already visible, the window is centered on the screen using the value in its center method and made visible and key. If it is already visible, it is simply made key.

Returns an integer indicating the reason that this method returned.

This method runs a modal event loop for the specified window synchronously. It displays the specified window, makes it key, starts the run loop, and processes events for that window. (You do not need to show the window yourself.) While the app is in that loop, it does not respond to any other events (including mouse, keyboard, or window-close events) unless they are associated with the window. It also does not perform any tasks (such as firing timers) that are not associated with the modal run loop. In other words, this method consumes only enough CPU time to process events and dispatch them to the action methods associated with the modal window.

You can exit the modal loop by calling the `stopModal`, `stopModalWithCode`, or `abortModal` methods from your modal window code. If you use the `stopModalWithCode` method to stop the modal event loop, this method returns the argument passed to `stopModalWithCode`. If you use `stopModal` instead, this method returns the constant `NSModalResponseStop` (-1000). If you use `abortModal`, this method returns the constant `NSModalResponseAbort` (-1001).

8.3.54 runPageLayout

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Displays the app's page layout panel, an instance of `NSPageLayout`.

Example:

```
NSApplicationMBS.sharedApplication.runPageLayout
```

Notes: If the `NSPageLayout` instance does not exist, this method creates one. This method is typically invoked when the user chooses Page Setup from the application's File menu.

8.3.55 sendEvent(theEvent as NSEventMBS)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Dispatches an event to other objects.

Notes: You rarely invoke `sendEvent` directly, although you might want to override this method to perform some action on every event. `sendEvent` messages are sent from the main event loop (the `run` method). `sendEvent` is the method that dispatches events to the appropriate responders—`NSApp` handles application events, the `NSWindow` object indicated in the event record handles window-related events, and mouse and key events are forwarded to the appropriate `NSWindow` object for further dispatching.

If you need to override `sendEvent` method, please call MBS support.

8.3.56 sharedApplication as NSApplicationMBS

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Returns the application instance, creating it if it doesn't exist yet.

Notes: The plugin makes sure that there is only one application object by returning the same object each time.

8.3.57 showHelp

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

Function: Shows help.

Example:

```
NSApplicationMBS.sharedApplication.showHelp
```

Notes: If your project is properly registered, and the necessary keys have been set in the property list, this method launches Help Viewer and displays the first page of your app,Äôs help book.

8.3.58 startDictation

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

Function: Starts dictation.

Example:

```
NSApplicationMBS.sharedApplication.startDictation
```

Notes: For Mac OS X 10.8.

8.3.59 stopDictation

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

Function: Stops dictation.

Notes: For Mac OS X 10.8.

8.3.60 terminate

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Terminates the receiver.

Example:

```
NSApplicationMBS.sharedApplication.terminate
```

Notes: This method is typically invoked when the user chooses Quit or Exit from the application’s menu.

When invoked, this method performs several steps to process the termination request. First, it asks the application’s document controller (if one exists) to save any unsaved changes in its documents. During this process, the document controller can cancel termination in response to input from the user. If the document controller does not cancel the operation, this method then calls the delegate’s `applicationShouldTerminate` method. If `applicationShouldTerminate` returns `NSTerminateCancel`, the termination process is aborted and control is handed back to the main event loop. If the method returns `NSTerminateLater`, the application runs its run loop in the `NSModalPanelRunLoopMode` mode until the `replyToApplicationShouldTerminate` method is called with the value `true` or `false`. If the `applicationShouldTerminate` method returns `NSTerminateNow`, this method posts a `NSApplicationWillTerminateNotification` notification to the default notification center.

Do not bother to put final cleanup code in your application’s `main()` function—it will never be executed. If cleanup is necessary, perform that cleanup in the delegate’s `applicationWillTerminate` method.

8.3.61 `unhide`

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Restores hidden windows to the screen and makes the receiver active.

Example:

```
NSApplicationMBS.sharedApplication.unhide
```

8.3.62 `unhideAllApplications`

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Unhides all applications, including the receiver.

Example:

```
NSApplicationMBS.sharedApplication.unhideAllApplications
```

Notes: This action causes each application to order its windows to the front, which could obscure the currently active window in the active application.

8.3.63 `unhideWithoutActivation`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Restores hidden windows without activating their owner (the receiver).

Example:

```
NSApplicationMBS.sharedApplication.unhideWithoutActivation
```

Notes: When this method begins, it posts an `NSApplicationWillUnhideNotification` to the default notification center. If it completes successfully, it posts an `NSApplicationDidUnhideNotification`.

8.3.64 unregisterForRemoteNotifications

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Unregisters for remote notifications.

Example:

```
NSApplicationMBS.sharedApplication.unregisterForRemoteNotifications
```

Notes: Available on Mac OS X 10.7 or newer.

8.3.65 updateWindows

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

Function: Sends an update message to each onscreen window.

Notes: This method is invoked automatically in the main event loop after each event when running in `NSDefaultRunLoopMode` or `NSModalRunLoopMode`. This method is not invoked automatically when running in `NSEventTrackingRunLoopMode`.

When this method begins, it posts an `NSApplicationWillUpdateNotification` to the default notification center. When it successfully completes, it posts an `NSApplicationDidUpdateNotification`.

8.3.66 updateWindowsItem(win as NSWindowMBS)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Updates the Window menu item for a given window to reflect the edited status of that window.

Notes: win: The window whose menu item is to be updated.

You rarely need to invoke this method because it is invoked automatically when the edit status of an `NSWin-`

dow object is set.

8.3.67 windows as NSWindowMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns an array containing the receiver's window objects.

Example:

```
// show all window titles in message boxes
for each w as NSWindowMBS in NSApplicationMBS.sharedApplication.windows
  MsgBox w.Title
next
```

Notes: Returns an array of NSWindow objects. This array includes both onscreen and offscreen windows.

8.3.68 windowWithWindowNumber(windowNumber as Integer) as NSWindowMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns the window corresponding to the specified window number.

Example:

```
dim w as window = window1
w.Title = "This is our test window"

// get a window ID somewhere
dim WindowID as Integer = CGWindowMBS.GetWindowID(w)

// now find back the window
dim n as NSWindowMBS = NSApplicationMBS.sharedApplication.windowWithWindowNumber(windowid)

MsgBox n.Title
```

Notes: windowNumber: The unique window number associated with the desired NSWindow object.

Returns the desired window object or nil if the window could not be found.

8.3.69 Properties

8.3.70 activationPolicy as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: The application's activation policy.

Notes: Currently, `NSApplicationActivationPolicyNone` and `NSApplicationActivationPolicyAccessory` may be changed to `NSApplicationActivationPolicyRegular`, but other modifications are not supported.

Update: Seems to work better newer MacOS versions, so between 10.6 and 10.15 Apple implemented more transitions.

Available in Mac OS X v10.6 and later.
(Read and Write property)

8.3.71 applicationIconImage as NSImageMBS

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: The image used for the receiver's icon.

Example:

```
dim pic as Picture = LogoMBS(500)
dim mask as new Picture(500, 500, 32)
dim g as Graphics = mask.Graphics
```

```
g.ForeColor = &cFFFFFF
g.FillRect 0,0,g.Width, g.Height
g.ForeColor = &c000000
g.FillOval 0,0,g.Width, g.Height
```

```
dim n as new NSImageMBS(pic,mask)
```

```
NSApplicationMBS.sharedApplication.applicationIconImage = n
```

Notes: An image containing the application's icon.

This property can set the icon in the dock application tile. This method scales the image as necessary so that it fits in the dock tile. You can use this method to change your application icon while running. To restore your application's original icon, you pass nil to this method.
(Read and Write property)

8.3.72 currentEvent as NSEventMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the current event, the last event the receiver retrieved from the event queue.

Example:

```
dim e as NSEventMBS = NSApplicationMBS.sharedApplication.currentEvent
```

Notes: The last event object retrieved by the application.

NSApp receives events and forwards them to the affected NSWindow objects, which then distribute them to the objects in its view hierarchy.

Only for Cocoa applications.
(Read only property)

8.3.73 currentSystemPresentationOptions as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns the set of application presentation options that are currently in effect for the system.

Example:

```
dim a as new NSApplicationMBS
```

```
MsgBox "currentSystemPresentationOptions: "+str(a.currentSystemPresentationOptions)
```

Notes: The presentation options. See NSApplicationPresentation* constants and combine them using a C bitwise OR operator.

These are the presentation options that have been put into effect by the currently active application.
(Read only property)

8.3.74 dockTile as NSDockTileMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the application's Dock tile.

Example:

```
dim a as new NSApplicationMBS
```

```
a.dockTile.badgeLabel = "Hello"
```

Notes: Available in Mac OS X v10.5 and later.

Returns NSDockTileMBS object for the dock tile.
(Read only property)

8.3.75 Handle as Integer

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: The internal reference to the NSApplication object.

Notes: (Read and Write property)

8.3.76 helpMenu as NSMenuMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The help menu used by the application.

Notes: If helpMenu is nil, the system will append the help to an appropriate menu and will not return a reference to that menu when this method is called.

Available in Mac OS X v10.6 and later.

If helpMenu is a non-nil menu it is set as the Help menu, and Spotlight for Help will be installed in it. If helpMenu is nil, AppKit will install Spotlight for Help into a menu of its choosing, and that menu is not returned from helpMenu.

If you wish to completely suppress Spotlight for Help, you can set a menu that does not appear in the menu bar.

NSApplication retains its Help menu and releases it when a different menu is set.
(Read and Write property)

8.3.77 isActive as Boolean

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether this is the active application.

Example:

```
dim a as NSApplicationMBS = NSApplicationMBS.sharedApplication
MsgBox "isActive:" +str(a.isActive)
```

Notes: True if this is the active application; false otherwise.
(Read only property)

8.3.78 isFullKeyboardAccessEnabled as Boolean

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Returns that status of Full Keyboard Access set in the Keyboard preference pane.

Example:

```
dim a as NSApplicationMBS = NSApplicationMBS.sharedApplication
MsgBox "isFullKeyboardAccessEnabled:" +str(a.isFullKeyboardAccessEnabled)
```

Notes: True if Full Keyboard Access is enabled, otherwise false.

You may use this status to implement your own key loop or to implement in-control tabbing behavior similar to NSTableView.

Available in Mac OS X v10.6 and later.
(Read only property)

8.3.79 isHidden as Boolean

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the receiver is hidden.

Example:

```
MsgBox str(NSApplicationMBS.sharedApplication.isHidden)
```

Notes: True if the receiver is hidden, false otherwise.
(Read only property)

8.3.80 isRunning as Boolean

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the main event loop is running.

Example:

```
MsgBox str(NSApplicationMBS.sharedApplication.isRunning)
```

Notes: True if the main event loop is running; false otherwise.
False means the stop: method was invoked.

Should always be true for a Xojo application.
(Read only property)

8.3.81 keyWindow as NSWindowMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the window that currently receives keyboard events.

Example:

```
// title of front window  
MsgBox NSApplicationMBS.sharedApplication.keyWindow.Title
```

Notes: The window object currently receiving keyboard events or nil if there is no key window.
This method might return nil if the application hasn't finished loading yet or if the receiver is not active.

Does return nil in Carbon applications.
(Read only property)

8.3.82 mainMenu as NSMenuMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The menu object representing the application's menu bar.

Example:

```
// shows titles of the menus in a Cocoa app
dim m as NSMenuMBS = NSApplicationMBS.sharedApplication.mainMenu

if m<>nil then
dim c as Integer = m.numberOfItems-1
for i as Integer = 0 to c

MsgBox m.Item(i).Title
next
end if
```

Notes: Returns nil on a Carbon application.
(Read and Write property)

8.3.83 mainWindow as NSWindowMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the main window.

Example:

```
// shows title in Cocoa, but not in Carbon
MsgBox NSApplicationMBS.sharedApplication.mainWindow.Title
```

Notes: The application's main window or nil if there is no main window.
This method might return nil if the application hasn't finished loading, if the receiver is not active, or if the application is hidden.
(Read only property)

8.3.84 presentationOptions as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: The presentation options that should be in effect for the system when this application is active.

Example:

```
dim a as new NSApplicationMBS

// hide dock
a.presentationOptions = NSApplicationMBS.NSApplicationPresentationAutoHideDock
```

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

Only certain combinations of "NSApplicationPresentationOptions" flags are supported. When given an invalid combination of option flags this method raises an exception `NSInvalidArgumentException` exception..

See `NSApplicationPresentation*` constants.
(Read and Write property)

8.3.85 `servicesProvider` as `NSServiceProviderMBS`

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Registers or queries the service provider.

Notes: The service provider is an object that performs all services the application provides to other applications. When another application requests a service from the receiver, it sends the service request to `aProvider`. Service requests can arrive immediately after the service provider is set, so invoke this method only when your application is ready to receive requests.

Please keep an object reference around so the object is not going out of scope too early!
(Read and Write property)

8.3.86 `userInterfaceLayoutDirection` as `Integer`

Plugin Version: 16.3, Platform: macOS, Targets: All.

Function: The layout direction of the user interface.

Notes: This property contains the general user interface layout flow directions. For a list of possible values, see `NSUserInterfaceLayoutDirection`.

`NSUserInterfaceLayoutDirectionLeftToRight` = 0
`NSUserInterfaceLayoutDirectionRightToLeft` = 1

Available in OS X v10.6 and later.
(Read only property)

8.3.87 windowsMenu as NSMenuMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The window menu or nil if such a menu does not exist or has not yet been created.

Notes: (Read and Write property)

8.3.88 Constants

Constants

Constant	Value	Description
NSRemoteNotificationTypeAlert	4	One of the notification type constants. Available in Mac OS X 10.8.
NSRemoteNotificationTypeBadge	1	One of the notification type constants.
NSRemoteNotificationTypeNone	0	One of the notification type constants.
NSRemoteNotificationTypeSound	2	One of the notification type constants. Available in Mac OS X 10.8.

Activation Policy Constants

Constant	Value	Description
NSApplicationActivationPolicyAccessory	1	The application does not appear in the Dock and does not have a menu bar, but it may be activated programmatically or by clicking on one of its windows. This corresponds to value of the LSUIElement key in the application's Info.plist being 1. Available in Mac OS X v10.6 and later.
NSApplicationActivationPolicyProhibited	2	The application does not appear in the Dock and may not create windows or be activated. This corresponds to the value of the LSBackgroundOnly key in the application's Info.plist being 1. This is also the default for unexecutables that do not have Info.plists. Available in Mac OS X v10.6 and later.
NSApplicationActivationPolicyRegular	0	The application is an ordinary app that appears in the Dock and has a menu bar and user interface. This is the default for bundled apps, unless overridden in the application's Info.plist. Available in Mac OS X v10.6 and later.

Presentation Option Constants.

Constant	Value	Description
<code>NSApplicationPresentationAutoHideDock</code>	1	
<code>NSApplicationPresentationAutoHideMenuBar</code>	4	
<code>NSApplicationPresentationAutoHideToolbar</code>	2048	Fullscreen window toolbar is detached from window and hidden. May be used only when both <code>NSApplicationPresentationFullScreen</code> and <code>NSApplicationPresentationAutoHideMenuBar</code> are set. Available on Mac OS X 10.7 or newer.
<code>NSApplicationPresentationDefault</code>	0	
<code>NSApplicationPresentationDisableAppleMenu</code>	16	
<code>NSApplicationPresentationDisableForceQuit</code>	64	
<code>NSApplicationPresentationDisableHideApplication</code>	256	
<code>NSApplicationPresentationDisableMenuBarTransparency</code>	512	
<code>NSApplicationPresentationDisableProcessSwitching</code>	32	
<code>NSApplicationPresentationDisableSessionTermination</code>	128	
<code>NSApplicationPresentationFullScreen</code>	1024	Application is in fullscreen mode. Available on Mac OS X 10.7 or newer.
<code>NSApplicationPresentationHideDock</code>	2	
<code>NSApplicationPresentationHideMenuBar</code>	8	

Request type constants.

Constant	Value	Description
<code>NSCriticalRequest</code>	0	The user attention request is a critical request. The dock icon will bounce until either the application becomes active or the request is canceled.
<code>NSInformationalRequest</code>	10	The user attention request is an informational request. The dock icon will bounce for one second. The request, though, remains active until either the application becomes active or the request is canceled.

8.4 class NSDockTileMBS

8.4.1 class NSDockTileMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The cocoa class to customize the dock tile.

Notes: The NSDockTile class lets you customize the visual representation for your application's miniaturized windows and application icon as they appear in the Dock. You do not create Dock tile objects explicitly in your application. Instead, you retrieve the Dock tile for an existing window or for the application by calling that object's dockTile method.

Typically, you do not subclass the NSDockTile class. Instead, you use the methods of the class to make the following customizations:

- Badge the tile with a custom string.
- Remove or show the application icon badge.
- Draw the tile content yourself.

If you decide to draw the tile content yourself, you must provide a custom content view to handle the drawing.

Application Dock Tiles

An application Dock tile defaults to display the application's applicationIconImage.

The application Dock tile never shows a smaller application icon badge.

Whether using the default or custom view, the application Dock tile may be badged with a short custom string.

Window Dock Tiles

A window Dock tile defaults to display a miniaturized version of the windows contents with a badge derived from the application Dock icon, including any customized application Dock icon. The default window Dock tile image may not be badged with a custom string.

A window Dock tile can use a custom view to draw the Dock icon. If a custom view is used, no application badge will be added, but the text label will be overlaid on top of the icon.

Available in Mac OS X v10.5 and later.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Real Studio Plugins, version 11.2pr9](#)

8.4.2 Methods

8.4.3 Constructor

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

8.4.4 display

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Redraws the dock tile's content.

Notes: If a custom content view is provided, Cocoa calls the `drawRect` method of that view (and its sub-views) to draw the tile's content.

You can call this method to force the redrawing of the dock tile contents. You might do this if the contents of the underlying application or window change in a way that would require a refreshing of the tile. Some types of system activity, such as resizing the dock, may trigger automatic redraws of the tile. In most cases, however, your application is responsible for triggering redraws.

Cocoa does not automatically redraw the contents of your dock tile. Instead, your application must explicitly send `display` messages to the dock tile object whenever the contents of your view change and need to be redrawn.

8.4.5 owner as Variant

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the object represented by the dock tile.

Example:

```
dim d as NSDockTileMBS = NSApplicationMBS.sharedApplication.dockTile

if d<>nil then
dim t as Introspection.TypeInfo = Introspection.GetType(d.owner)
MsgBox t.FullName // shows NSApplicationMBS
```

end if

Notes: The object represented by the dock tile. This is either the NSApplicationMBS object or one of your application's NSWindowMBS objects.

8.4.6 size as NSSizeMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the size of the tile.

Example:

```
dim d as NSDockTileMBS = NSApplicationMBS.sharedApplication.dockTile

if d<>nil then
  MsgBox str(D.size.Width)+" x "+str(d.size.Height)
  // 128 x 128 in Mac OS X 10.5 and 10.6
end if
```

Notes: The size returned by this method corresponds to the size of the backing store in the dock, which may be bigger than the actual tile displayed on the screen.

8.4.7 Properties

8.4.8 Handle as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The internal reference to the docktile object.

Notes: (Read and Write property)

8.4.9 badgeLabel as string

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The tile's current badge label.

Example:

```
dim d as NSDockTileMBS = NSApplicationMBS.sharedApplication.dockTile
```

```

if d<>nil then
// this works in Carbon and Cocoa applications :-)

d.badgeLabel = "Hello"
d.showsApplicationBadge = true
end if

```

Notes: The localized string to be displayed in the tile's badging area. This string may be empty. (Read and Write computed property)

8.4.10 contentView as NSViewMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The view used to draw the dock tile contents.

Notes: The view you specify should be height and width resizable.

Cocoa does not automatically redraw the contents of your dock tile. Instead, your application must explicitly send display messages to the dock tile object whenever the contents of your view change and need to be redrawn. Your dock tile view is responsible for drawing the entire contents of the dock tile. Your view does not need to draw the application or custom string badges. (Read and Write computed property)

8.4.11 showsApplicationBadge as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Whether the tile should be badged with the application's icon.

Example:

```

dim d as NSDockTileMBS = NSApplicationMBS.sharedApplication.dockTile

if d<>nil then
// this works in Carbon and Cocoa applications :-)

d.badgeLabel = "Hello"
d.showsApplicationBadge = true
end if

```

Notes: Miniaturized windows include the application badge by default to convey the associated application to the user. In Mac OS X v10.5 and later, application tiles do not support the application badge. A minia-

turized window with a custom view does not draw the application badge.

The application icon is positioned automatically in the tile by the NSDockTile object.
(Read and Write computed property)

8.5 class NSFontManagerMBS

8.5.1 class NSFontManagerMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: NSFontManager is the center of activity for the font conversion system.

Example:

```
// pick some font
dim n1 as NSFontMBS = nsfontmbs.fontWithName("Book Antiqua bold", 12)
dim fm as new NSFontManagerMBS

// remove all font traits
dim n2 as NSFontMBS = fm.convertFontToNotHaveTrait(n1, fm.traitsOfFont(n1))

// show name of base font
MsgBox n2.fontName
```

Notes: It records the currently selected font, updates the Font panel and Font menu to reflect the selected font, initiates font changes, and converts fonts in response to requests from text-bearing objects. In a more prosaic role, NSFontManager can be queried for the fonts available to the application and for the particular attributes of a font, such as whether it's condensed or extended.

As of Mac OS X version 10.3, font collections are managed by NSFontManager.

Blog Entries

- [MBS Xojo Plugins, version 17.3pr5](#)
- [MBS Releases the MBS Xojo / Real Studio plug-ins in version 16.4](#)
- [MBS Xojo / Real Studio Plugins, version 16.4pr9](#)
- [MBS Xojo / Real Studio Plugins, version 15.4pr5](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr1](#)
- [MBS Real Studio Plugins, version 12.0pr6](#)
- [MBS Real Studio Plugins, version 11.3pr14](#)

8.5.2 Methods

8.5.3 addCollection(collectionName as String, Options as Integer = 0) as Boolean

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Adds a specified font collection to the font manager with a given set of options.

Notes: collectionName: The collection to add.

Options: Pass NSFontCollectionApplicationOnlyMask to make the collection available only to the application.

Return true if the font collection was successfully added; otherwise, false.

8.5.4 addFontDescriptorsToCollection(descriptors() as NSFontDescriptorMBS, collectionName as String)

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Adds an array of font descriptors to the specified font collection.

Example:

```
dim fontManager as NSFontManagerMBS = NSFontManagerMBS.sharedFontManager

dim collectionNames() as string = fontManager.collectionNames
dim collectionName as string = collectionNames(collectionNames.Ubound) // get last one

dim fonts() as NSFontDescriptorMBS = fontManager.fontDescriptorsInCollection(collectionName)

// add Helvetica to this collection

dim font as NSFontMBS = NSFontMBS.fontWithName("Helvetica", 12)
dim fontDescriptor as NSFontDescriptorMBS = font.fontDescriptor

fontManager.addFontDescriptorsToCollection(array(fontDescriptor), collectionName)

dim fonts2() as NSFontDescriptorMBS = fontManager.fontDescriptorsInCollection(collectionName)

// fonts2 should have one more entry than fonts
Break
```

Notes: descriptors: The font descriptors to add.

collectionName: The font collection to which descriptors are added.

8.5.5 availableFontFamilies as string()

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns the names of the font families available in the system.

Example:

```
dim n as new NSFontManagerMBS
```

```
dim names() as string = n.availableFontFamilies
```

```
MsgBox Join(names,EndOfLine)
```

Notes: These fonts are in various system font directories.

8.5.6 availableFontNamesMatchingFontDescriptor(descriptor as NSFontDescriptorMBS) as String()

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Returns the names of the fonts that match the attributes in the given font descriptor.

Example:

```
// ask for monospace font trait
Dim traitsAttributes As New Dictionary
traitsAttributes.Value(NSFontDescriptorMBS.NSFontSymbolicTrait) = NSFontDescriptorMBS.NSFontMonoSpaceTrait

// 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-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
```

8.5.7 availableFontNamesWithTraits(traits as Integer) as string()

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Returns the names of the fonts available in the system whose traits are described exactly by the given font trait mask (not the NSFont objects themselves).

Example:

```

dim n as new NSFontManagerMBS
dim names() as string = n.availableFontNamesWithTraits(n.NSBoldFontMask)

MsgBox str(UBound(names)+1)+" fonts: "+Join(names,", ")

```

Notes: traits: The font traits for which to return font names. You specify the desired traits by combining the font trait mask values described in Constants using the bitwise or operator.

Returns the names of the corresponding fonts.

These fonts are in various system font directories.

If fontTraitMask is 0, this method returns all fonts that are neither italic nor bold. This result is the same one you'd get if fontTraitMask were NSUnitalicFontMask | NSUnboldFontMask.

8.5.8 availableFonts as string()

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns the names of the fonts available in the system (not the NSFont objects themselves).

Example:

```

dim n as new NSFontManagerMBS
dim names() as string = n.availableFonts

MsgBox Join(names,EndOfLine)

```

Notes: These fonts are in various system font directories.

8.5.9 availableMembersOfFontFamily(FontFamily as string) as Variant()

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns an array with one entry for each available member of a font family.

Example:

```

dim n as new NSFontManagerMBS
dim members() as Variant = n.availableMembersOfFontFamily("Times")

for each m as Variant in members

```

```
dim member() as Variant = m
```

```
MsgBox "Postscript name: "+member(0)+EndOfLine+"Suffix: "+member(1)+EndOfLine+"Font weight: "+member(2)+EndOfLine+"Font trait: "+member(3)
next
```

Notes: family: The name of a font family, like one that availableFontFamilies returns.

Returns the available members of family.

Each entry of the returned variant array is another variant array with four members, as follows:

0. The PostScript font name, as a string.
1. The part of the font name used in the font panel that's not the font name, as a string. This value is not localized—for example, "Roman", "Italic", or "Bold".
2. The font's weight, as a double.
3. The font's traits, as a double.

The members of the family are arranged in the font panel order (narrowest to widest, lightest to boldest, plain to italic).

8.5.10 collectionNames as string()

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns the names of the currently loaded font collections.

Example:

```
dim n as new NSFontManagerMBS
dim names() as string = n.collectionNames
```

```
MsgBox Join(names,EndOfLine)
```

Notes: The names of the current font collections.

8.5.11 Constructor

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Initializes the object with the shared instance of the font manager for the application, creating it if necessary.

8.5.12 convertAttributes(dic as dictionary) as dictionary

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Converts attributes in response to an object initiating an attribute change, typically the Font panel or Font menu.

8.5.13 convertFont(font as NSFontMBS) as NSFontMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Converts the given font according to the object that initiated a font change, typically the Font panel or Font menu.

8.5.14 convertFontToFace(font as NSFontMBS, face as string) as NSFontMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Returns a font whose traits are as similar as possible to those of the given font except for the typeface, which is changed to the given typeface.

Example:

```

dim fontManager as new NSFontManagerMBS
// you have a font
dim font as NSFontMBS = NSFontMBS.fontWithName("Helvetica", 12)

// change font face
font = fontManager.convertFontToFace(font, "Helvetica-BoldOblique")

MsgBox font.fontName // shows Helvetica-BoldOblique

```

Notes: Font: The font whose traits are matched.

face: The new typeface; a fully specified family-face name, such as Helvetica-BoldOblique or Times-Roman.

Returns a font with matching traits and the given typeface, or aFont if it can't be converted.

This method attempts to match the weight and posture of aFont as closely as possible. Italic is mapped to Oblique, for example. Weights are mapped based on an approximate numeric scale of 0 to 15.

8.5.15 convertFontToFamily(font as NSFontMBS, family as string) as NSFontMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Returns a font whose traits are as similar as possible to those of the given font except for the font family, which is changed to the given family.

Example:

```
dim fontManager as new NSFontManagerMBS
// you have a font
dim font as NSFontMBS = NSFontMBS.fontWithName("Helvetica", 12)

// change font family
font = fontManager.convertFontToFamily(font, "Arial")

MsgBox font.fontName // shows ArialMT
```

Notes: Font: The font whose traits are matched.

family: The new font family; a generic font name, such as Helvetica or Times.

Returns a font with matching traits and the given family, or aFont if it can't be converted.

This method attempts to match the weight and posture of aFont as closely as possible. Italic is mapped to Oblique, for example. Weights are mapped based on an approximate numeric scale of 0 to 15.

8.5.16 convertFontToHaveTrait(font as NSFontMBS, trait as Integer) as NSFontMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Returns a font whose traits are the same as those of the given font, except that the traits are changed to include the single specified trait.

Example:

```
dim n as new NSFontManagerMBS
dim f as NSFontMBS = NSFontMBS.fontWithName("Times", 12.0)
dim g as NSFontMBS = n.convertFontToHaveTrait(f, n.NSBoldFontMask)
```

```
MsgBox g.fontName // Times-Bold
```

Notes: Font: The font whose traits are matched.

Trait: The new trait; may be any one of the traits described in Constants. Using NSUnboldFontMask or NSUnitalicFontMask removes the bold or italic trait, respectively.

Returns a font with matching traits including the given trait, or font if it can't be converted.

Using NSUnboldFontMask or NSUnitalicFontMask removes the bold or italic trait, respectively.

8.5.17 convertFontToNotHaveTrait(font as NSFontMBS, trait as Integer) as NSFontMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Returns an NSFont object with the same traits as the given font, except for the traits in the given font trait mask, which are removed.

Example:

```
dim n as new NSFontManagerMBS
dim f as NSFontMBS = NSFontMBS.fontWithName("Times-Bold", 12.0)
dim g as NSFontMBS = n.convertFontToHaveTrait(f, n.NSUnBoldFontMask)
```

```
MsgBox g.fontName // Times-Roman
```

Notes: Font: The font whose traits are matched.

trait: The mask for the traits to remove, created using the bitwiseOr operator to combine the traits described in Constants. Using NSUnboldFontMask or NSUnitalicFontMask removes the bold or italic trait, respectively.

Returns a font with matching traits minus the given traits, or font if it can't be converted.

8.5.18 convertFontToSize(font as NSFontMBS, size as Double) as NSFontMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Returns an NSFont object whose traits are the same as those of the given font, except for the size, which is changed to the given size.

Example:

```
dim fontManager as new NSFontManagerMBS
// you have a font
```

```

dim font as NSFontMBS = NSFontMBS.fontWithName("Helvetica", 12)

// change font size
font = fontManager.convertFontToSize(font, 20)

MsgBox str(font.pointSize)

```

Notes: Font: The font whose traits are matched.
size: The new font size.

Returns a font with matching traits except in the new size, or aFont if it can't be converted.

8.5.19 convertFontTraits(traits as Integer) as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Converts font traits to a new traits mask value.

Notes: traits: The current font traits.

Returns the new traits mask value to be used by convertFont:.

This method is intended to be invoked to query the font traits while the action message (usually changeFont:) is being invoked when the current font action is either NSAddTraitFontAction or NSRemoveTraitFontAction.

Available in Mac OS X v10.5 and later.

8.5.20 convertWeightOfFont(font as NSFontMBS, up as boolean) as NSFontMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Returns an NSFont object whose weight is greater or lesser than that of the given font, if possible.

Example:

```

dim n as new NSFontManagerMBS
dim f as NSFontMBS = NSFontMBS.fontWithName("Times", 12.0)
dim g as NSFontMBS = n.convertWeightOfFont(f, true)

MsgBox g.fontName // Times-Bold

```

Notes: up: If true, a heavier font is returned; if it's false, a lighter font is returned.
 Font: The font whose weight is increased or decreased.

Returns a font with matching traits except for the new weight, or font if it can't be converted.

8.5.21 fontDescriptorsInCollection(collectionName as String) as NSFontDescriptorMBS()

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Returns an array of the font descriptors in the collection specified by the given collection name.
Example:

```
dim fontManager as NSFontManagerMBS = NSFontManagerMBS.sharedFontManager
dim collectionNames() as string = fontManager.collectionNames
dim collectionName as string = collectionNames(collectionNames.Ubound) // get last one
dim fonts() as NSFontDescriptorMBS = fontManager.fontDescriptorsInCollection(collectionName)
Break
```

8.5.22 fontHasTraits(fontName as string, Traits as Integer) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the given font has all the specified traits.
Example:

```
dim n as NSFontMBS = NSFontMBS.boldSystemFontOfSize(12)
dim m as new NSFontManagerMBS

dim isBold as Boolean = m.fontHasTraits(n.fontName, m.NSBoldFontMask)
dim isItalic as Boolean = m.fontHasTraits(n.fontName, m.NSItalicFontMask)
```

```
MsgBox "is bold: "+str(isBold)+"
EndOfLine+"is italic: "+str(isItalic)
```

Notes: typeface: The name of the font.
 fontTraitMask: The font traits to test, specified by combining the font trait mask values described in Constants using the bitwiseOR operation.

Returns true if the font named typeface has all the traits specified in fontTraitMask; false if it doesn't.

Using `NSUnboldFontMask` returns true if the font is not bold, false otherwise. Using `NSUnitalicFontMask` returns true if the font is not italic, false otherwise.

8.5.23 `isMultiple` as boolean

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the last font selection recorded has multiple fonts.

Notes: Returns true if the last font selection recorded has multiple fonts; false if it's a single font.

8.5.24 `orderFrontFontPanel`

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: This action method opens the Font panel by sending it an `orderFront` message, creating the Font panel if necessary.

8.5.25 `orderFrontStylesPanel`

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: This action method opens the Font styles panel.

8.5.26 `removeCollection(collectionName as String) as Boolean`

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Removes the specified font collection.

Notes: Returns true on success and false on failure.

8.5.27 `removeFontDescriptorFromCollection(descriptor as NSFontDescriptorMBS, collectionName as String)`

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Removes the specified font descriptor from the specified collection.

Notes: `descriptor`: The font descriptor to remove.

`collection`: The font collection from which to remove the descriptor.

8.5.28 selectedFont as NSFontMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Returns the last font recorded.

8.5.29 setSelectedAttributes(dic as dictionary, isMultiple as boolean)

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Informs the paragraph and character formatting panels when text in a selection has changed attributes.

Notes: dic: The new attributes.

isMultiple: If true, informs the panel that multiple fonts or attributes are enclosed within the selection.

This method is used primarily by NSTextView.

Available in Mac OS X v10.3 and later.

8.5.30 setSelectedFont(font as NSFontMBS, isMultiple as boolean)

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Records the given font as the currently selected font and updates the Font panel to reflect this.

Notes: font: The font to set as selected.

isMultiple: If true, the Font panel indicates that more than one font is contained in the selection; if false, it does not.

An object that manipulates fonts should invoke this method whenever it becomes first responder and whenever its selection changes. It shouldn't invoke this method in the process of handling a changeFont message, as this causes the font manager to lose the information necessary to effect the change. After all fonts have been converted, the font manager itself records the new selected font.

8.5.31 sharedFontManager as NSFontManagerMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Returns the shared instance of the font manager for the application, creating it if necessary.

8.5.32 traitsOfFont(font as NSFontMBS) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Returns the traits of the given font.

Example:

```
dim n as NSFontMBS = NSFontMBS.boldSystemFontOfSize(12)
dim m as new NSFontManagerMBS

// shows 2 which is m.NSBoldFontMask
MsgBox str(m.traitsOfFont(n))
```

Notes: Font: The font whose traits are returned.

Returns the font traits, returned as a mask created by combining values listed in Constants with the bitwiseOR operation.

8.5.33 weightOfFont(font as NSFontMBS) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Returns a rough numeric measure the weight of the given font.

Example:

```
dim n as NSFontMBS = NSFontMBS.boldSystemFontOfSize(12)
dim m as new NSFontManagerMBS

MsgBox str(m.weightOfFont(n))
```

Notes: Font: The font whose weight is returned.

A rough numeric measure the weight of the given font, where 0 indicates the lightest possible weight, 5 indicates a normal or book weight, and 9 or more indicates a bold or heavier weight.

8.5.34 Properties

8.5.35 Handle as Integer

Plugin Version: 15.4, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.

Notes: (Read and Write property)

8.5.36 Enabled as boolean

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Whether the font conversion system's user interface items (the Font panel and Font menu items) are enabled.

Notes: True if the font conversion system's user interface items (the Font panel and Font menu items) are enabled; false if they're not.

(Read and Write computed property)

8.5.37 Constants

Constants

Constant	Value	Description
<code>NSAddTraitFontAction</code>	2	One of the constants for use with <code>modifyFont</code> method. Converts the font to have an additional trait using <code>convertFonttoHaveTrait</code> .
<code>NSBoldFontMask</code>	2	One of the constants for use with mask of traits assigned to a font. A mask that specifies a bold font.
<code>NSCompressedFontMask</code>	<code>&h00000200</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies a compressed font.
<code>NSCondensedFontMask</code>	<code>&h00000040</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies a condensed font.
<code>NSExpandedFontMask</code>	<code>&h00000020</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies an expanded font.
<code>NSFixedPitchFontMask</code>	<code>&h00000400</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies a fixed pitch font.
<code>NSFontCollectionApplicationOnlyMask</code>	1	One of the constants for options accepted by <code>addCollection</code> . Makes the collection available only to the application. This option is not implemented.
<code>NSHeavierFontAction</code>	5	One of the constants for use with <code>modifyFont</code> method. Converts the font to a heavier weight using <code>convertWeightofFonttoHeavier</code> .
<code>NSItalicFontMask</code>	1	One of the constants for use with mask of traits assigned to a font. A mask that specifies an italic font.
<code>NSLighterFontAction</code>	6	One of the constants for use with <code>modifyFont</code> method. Converts the font to a lighter weight using <code>convertWeightofFonttoLighter</code> .
<code>NSNarrowFontMask</code>	<code>&h00000010</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies a narrow font.
<code>NSNoFontChangeAction</code>	0	One of the constants for use with <code>modifyFont</code> method. No action; the font is returned unchanged.
<code>NSNonStandardCharacterSetFontMask</code>	8	One of the constants for use with mask of traits assigned to a font. A mask that specifies a font that uses a non-standard character set.
<code>NSPosterFontMask</code>	<code>&h00000100</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies a poster-style font.
<code>NSRemoveTraitFontAction</code>	7	One of the constants for use with <code>modifyFont</code> method. Converts the font to remove a trait using <code>convertFonttoNotHaveTrait</code> .
<code>NSSizeDownFontAction</code>	4	One of the constants for use with <code>modifyFont</code> method. Converts the font to a smaller size using <code>convertFonttoSize</code> .
<code>NSSizeUpFontAction</code>	3	One of the constants for use with <code>modifyFont</code> method. Converts the font to a larger size using <code>convertFonttoSize</code> .
<code>NSSmallCapsFontMask</code>	<code>&h00000080</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies a small-caps font.
<code>NSUnboldFontMask</code>	4	One of the constants for use with mask of traits assigned to a font. A mask that specifies a font that is not bold.
<code>NSUnitalicFontMask</code>	<code>&h01000000</code>	One of the constants for use with mask of traits assigned to a font. A mask that specifies a font that is not italic.
<code>NSViaPanelFontAction</code>	1	One of the constants for use with <code>modifyFont</code> method. Converts the font according to the <code>NSFontPanel</code> method <code>panelConvertFont</code> .

8.6 class NSFontPanelMBS

8.6.1 class NSFontPanelMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: The NSFontPanel class implements the Font panel—a user interface object that displays a list of available fonts, letting the user preview them and change the font used to display text.

Example:

```
NSFontPanelMBS.sharedFontPanel.Show
```

Notes: The actual changes are made through conversion messages sent to the shared NSFontManager instance. There's only one Font panel for each application.

Subclass of the NSPanelMBS class.

Blog Entries

- [RTF functions in MBS Plugins](#)
- [MBS Real Studio Plugins, version 12.5pr6](#)

8.6.2 Methods

8.6.3 Constructor

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: The Constructor.

8.6.4 convertAttributes(old as dictionary) as dictionary

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Converts attributes in response to an object initiating an attribute change, typically the Font panel.

Notes: Call this function only in the changeAttributes event.

8.6.5 convertFont(oldFont as NSFontMBS) as NSFontMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Converts the given font according to the object that initiated a font change, typically the Font panel.

Notes: Call this function only in the `ChangeFont` event.

8.6.6 Destructor

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: The destructor.

8.6.7 `panelConvertFont(font as NSFontMBS) as NSFontMBS`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Converts the specified font using the settings in the receiver, with the aid of the shared `NSFontManager` if necessary.

Notes: `Font`: The font to be converted.

Returns the converted font, or `aFont` itself if it can't be converted.

For example, if `aFont` is Helvetica Oblique 12.0 point and the user has selected the Times font family (and nothing else) in the Font panel, the font returned is Times Italic 12.0 point.

8.6.8 `reloadDefaultFontFamilies`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Triggers a reload to the default state, so that the delegate is called.

Notes: This reloading provides the delegate opportunity to scrutinize the default list of fonts to be displayed in the panel.

8.6.9 `setPanelFont(font as NSFontMBS, isMultiple as boolean)`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Sets the selected font in the receiver to the specified font.

Notes: `Font`: The font to be selected.

`flag`: If false, selects the specified font; otherwise selects no font and displays a message in the preview area indicating that multiple fonts are selected.

You normally don't use this method directly; instead, you send `setSelectedFont` to the shared `NSFontManager`, which in turn invokes this method.

8.6.10 sharedFontPanel as NSFontPanelMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Returns the single NSFontPanel instance for the application, creating it if necessary.

Example:

```
NSFontPanelMBS.sharedFontPanel.Show
```

8.6.11 sharedFontPanelExists as boolean

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Returns true if the shared Font panel has been created, false if it hasn't.

8.6.12 worksWhenModal as boolean

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the receiver allows fonts to be changed in modal windows and panels.

8.6.13 Properties

8.6.14 accessoryView as NSViewMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: The accessory view.

Notes: Establishes the specified view as the receiver's accessory view, allowing you to add custom controls to your application's Font panel without having to create a subclass.

(Read and Write computed property)

8.6.15 Enabled as boolean

Plugin Version: 9.8, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver's Set button is enabled.

Notes: The receiver continues to reflect the font of the selection for cooperating text objects regardless of

this setting.
(Read and Write computed property)

8.6.16 Events

8.6.17 `changeAttributes`

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

Function: The event called whenever attributes in the font panel changed.

Notes: Use `convertAttributes` to know what changed.

8.6.18 `changeFont`

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

Function: The event called whenever the font in the font panel changed.

Notes: Use `ConvertFont` to know what changed.

8.6.19 `validModesForFontPanel` as Integer

Plugin Version: 12.5, Platform: macOS, Targets: .

Function: Returns the mode mask corresponding to the expected font panel mode.

Notes: The mode masks are defined in constants.

By default the plugin returns `NSFontPanelAllModesMask`.

8.6.20 Constants

Mode Constants

Constant	Value	Description
NSFontPanelAllEffectsModeMask	&hFFF00	Display all the effects user interface items.
NSFontPanelAllModesMask	&hFFFFFFFF	Display all the available adornments.
NSFontPanelCollectionModeMask	4	Display the font collections column.
NSFontPanelDocumentColorEffectModeMask	2048	Display the document color button.
NSFontPanelFaceModeMask	1	Display the typeface column.
NSFontPanelShadowEffectModeMask	4096	Display the shadow effects button.
NSFontPanelSizeModeMask	2	Display the font size column.
NSFontPanelStandardModesMask	&hFFFF	Display the standard default font panel—that is, including face, and size columns.
NSFontPanelStrikethroughEffectModeMask	512	Display the strike-through popup menu.
NSFontPanelTextColorEffectModeMask	1024	Display the text color button.
NSFontPanelUnderlineEffectModeMask	256	Display the underline popup menu.

8.7 class `NSLayoutManagerMBS`

8.7.1 class `NSLayoutManagerMBS`

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: An `NSLayoutManager` object coordinates the layout and display of characters held in an `NSTextStorage` object.

Notes: It maps Unicode character codes to glyphs, sets the glyphs in a series of `NSTextContainer` objects, and displays them in a series of `NSTextView` objects. In addition to its core function of laying out text, an `NSLayoutManager` object coordinates its `NSTextView` objects, provides services to those text views to support `NSRulerView` instances for editing paragraph styles, and handles the layout and display of text attributes not inherent in glyphs (such as underline or strikethrough). You can create a subclass of `NSLayoutManager` to handle additional text attributes, whether inherent or not.

Text Antialiasing

`NSLayoutManager` provides the threshold for text antialiasing. It looks at the `AppleAntiAliasingThreshold` default value. If the font size is smaller than or equal to this threshold size, the text is rendered aliased by `NSLayoutManager`. You can change the threshold value from the Appearance pane of System Preferences.

Thread Safety of `NSLayoutManager`

Generally speaking, a given layout manager (and associated objects) should not be used in more than one block, operation, or thread at a time. Most layout managers are used on the main thread, since it is the main thread on which their text views are displayed, and since background layout occurs on the main thread. If it is intended that a layout manager should be used on a background thread, first make sure that text views associated with that layout manager (if any) are not displayed while the layout manager is being used on the background thread, and, second, turn off background layout for that layout manager while it is being used on the background thread.

Noncontiguous Layout

Noncontiguous layout is an optional layout manager behavior new in Mac OS X v10.5. Previously, both glyph generation and layout were always performed, in order, from the beginning to the end of the document. When noncontiguous layout is turned on, however, the layout manager gains the option of performing glyph generation or layout for one portion of the document without having done so for previous sections. This can provide significant performance improvements for large documents.

Noncontiguous layout is not turned on automatically because direct clients of `NSLayoutManager` typically have relied on the previous behavior—for example, by forcing layout for a given glyph range, and then assuming that previous glyphs would therefore be laid out. Clients who use `NSLayoutManager` only indirectly—for example, those who use `NSTextView` without directly calling the underlying layout manager—can usually turn on noncontiguous layout without difficulty. Clients using `NSLayoutManager` directly need to examine their usage before turning on noncontiguous layout.

To turn on noncontiguous layout, use `AllowsNonContiguousLayout`. In addition, see the other methods in

”Managing Noncontiguous Layout,” many of which enable you to ensure that glyph generation and layout are performed for specified portions of the text. The behavior of a number of other layout manager methods is affected by the state of noncontiguous layout, as noted in the discussion sections of those method descriptions.

So far the plugin implements a small subset of the functions in NSLayoutManager. If you miss a function, please email us and we can check whether we can add it for you.

Blog Entries

- [MBS Xojo Plugins, version 22.0pr6](#)
- [MBS Releases the MBS Xojo / Real Studio plug-ins in version 16.4](#)
- [MBS Xojo / Real Studio Plugins, version 16.4pr7](#)
- [MBS Real Studio Plugins, version 13.1pr8](#)
- [MBS Real Studio Plugins, version 12.5pr7](#)
- [MBS Real Studio Plugins, version 12.5pr5](#)
- [MBS Real Studio Plugins, version 12.4pr4](#)
- [Show invisible characters in NSTextView/TextArea](#)
- [MBS Real Studio Plugins, version 12.1pr5](#)

8.7.2 Methods

8.7.3 `addTextContainer(container as NSTextContainerMBS)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Appends the given text container to the series of text containers where the receiver arranges text.

Notes: container: The text container to append.

Invalidates glyphs and layout as needed, but doesn’t perform glyph generation or layout.

8.7.4 `characterIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS, byref partialFraction as Double) as Integer`

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

Function: Returns the index of the character falling under the given point, expressed in the given container’s coordinate system.

Notes: point: The point to test.

container: The text container within which the point is tested.

`partialFraction`: A fraction of the distance from the insertion point, logically before the given character to the next one.

Returns the index of the character falling under point.

Analogous to `glyphIndexForPoint:inTextContainer`, but expressed in character index terms. The method returns the index of the character falling under point, expressed in coordinate system of container; if no character is under the point, the nearest character is returned, where nearest is defined according to the requirements of selection by mouse. However, this is not simply equivalent to taking the result of the corresponding glyph index method and converting it to a character index, because in some cases a single glyph represents more than one selectable character, for example an fi ligature glyph. In that case, there is an insertion point within the glyph, and this method returns one character or the other, depending on whether the specified point lies to the left or the right of that insertion point.

In general, this method returns only character indexes for which there is an insertion point. The `partialFraction` is a fraction of the distance from the insertion point, logically before the given character to the next one, which may be either to the right or to the left depending on directionality.

Available in OS X v10.6 and later.

8.7.5 Constructor

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Creates new `NSLayoutManager` object.

8.7.6 `glyphIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS) as Integer`

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

Function: This method is a primitive for `glyphIndexForPoint`. You should always call the main method, not the primitives.

See also:

- 8.7.7 `glyphIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS, byref partialFraction as Double) as Integer` 357

8.7.7 glyphIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS, byref partialFraction as Double) as Integer

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

Function: Returns the index of the glyph falling under the given point, expressed in the given container's coordinate system.

Notes: point: The point for which to return the glyph, in coordinates of container.

container: The container in which the returned glyph is laid out.

partialFraction: On output, the fraction of the distance between the location of the glyph returned and the location of the next glyph.

Returns the index of the glyph falling under the given point, expressed in the given container's coordinate system.

If no glyph is under point, the nearest glyph is returned, where nearest is defined according to the requirements of selection by mouse. Clients who wish to determine whether the point actually lies within the bounds of the glyph returned should follow this with a call to boundingRectForGlyphRange and test whether the point falls in the rectangle returned by that method. If partialFraction is non-NULL, it returns by reference the fraction of the distance between the location of the glyph returned and the location of the next glyph.

For purposes such as dragging out a selection or placing the insertion point, a partial percentage less than or equal to 0.5 indicates that point should be considered as falling before the glyph index returned; a partial percentage greater than 0.5 indicates that it should be considered as falling after the glyph index returned. If the nearest glyph doesn't lie under point at all (for example, if point is beyond the beginning or end of a line), this ratio is 0 or 1.

If the glyph stream contains the glyphs "A" and "b", with the width of "A" being 13 points, and the user clicks at a location 8 points into "A", partialFraction is 8/13, or 0.615. In this case, the point given should be considered as falling between "A" and "b" for purposes such as dragging out a selection or placing the insertion point.

Performs glyph generation and layout if needed.

As part of its implementation, this method calls fractionOfDistanceThroughGlyphForPoint and glyphIndexForPoint. To change this method's behavior, override those two methods instead of this one.

See also:

- 8.7.6 glyphIndexForPoint(point as NSPointMBS, container as NSTextContainerMBS) as Integer 356

8.7.8 glyphRangeForTextContainer(container as NSTextContainerMBS) as NSRangeMBS

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

Function: Returns the range of glyphs laid out within the given text container.

Notes: This is a less efficient method than the similar `textContainerForGlyphAtIndex`. Performs glyph generation and layout if needed.

8.7.9 `lineFragmentRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS) as NSRectMBS`

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

Function: Returns the rectangle for the line fragment in which the given glyph is laid out and (optionally), by reference, the whole range of glyphs that are in that fragment.

Notes: `glyphIndex`: The glyph for which to return the line fragment rectangle.
`effectiveGlyphRange`: On output, the range for all glyphs in the line fragment.

Returns the line fragment in which the given glyph is laid out.

This method causes glyph generation and layout for the line fragment containing the specified glyph, or if noncontiguous layout is not enabled, for all of the text up to and including that line fragment.

Line fragment rectangles are always in container coordinates.

Overriding this method is not recommended. If the the line fragment rectangle needs to be modified, that should be done at the typesetter level or by calling `setLineFragmentRect:forGlyphRange`.

See also:

- 8.7.10 `lineFragmentRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS, withoutAdditionalLayout as boolean) as NSRectMBS` 358

8.7.10 `lineFragmentRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS, withoutAdditionalLayout as boolean) as NSRectMBS`

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

Function: Returns the line fragment rectangle containing the glyph at the given glyph index.

Notes: `glyphIndex`: The glyph for which to return the line fragment rectangle.

`effectiveGlyphRange`: On output, the range for all glyphs in the line fragment.

`withoutAdditionalLayout`: If true, glyph generation and layout are not performed, so this option should not be used unless layout is known to be complete for the range in question, or unless noncontiguous layout is enabled; if false, both are performed as needed.

Returns the line fragment in which the given glyph is laid out.

This method is primarily for use from within NSTypesetter, after layout is complete for the range in question, but before the layout manager's call to NSTypesetter has returned. In that case glyph and layout holes have not yet been recalculated, so the layout manager does not yet know that layout is complete for that range, and this variant must be used.

Overriding this method is not recommended. If the the line fragment rectangle needs to be modified, that should be done at the typesetter level or by calling setLineFragmentRect.

See also:

- 8.7.9 lineFragmentRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS) as NSRectMBS 358

8.7.11 lineFragmentUsedRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS) as NSRectMBS

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

Function: Returns the usage rectangle for the line fragment in which the given glyph is laid and (optionally) by reference the whole range of glyphs that are in that fragment.

Notes: glyphIndex: The glyph for which to return the line fragment used rectangle.
effectiveGlyphRange: On output, the range for all glyphs in the line fragment.

Returns the used rectangle for the line fragment in which the given glyph is laid out.

This method causes glyph generation and layout for the line fragment containing the specified glyph, or if noncontiguous layout is not enabled, up to and including that line fragment.

Line fragment used rectangles are always in container coordinates.

Overriding this method is not recommended. If the the line fragment used rectangle needs to be modified, that should be done at the typesetter level or by calling setLineFragmentRect.

See also:

- 8.7.12 lineFragmentUsedRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS, withoutAdditionalLayout as boolean) as NSRectMBS 359

8.7.12 lineFragmentUsedRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS, withoutAdditionalLayout as boolean) as NSRectMBS

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

Function: Returns the usage rectangle for the line fragment in which the given glyph is laid and (optionally) by reference the whole range of glyphs that are in that fragment.

Notes: `glyphIndex`: The glyph for which to return the line fragment used rectangle.

`effectiveGlyphRange`: On output, the range for all glyphs in the line fragment.

`withoutAdditionalLayout`: If true, glyph generation and layout are not performed, so this option should not be used unless layout is known to be complete for the range in question, or unless noncontiguous layout is enabled; if false, both are performed as needed.

Returns the used rectangle for the line fragment in which the given glyph is laid out.

This method causes glyph generation and layout for the line fragment containing the specified glyph, or if noncontiguous layout is not enabled, up to and including that line fragment.

Line fragment used rectangles are always in container coordinates.

Overriding this method is not recommended. If the the line fragment used rectangle needs to be modified, that should be done at the typesetter level or by calling `setLineFragmentRect`.

See also:

- 8.7.11 `lineFragmentUsedRectForGlyphAtIndex(glyphIndex as Integer, byref effectiveRange as NSRangeMBS) as NSRectMBS` 359

8.7.13 `locationForGlyphAtIndex(glyphIndex as Integer) as NSPointMBS`

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

Function: Returns the location for the given glyph within its line fragment.

Notes: `glyphIndex`: The glyph whose location is returned.

Returns the location of the given glyph.

If the given glyph does not have an explicit location set for it (for example, if it is part of (but not first in) a sequence of nominally spaced characters), the location is calculated by glyph advancements from the location of the most recent preceding glyph with a location set.

Glyph locations are relative to their line fragment rectangle's origin. The line fragment rectangle in turn is defined in the coordinate system of the text container where it resides.

This method causes glyph generation and layout for the line fragment containing the specified glyph, or if noncontiguous layout is not enabled, up to and including that line fragment.

8.7.14 rangeOfNominallySpacedGlyphsContainingIndex(glyphIndex as Integer) as NSRangeMBS

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

Function: Returns the range for the glyphs around the given glyph that can be displayed using only their advancements from the font, without pairwise kerning or other adjustments to spacing.

Notes: glyphIndex: Index of the glyph to test.

Returns the range of nominally spaced glyphs.

The range returned begins with the first glyph, counting back from glyphIndex, that has a location set, and it continues up to, but does not include, the next glyph that has a location set.

Performs glyph generation and layout if needed.

8.7.15 rectArrayForCharacterRange(charRange as NSRangeMBS, selCharRange as NSRangeMBS, container as NSTextContainerMBS, byref rectCount as Integer) as NSRectMBS()

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

Function: Returns an array of rectangles and, by reference, the number of such rectangles, that define the region in the given container enclosing the given character range.

Notes: charRange: The character range for which to return rectangles.

selCharRange: Selected characters within charRange, which can affect the size of the rectangles; it must be equal to or contain charRange. If the caller is interested in this more from an enclosing point of view rather than a selection point of view, pass { NSNotFound, 0 } as the selected range.

container: The text container in which the text is laid out.

rectCount: The number of rectangles returned.

Returns the array of rectangles enclosing the given range.

These rectangles can be used to draw the text background or highlight for the given range of characters. If a selected range is given in selCharRange, the rectangles returned are correct for drawing the selection. Selection rectangles are generally more complicated than enclosing rectangles and supplying a selected range is the clue this method uses to determine whether to go to the trouble of doing this special work.

The number of rectangles returned isn't necessarily the number of lines enclosing the specified range. Contiguous lines can share an enclosing rectangle, and lines broken into several fragments have a separate enclosing rectangle for each fragment.

These rectangles don't necessarily enclose glyphs that draw outside their line fragment rectangles; use `boundingRectForGlyphRange` to determine the area that contains all drawing performed for a range of glyphs.

Performs glyph generation and layout if needed.

8.7.16 `rectArrayForGlyphRange(glyphRange as NSRangeMBS, selGlyphRange as NSRangeMBS, container as NSTextContainerMBS, byref rectCount as Integer) as NSRectMBS()`

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

Function: Returns an array of rectangles and, by reference, the number of such rectangles, that define the region in the given container enclosing the given glyph range.

Notes: `glyphRange`: The glyph range for which to return rectangles.

`selGlyphRange`: Selected glyphs within `glyphRange`, which can affect the size of the rectangles; it must be equal to or contain `glyphRange`. If the caller is interested in this more from an enclosing point of view rather than a selection point of view, pass `{ NSNotFound, 0 }` as the selected range.

`container`: The text container in which the text is laid out.

`rectCount`: The number of rectangles returned.

Returns the array of rectangles enclosing the given range.

These rectangles can be used to draw the text background or highlight for the given range of characters. If a selected range is given in `selGlyphRange`, the rectangles returned are correct for drawing the selection. Selection rectangles are generally more complicated than enclosing rectangles and supplying a selected range is the clue this method uses to determine whether to go to the trouble of doing this special work.

The number of rectangles returned isn't necessarily the number of lines enclosing the specified range. Contiguous lines can share an enclosing rectangle, and lines broken into several fragments have a separate enclosing rectangle for each fragment.

The purpose of this method is to calculate line rectangles for drawing the text background and highlighting. These rectangles don't necessarily enclose glyphs that draw outside their line fragment rectangles; use `boundingRectForGlyphRange` to determine the area that contains all drawing performed for a range of glyphs.

Performs glyph generation and layout if needed.

8.7.17 `removeTextContainerAtIndex(index as Integer)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Removes the text container at the given index and invalidates the layout as needed.

Notes: index: The index of the text container to remove.

This method invalidates glyph information as needed.

8.7.18 replaceGlyphAtIndex(glyphIndex as Integer, newGlyph as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Replaces the glyph at the given index with a new glyph.

Notes: glyphIndex: Index of the glyph to replace.

newGlyph: The new glyph.

Doesn't alter the glyph-to-character mapping or invalidate layout information. The character index of the glyph is assumed to remain the same (although it can, of course, be set explicitly if needed).

This method is for use by the glyph-generation mechanism and doesn't perform any invalidation or generation of the glyphs or layout. This method should be invoked only during glyph generation and typesetting, in almost all cases only by the glyph generator or typesetter. For example, a custom glyph generator or typesetter might invoke it.

8.7.19 replaceTextStorage(newTextStorage as NSTextStorageMBS)

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Replaces the NSTextStorage object for the group of text-system objects containing the receiver with the given text storage object.

Notes: All NSLayoutManager objects sharing the original NSTextStorage object then share the new one. This method makes all the adjustments necessary to keep these relationships intact, unlike setting textStorage property.

8.7.20 setCharacterIndex(charIndex as Integer, glyphIndex as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Sets the index of the character corresponding to the glyph at the given glyph index.

Notes: charIndex: The index to set.

glyphIndex: The glyph corresponding to the character whose index is set. The glyph must already be present.

This method is for use by the glyph-generation mechanism and doesn't perform any invalidation or generation of the glyphs or layout. This method should be invoked only during glyph generation and typesetting,

in almost all cases only by the glyph generator or typesetter. For example, a custom glyph generator or typesetter might invoke it.

8.7.21 `setExtraLineFragmentRect(fragmentRect as NSRectMBS, usedRect as NSRectMBS, TextContainer as NSTextContainerMBS)`

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Sets the bounds and container for the extra line fragment.

Notes: `fragmentRect`: The rectangle to set.

`usedRect`: Indicates where the insertion point is drawn.

`TextContainer`: The text container where the rectangle is to be laid out.

The extra line fragment is used when the text backing ends with a hard line break or when the text backing is totally empty, to define the extra line which needs to be displayed at the end of the text. If the text backing is not empty and does not end with a hard line break, this should be set to `NSRectMBS.Zero` and `nil`.

Line fragment rectangles and line fragment used rectangles are always in container coordinates.

This method is used by the layout mechanism and should be invoked only during typesetting, in almost all cases only by the typesetter. For example, a custom typesetter might invoke it.

8.7.22 `setLineFragmentRect(fragmentRect as NSRectMBS, glyphRange as NSRangeMBS, usedRect as NSRectMBS)`

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Associates the given line fragment bounds with the given range of glyphs.

Notes: `fragmentRect`: The rectangle of the line fragment.

`glyphRange`: The range of glyphs to be associated with `fragmentRect`.

`usedRect`: The portion of `fragmentRect` that actually contains glyphs or other marks that are drawn (including the text container, line fragment padding). Must be equal to or contained within `fragmentRect`.

The typesetter must specify the text container first with `setTextContainer`, and it sets the exact positions of the glyphs afterwards with `setLocation`.

In the course of layout, all glyphs should end up being included in a range passed to this method, but only glyphs that start a new line fragment should be at the start of such ranges.

Line fragment rectangles and line fragment used rectangles are always in container coordinates.

This method is used by the layout mechanism and should be invoked only during typesetting, in almost all cases only by the typesetter. For example, a custom typesetter might invoke it.

8.7.23 `usedRectForTextContainer(container as NSTextContainerMBS) as NSRectMBS`

Plugin Version: 17.2, Platform: macOS, Targets: Desktop only.

Function: Returns the bounding rectangle for the glyphs laid out in the given text container.

Notes: Returns the text container's currently used area, which determines the size that the view would need to be in order to display all the glyphs that are currently laid out in the container. This causes neither glyph generation nor layout.

Used rectangles are always in container coordinates.

8.7.24 Properties

8.7.25 `allowsNonContiguousLayout` as boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Whether noncontiguous layout is enabled.

Notes: Setting to true allows but does not require the layout manager to use noncontiguous layout, and the layout manager may in fact not do so, depending on its configuration.

(Read and Write property)

8.7.26 `attributedString` as NSAttributedStringMBS

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Returns the text storage object from which the NSGlyphGenerator object procures characters for glyph generation.

Notes: This method is part of the NSGlyphStorage protocol, for use by the glyph generator. For NSLayoutManager the attributed string is equivalent to the text storage.

Available in Mac OS X v10.5 and later.

(Read only property)

8.7.27 `backgroundLayoutEnabled` as boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver generates glyphs and lays them out when the application's run loop is idle.

Notes: If true, background layout is enabled; if false, the receiver performs glyph generation and layout

only when necessary.

(Read and Write property)

8.7.28 font as NSFontMBS

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: The font for the invisible character drawing.

Notes: Use nil font for using the font of the current text.

This method is only available if the `NSLayoutManagerMBS` object has been created with new `NSLayoutManagerMBS`, so the plugin can use the special `NSLayoutManager` subclass with support for invisible character drawing.

(Read and Write property)

8.7.29 Handle as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.

Notes: (Read and Write property)

8.7.30 hasNonContiguousLayout as boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Whether the layout manager currently has any areas of noncontiguous layout.

Notes: There may be times at which there is no noncontiguous layout, such as when layout is complete; this method enables the layout manager to report that to clients.

(Read only property)

8.7.31 hyphenationFactor as Double

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The threshold controlling when hyphenation is done.

Notes: factor: The hyphenation factor, ranging from 0.0 to 1.0. By default, the value is 0.0, meaning hyphenation is off. A factor of 1.0 causes hyphenation to be attempted always.

Whenever (width of the real contents of the line) / (the line fragment width) is below factor, hyphenation is attempted when laying out the line. Hyphenation slows down text layout and increases memory usage, so

it should be used sparingly.

May be overridden on a per-paragraph basis by the NSParagraphStyle method `hyphenationFactor`.
(Read and Write property)

8.7.32 `showInvisibleCharacters` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Whether to show invisible characters.

Notes: This method is only available if the `NSLayoutManagerMBS` object has been created with new `NSLayoutManagerMBS`, so the plugin can use the special `NSLayoutManager` subclass with support for invisible character drawing. (from MBS Plugin)
(Read and Write property)

8.7.33 `showsControlCharacters` as boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Whether to substitute visible glyphs for control characters in layout.

Example:

```
if TargetCocoa then
```

```
  dim t as NSTextViewMBS = TextArea1.NSTextViewMBS
```

```
  dim l as NSLayoutManagerMBS = t.layoutManager
```

```
  l.showsControlCharacters = true
```

```
else
```

```
  // not supported
```

```
break
```

```
end if
```

Notes: If true, the receiver substitutes visible glyphs for control characters if the font and script support it; if false, it doesn't. The default is false.
(Read and Write property)

8.7.34 `showsInvisibleCharacters` as boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Whether to substitute visible glyphs for whitespace and other typically invisible characters in layout.

Example:

```
if TargetCocoa then

dim t as NSTextViewMBS = TextArea1.NSTextViewMBS
dim l as NSLayoutManagerMBS = t.layoutManager

l.showsInvisibleCharacters = true

else
// not supported
break
end if
```

Notes: If true, the receiver substitutes visible glyphs for invisible characters if the font and script support it; if false, it doesn't. The default is false. (from Apple framework)
(Read and Write property)

8.7.35 textColor as NSColorMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The text color for drawing invisible characters.

Notes: This method is only available if the NSLayoutManagerMBS object has been created with new NSLayoutManagerMBS, so the plugin can use the special NSLayoutManager subclass with support for invisible character drawing.
(Read and Write property)

8.7.36 textStorage as NSTextStorageMBS

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The text storage.

Notes: (Read and Write property)

8.7.37 usesFontLeading as Boolean

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the receiver uses the leading provided in the font.

Notes: (Read and Write property)

8.7.38 usesScreenFonts as boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Whether screen fonts to calculate layout and display text.

Notes: If true, the receiver uses screen fonts; if false, it doesn't.

(Read and Write property)

8.7.39 InvisibleCharMapping(character as Integer) as string

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Sets/gets character mapping for invisible character.

Example:

```
dim l as new NSLayoutManagerMBS
// ...
// set tab to map to plus sign
l.InvisibleCharMapping(9) = "+"
// set space to map to star sign
l.InvisibleCharMapping(asc(" ")) = "*"
```

Notes: By default characters are set for endofline, tab and spaces.

Set showInvisibleCharacters to true and put here all the characters you need.

(Read and Write computed property)

8.8 class NSRunLoopMBS

8.8.1 class NSRunLoopMBS

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The NSRunLoop class declares the programmatic interface to objects that manage input sources.

Notes: An NSRunLoop object processes input for sources such as mouse and keyboard events from the window system, NSPort objects, and NSConnection objects. An NSRunLoop object also processes NSTimer events.

Your application cannot either create or explicitly manage NSRunLoop objects. Each NSThread object, including the application's main thread, has an NSRunLoop object automatically created for it as needed. If you need to access the current thread's run loop, you do so with the class method `currentRunLoop`.

Note that from the perspective of NSRunLoop, NSTimer objects are not "input"—they are a special type, and one of the things that means is that they do not cause the run loop to return when they fire.

Warning:

The NSRunLoop class is generally not considered to be thread-safe and its methods should only be called within the context of the current thread. You should never try to call the methods of an NSRunLoop object running in a different thread, as doing so might cause unexpected results.

Blog Entries

- [MBS Xojo Plugins, version 21.1pr5](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr1](#)

8.8.2 Methods

8.8.3 AddDummyPort

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Adds a dummy port as event source.

Notes: Run loops don't loop unless there is an event source, so you can add a dummy one here.

8.8.4 allModes as string()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns array with all mode strings.

8.8.5 Constructor

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Creates a new object using current run loop.

8.8.6 `currentRunLoop` as `NSRunLoopMBS`

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Returns the `NSRunLoop` object for the current thread.

Notes: If a run loop does not yet exist for the thread, one is created and returned.

8.8.7 `mainRunLoop` as `NSRunLoopMBS`

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Returns the run loop of the main thread.

Notes: An object representing the main thread's run loop.

Available in OS X v10.5.

8.8.8 `NSDefaultRunLoopMode` as string

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: One of the runloop mode.

Notes: The mode to deal with input sources other than `NSConnection` objects.

This is the most commonly used run-loop mode.

8.8.9 `NSRunLoopCommonModes` as string

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: One of the runloop mode.

Notes: Objects added to a run loop using this value as the mode are monitored by all run loop modes that have been declared as a member of the set of "common" modes; see the description of `CFRunLoopAddCommonMode` for details.

Available in OS X v10.5 and later.

8.8.10 run

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Puts the receiver into a permanent loop, during which time it processes data from all attached input sources.

Notes: If no input sources or timers are attached to the run loop, this method exits immediately; otherwise, it runs the receiver in the `NSDefaultRunLoopMode` by repeatedly invoking `runMode:beforeDate:`. In other words, this method effectively begins an infinite loop that processes data from the run loop's input sources and timers.

Manually removing all known input sources and timers from the run loop is not a guarantee that the run loop will exit. OS X can install and remove additional input sources as needed to process requests targeted at the receiver's thread. Those sources could therefore prevent the run loop from exiting.

If you want the run loop to terminate, you shouldn't use this method. Instead, use one of the other run methods and also check other arbitrary conditions of your own, in a loop.

See also:

- 8.8.11 `run(Seconds as Double)` 372

8.8.11 run(Seconds as Double)

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Runs the runloop for the given number of seconds.

See also:

- 8.8.10 `run` 372

8.8.12 runMode(Mode as string, Seconds as Double) as boolean

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Runs the runloop for the given number of seconds in the given mode.

8.8.13 runModeUntilDate(Mode as string, limitDate as date) as boolean

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

Function: Runs the loop once, blocking for input in the specified mode until a given date.

Notes: mode: The mode in which to run. You may specify custom modes or use one of the modes listed in "Run Loop Modes."

limitDate: The date until which to block.

Returns true if the run loop ran and processed an input source or if the specified timeout value was reached; otherwise, false if the run loop could not be started.

If no input sources or timers are attached to the run loop, this method exits immediately and returns false; otherwise, it returns after either the first input source is processed or limitDate is reached. Manually removing all known input sources and timers from the run loop does not guarantee that the run loop will exit immediately. OS X may install and remove additional input sources as needed to process requests targeted at the receiver's thread. Those sources could therefore prevent the run loop from exiting.

Note: A timer is not considered an input source and may fire multiple times while waiting for this method to return

See also:

- 8.8.14 runModeUntilDate(Mode as string, limitDate as dateTime) as boolean

373

8.8.14 runModeUntilDate(Mode as string, limitDate as dateTime) as boolean

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

Function: Runs the loop once, blocking for input in the specified mode until a given date.

Notes: mode: The mode in which to run. You may specify custom modes or use one of the modes listed in "Run Loop Modes."

limitDate: The date until which to block.

Returns true if the run loop ran and processed an input source or if the specified timeout value was reached; otherwise, false if the run loop could not be started.

If no input sources or timers are attached to the run loop, this method exits immediately and returns false; otherwise, it returns after either the first input source is processed or limitDate is reached. Manually removing all known input sources and timers from the run loop does not guarantee that the run loop will exit immediately. OS X may install and remove additional input sources as needed to process requests targeted at the receiver's thread. Those sources could therefore prevent the run loop from exiting.

Note: A timer is not considered an input source and may fire multiple times while waiting for this method to return

See also:

- 8.8.13 runModeUntilDate(Mode as string, limitDate as date) as boolean

372

8.8.15 runOnce

Plugin Version: 21.1, Platform: macOS, Targets: All.

Function: Run the runloop once.

Notes: Great to process events now in a timer in web or console projects.

8.8.16 runUntilDate(limitDate as date)

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

Function: Runs the loop until the specified date, during which time it processes data from all attached input sources.

Notes: limitDate: The date up until which to run.

If no input sources or timers are attached to the run loop, this method exits immediately; otherwise, it runs the receiver in the NSDefaultRunLoopMode by repeatedly invoking runMode until the specified expiration date.

Manually removing all known input sources and timers from the run loop is not a guarantee that the run loop will exit. OS X can install and remove additional input sources as needed to process requests targeted at the receiver's thread. Those sources could therefore prevent the run loop from exiting.

See also:

- 8.8.17 runUntilDate(limitDate as dateTime) 374

8.8.17 runUntilDate(limitDate as dateTime)

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

Function: Runs the loop until the specified date, during which time it processes data from all attached input sources.

Notes: limitDate: The date up until which to run.

If no input sources or timers are attached to the run loop, this method exits immediately; otherwise, it runs the receiver in the NSDefaultRunLoopMode by repeatedly invoking runMode until the specified expiration date.

Manually removing all known input sources and timers from the run loop is not a guarantee that the run loop will exit. OS X can install and remove additional input sources as needed to process requests targeted at the receiver's thread. Those sources could therefore prevent the run loop from exiting.

See also:

- 8.8.16 runUntilDate(limitDate as date) 374

8.8.18 Properties

8.8.19 currentMode as String

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Returns the receiver's current input mode.

Notes: The receiver's current input mode. This method returns the current input mode only while the receiver is running; otherwise, it returns nil.

(Read only property)

8.8.20 Handle as Integer

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The internal object handle.

Notes: (Read and Write property)

8.9 class NSSavePanelObserverMBS

8.9.1 class NSSavePanelObserverMBS

Plugin Version: 20.4, Platform: macOS, Targets: Desktop only.

Function: Intercepts save and open dialogs on macOS to provide events to customize.

Notes: You can only have one instance of this class at a time.

First time you create an object, we initialize the necessary feature.

See also WinFileDialogObserverMBS class for Windows.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 20.4](#)
- [MBS Xojo Plugins, version 20.4pr7](#)

Xojo Developer Magazine

- [18.6, page 10: News](#)

8.9.2 Properties

8.9.3 Enabled as Boolean

Plugin Version: 20.4, Platform: macOS, Targets: Desktop only.

Function: Whether the feature is enabled.

Notes: If value is false, no events will be called.

(Read and Write property)

8.9.4 Events

8.9.5 DidShowPanel(panel as NSSavePanelMBS, Result as Integer)

Plugin Version: 20.4, Platform: macOS, Targets: .

Function: Event called after panel showed.

Notes: You can read results from whatever modification you made.

8.9.6 WillShowPanel(panel as NSSavePanelMBS)

Plugin Version: 20.4, Platform: macOS, Targets: .

Function: Event called before panel shows.

Notes: You can do customization just before the dialog shows.

8.10 class NSServiceProviderMBS

8.10.1 class NSServiceProviderMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The class to do service handlers in Cocoa applications.

Notes: This class must be subclassed to implement services.

Use ServiceInvoked as the name of the selector when declaring service (NSMessage parameter).

Use NSUserData parameter to distinguish between all the services you offer.

This class implements NSServiceProvider for Xojo and Xojo for Cocoa applications. For Carbon, please use CarbonApplicationEventsMBS events.

see also:

<https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/SysServices/introduction.html>

Blog Entries

- [MBS Xojo Plugins, version 20.6pr2](#)
- [MBS Xojo / Real Studio plug-ins in version 14.2](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr4](#)

8.10.2 Methods

8.10.3 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The constructor.

8.10.4 Destructor

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The destructor.

8.10.5 Properties

8.10.6 Handle as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Internal object reference.

Notes: (Read and Write property)

8.10.7 Events

8.10.8 readSelectionFromPasteboard(pboard as NSPasteboardMBS) as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: .

Function: Reads data from the pasteboard.

Notes: pboard: The pasteboard containing the data to read.

Return true if your implementation was able to read the pasteboard data successfully; otherwise, false.

8.10.9 ServiceInvoked(pboard as NSPasteboardMBS, userData as string, byref error as string)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: This event is called when a service should perform.

Notes: Use ServiceInvoked as the name of the selector when declaring service (NSMessage parameter).

Use NSUserData parameter to distinguish between all the services you offer.

8.10.10 validRequestor(sendType as String, returnType as String) as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: .

Function: Indicates whether the receiver can send and receive the specified pasteboard types.

Notes: sendType: The pasteboard type the app needs to send.

returnType: The pasteboard type the app needs to receive.

Return true if your class can send and receive the specified types or false if the receiver knows of no object that can send and receive data of that type.

8.10.11 writeSelectionToPasteboard(pboard as NSPasteboardMBS, types() as String) as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: .

Function: Write the current selection to the specified pasteboard under each given type.

Notes: pboard: The pasteboard to write to.

types: An array of strings describing the types of data to write.

Return true if the data for any single type was successfully written, false otherwise.

8.11 class NSTimerMBS

8.11.1 class NSTimerMBS

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: The cocoa timer class.

Example:

```
dim n as new NSTimerMBS(5, true)
```

Notes: You use the NSTimer class to create timer objects or, more simply, timers. A timer waits until a certain time interval has elapsed and then fires, calling the action event. For example, you could create an NSTimer object that calls the action event, telling it to update itself after a certain time interval.

A timer is not a real-time mechanism; it fires only when one of the run loop modes to which the timer has been added is running and able to check if the timer's firing time has passed. Because of the various input sources a typical run loop manages, the effective resolution of the time interval for a timer is limited to on the order of 50-100 milliseconds. If a timer's firing time occurs during a long callout or while the run loop is in a mode that is not monitoring the timer, the timer does not fire until the next time the run loop checks the timer. Therefore, the actual time at which the timer fires potentially can be a significant period of time after the scheduled firing time.

Repeating Versus Non-Repeating Timers

You specify whether a timer is repeating or non-repeating at creation time. A non-repeating timer fires once and then invalidates itself automatically, thereby preventing the timer from firing again. By contrast, a repeating timer fires and then reschedules itself on the same run loop.

A repeating timer always schedules itself based on the scheduled firing time, as opposed to the actual firing time. For example, if a timer is scheduled to fire at a particular time and every 5 seconds after that, the scheduled firing time will always fall on the original 5 second time intervals, even if the actual firing time gets delayed. If the firing time is delayed so far that it passes one or more of the scheduled firing times, the timer is fired only once for that time period; the timer is then rescheduled, after firing, for the next scheduled firing time in the future.

Scheduling Timers in Run Loops

A timer object can be registered in only one run loop at a time, although it can be added to multiple run loop modes within that run loop. There are three ways to create a timer:

Use the Constructor to create the timer and schedule it on the current run loop in the default mode.

Once scheduled, the timer fires at the specified interval until it is invalidated. A non-repeating timer invalidates itself immediately after it fires. However, for a repeating timer, you must invalidate the timer

object yourself by calling its `invalidate` method. Calling this method requests the removal of the timer from the current run loop; as a result, you should always call the `invalidate` method from the same thread on which the timer was installed. Invalidating the timer immediately disables it so that it no longer affects the run loop. The run loop then removes and releases the timer, either just before the `invalidate` method returns or at some later point. Once invalidated, timer objects cannot be reused.

Blog Entries

- [MBS Xojo / Real Studio plug-ins in version 13.5](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr1](#)
- [MBS REALbasic plug-ins version 10.0](#)

Xojo Developer Magazine

- [12.1, page 9: News](#)

8.11.2 Methods

8.11.3 Constructor(`fireDate` as `date`, `timeInterval` as `Double`, `repeats` as `boolean`)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a new `NSTimer`.

Example:

```
dim d as new date
d.hour = d.hour + 1 // start in one hour
```

```
dim n as new NSTimerMBS(d, 5, true)
```

Notes: `fireDate`: The time at which the timer should first fire.

`timeInterval`: For a repeating timer, this parameter contains the number of seconds between firings of the timer. If seconds is less than or equal to 0.0, this method chooses the nonnegative value of 0.1 milliseconds instead.

`repeats`: If true, the timer will repeatedly reschedule itself until invalidated. If false, the timer will be invalidated after it fires.

The time is initialized such that, when added to a run loop, it will fire at `date` and then, if `repeats` is true, every seconds after that.

See also:

- [8.11.4 Constructor\(`fireDate` as `date`, `timeInterval` as `Double`, `repeats` as `boolean`, `runloop` as `NSRunLoopMBS`, `runloopMode` as `string`\)](#) 383

8.11. CLASS NSTIMERMBS	383
• 8.11.5 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean)	383
• 8.11.6 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string)	384
• 8.11.7 Constructor(timeInterval as Double, repeats as boolean)	385

8.11.4 Constructor(fireDate as date, timeInterval as Double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string)

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

Function: Initializes a new NSTimer.

Notes: fireDate: The time at which the timer should first fire.

timeInterval: For a repeating timer, this parameter contains the number of seconds between firings of the timer. If seconds is less than or equal to 0.0, this method chooses the nonnegative value of 0.1 milliseconds instead.

repeats: If true, the timer will repeatedly reschedule itself until invalidated. If false, the timer will be invalidated after it fires.

The time is initialized such that, when added to a run loop, it will fire at date and then, if repeats is true, every seconds after that.

Schedules the timer to run on the given runloop in the given mode.

See also:

• 8.11.3 Constructor(fireDate as date, timeInterval as Double, repeats as boolean)	382
• 8.11.5 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean)	383
• 8.11.6 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string)	384
• 8.11.7 Constructor(timeInterval as Double, repeats as boolean)	385

8.11.5 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean)

Plugin Version: 20.5, Platforms: macOS, iOS, Targets: All.

Function: Creates and returns a new NSTimer object and schedules it on the current run loop in the default mode.

Notes: timeInterval: The number of seconds between firings of the timer. If seconds is less than or equal to 0.0, this method chooses the nonnegative value of 0.1 milliseconds instead.

repeats: If true, the timer will repeatedly reschedule itself until invalidated. If false, the timer will be invalidated after it fires.

After seconds seconds have elapsed, the timer fires, calling the Action event.

See also:

- 8.11.3 Constructor(fireDate as date, timeInterval as Double, repeats as boolean) 382
- 8.11.4 Constructor(fireDate as date, timeInterval as Double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string) 383
- 8.11.6 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string) 384
- 8.11.7 Constructor(timeInterval as Double, repeats as boolean) 385

8.11.6 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string)

Plugin Version: 20.5, Platforms: macOS, iOS, Targets: All.

Function: Initializes a new NSTimer.

Notes: fireDate: The time at which the timer should first fire.

timeInterval: For a repeating timer, this parameter contains the number of seconds between firings of the timer. If seconds is less than or equal to 0.0, this method chooses the nonnegative value of 0.1 milliseconds instead.

repeats: If true, the timer will repeatedly reschedule itself until invalidated. If false, the timer will be invalidated after it fires.

The time is initialized such that, when added to a run loop, it will fire at date and then, if repeats is true, every seconds after that.

Schedules the timer to run on the given runloop in the given mode.

See also:

- 8.11.3 Constructor(fireDate as date, timeInterval as Double, repeats as boolean) 382
- 8.11.4 Constructor(fireDate as date, timeInterval as Double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string) 383
- 8.11.5 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean) 383
- 8.11.7 Constructor(timeInterval as Double, repeats as boolean) 385

8.11.7 Constructor(timeInterval as Double, repeats as boolean)

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: Creates and returns a new NSTimer object and schedules it on the current run loop in the default mode.

Example:

```
dim n as new NSTimerMBS(5, true)
MsgBox str(n.timeInterval)
```

Notes: timeInterval: The number of seconds between firings of the timer. If seconds is less than or equal to 0.0, this method chooses the nonnegative value of 0.1 milliseconds instead.

repeats: If true, the timer will repeatedly reschedule itself until invalidated. If false, the timer will be invalidated after it fires.

After seconds seconds have elapsed, the timer fires, calling the Action event.

See also:

- 8.11.3 Constructor(fireDate as date, timeInterval as Double, repeats as boolean) 382
- 8.11.4 Constructor(fireDate as date, timeInterval as Double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string) 383
- 8.11.5 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean) 383
- 8.11.6 Constructor(fireDate as dateTime, timeInterval as double, repeats as boolean, runloop as NSRunLoopMBS, runloopMode as string) 384

8.11.8 fire

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: Causes the receiver's message to be sent to its target.

Notes: You can use this method to fire a repeating timer without interrupting its regular firing schedule. If the timer is non-repeating, it is automatically invalidated after firing, even if its scheduled fire date has not arrived.

8.11.9 invalidate

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: Stops the receiver from ever firing again and requests its removal from its run loop.

Example:

```
dim n as new NSTimerMBS(5, true)

// later

n.invalidate
```

Notes: The destructor calls `invalidate` automatically for you.

This method is the only way to remove a timer from an `NSRunLoop` object. The `NSRunLoop` object removes and releases the timer, either just before the `invalidate` method returns or at some later point.

You must send this message from the thread on which the timer was installed. If you send this message from another thread, the input source associated with the timer may not be removed from its run loop, which could prevent the thread from exiting properly.

8.11.10 Timer(*t as timer*) as NSTimerMBS

Plugin Version: 14.2, Platforms: macOS, iOS, Targets: All.

Function: Creates a `NSTimerMBS` object referencing the given timer object.

Example:

```
dim n as NSTimerMBS = NSTimerMBS.timer(timer1)
n.tolerance = 0.1
```

Notes: Only for Cocoa 32bit currently.
Works fine in Xojo 2012 and Xojo 2014r1 in Cocoa target.

8.11.11 Properties

8.11.12 fireDate as date

Plugin Version: 10.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date at which the receiver will fire.

Example:

```
dim d as new date
d.hour = d.hour + 1 // start in one hour

dim n as new NSTimerMBS(d, 5, true)
```

```
MsgBox n.fireDate.ShortDate+" "+n.fireDate.ShortTime
```

Notes: The date at which the receiver will fire. If the timer is no longer valid, this method returns the last date at which the timer fired.

Use the `isValid` method to verify that the timer is valid.

You typically use this method to adjust the firing time of a repeating timer. Although resetting a timer's next firing time is a relatively expensive operation, it may be more efficient in some situations. For example, you could use it in situations where you want to repeat an action multiple times in the future, but at irregular time intervals. Adjusting the firing time of a single timer would likely incur less expense than creating multiple timer objects, scheduling each one on a run loop, and then destroying them.

You should not call this method on a timer that has been invalidated, which includes non-repeating timers that have already fired. You could potentially call this method on a non-repeating timer that had not yet fired, although you should always do so from the thread to which the timer is attached to avoid potential race conditions.

(Read and Write property)

8.11.13 fireDateTime as DateTime

Plugin Version: 20.5, Platforms: macOS, iOS, Targets: All.

Function: The date at which the receiver will fire.

Notes: (Read and Write property)

8.11.14 Handle as Integer

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: The internal reference to the NSTimer object.

Example:

```
dim n as new NSTimerMBS(5, true)
MsgBox str(n.Handle)
```

Notes: (Read and Write property)

8.11.15 isValid as boolean

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver is currently valid.

Example:

```
dim n as new NSTimerMBS(5, true)
MsgBox str(n.isValid)
```

Notes: True if the receiver is still capable of firing or false if the timer has been invalidated and is no longer capable of firing.

(Read only property)

8.11.16 tag as Variant

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: A value you can use in your app however you like.

Example:

```
dim n as new NSTimerMBS(5, true)
n.tag = window1 // some value you may later use
```

Notes: (Read and Write property)

8.11.17 timeInterval as Double

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: All.

Function: The receiver's time interval.

Example:

```
dim n as new NSTimerMBS(5, true)
MsgBox str(n.timeInterval)
```

Notes: The receiver's time interval. If the receiver is a non-repeating timer, returns 0 (even if a time interval was set).

(Read only property)

8.11.18 tolerance as Double

Plugin Version: 13.5, Platforms: macOS, iOS, Targets: All.

Function: The tolerance for this timer.

Notes: Requires Mac OS X 10.9.

Setting a tolerance for a timer allows it to fire later than the scheduled fire date, improving the ability of the system to optimize for increased power savings and responsiveness. The timer may fire at any time between its scheduled fire date and the scheduled fire date plus the tolerance. The timer will not fire before the scheduled fire date. For repeating timers, the next fire date is calculated from the original fire date regardless of tolerance applied at individual fire times, to avoid drift. The default value is zero, which means no additional tolerance is applied. The system reserves the right to apply a small amount of tolerance to certain timers regardless of the value of this property.

As the user of the timer, you will have the best idea of what an appropriate tolerance for a timer may be. A general rule of thumb, though, is to set the tolerance to at least 10% of the interval, for a repeating timer. Even a small amount of tolerance will have a significant positive impact on the power usage of your application. The system may put a maximum value of the tolerance.
(Read and Write property)

8.11.19 Events

8.11.20 Action

Plugin Version: 10.0, Platforms: macOS, iOS, Targets: .

Function: The event called when the timer fires.

8.12 class `NSWindowDelegateMBS`

8.12.1 class `NSWindowDelegateMBS`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: A Xojo class for a cocoa windows delegate.

Notes: The `NSWindowDelegate` protocol defines the methods that a delegate of `NSWindow` should implement. All methods in this protocol are optional.

By implementing these methods, the delegate may respond to window resizing, moving, exposing, minimizing, and a number of other window events.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 16.5pr3](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr1](#)
- [MBS Real Studio Plugins, version 12.5pr3](#)
- [Lion features for Real Studio](#)
- [MBS Real Studio Plugins, version 11.2pr11](#)

8.12.2 Methods

8.12.3 Constructor(win as DesktopWindow)

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

Function: Initializes the delegate class pointing to the Xojo window.

Notes: This class does not keep a reference to the window, so you can keep this delegate as a property of the window without a memory leak.

The original delegate from Xojo is preserved and all messages are forwarded to it. Also when this object is destroyed, the old delegate is restored.

See also:

- 8.12.4 Constructor(win as `NSWindowMBS`) 390
- 8.12.5 Constructor(win as window) 391

8.12.4 Constructor(win as `NSWindowMBS`)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

8.12. CLASS NSWINDOWDELEGATEMBS 391

Function: Initializes the delegate class pointing to the Cocoa window.

Notes: The original delegate on the window is preserved and all messages are forwarded to it. Also when this object is destroyed, the old delegate is restored.

See also:

- 8.12.3 Constructor(win as DesktopWindow) 390
- 8.12.5 Constructor(win as window) 391

8.12.5 Constructor(win as window)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Initializes the delegate class pointing to the Xojo window.

Notes: This class does not keep a reference to the window, so you can keep this delegate as a property of the window without a memory leak.

The original delegate from Xojo is preserved and all messages are forwarded to it. Also when this object is destroyed, the old delegate is restored.

See also:

- 8.12.3 Constructor(win as DesktopWindow) 390
- 8.12.4 Constructor(win as NSWindowMBS) 390

8.12.6 InstallRestoreEvents

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Activates plugin code to catch events `restoreStateWithCoder` and `encodeRestorableStateWithCoder`.

8.12.7 Events

8.12.8 concludeDragOperation(sender as NSDraggingInfoMBS)

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

Function: Invoked when the dragging operation is complete, signaling the receiver to perform any necessary clean-up.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

For this method to be invoked, the previous `performDragOperation` must have returned true.

The destination implements this method to perform any tidying up that it needs to do, such as updating its visual representation now that it has incorporated the dragged data. This message is the last message sent from sender to the destination during a dragging session.

If the sender object's `animatesToDestination` property was set to true in `prepareForDragOperation`, then the drag image is still visible. At this point you should draw the final visual representation in the view. When this method returns, the drag image is removed from the screen. If your final visual representation matches the visual representation in the drag, this is a seamless transition.

8.12.9 `customWindowsToEnterFullScreenForWindow(win as NSWindowMBS) as NSWindowMBS()`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked when the window is about to enter full screen mode.

Notes: win: The window to enter to full screen mode.

Return an array of windows involved in the animation to fullscreen for window; otherwise nil.

This method lets a window delegate to customize the animation by providing a custom window or windows containing layers or other effects. If you do not want to perform custom animation, you can omit the implementation of this method, or it can return nil.

Available in Mac OS X v10.7 and later.

8.12.10 `customWindowsToExitFullScreenForWindow(win as NSWindowMBS) as NSWindowMBS()`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: The system has started its animation out of fullscreen, including transitioning back to the desktop space.

Notes: win: The window to exit fullscreen.

Return true if the window wants to run a custom animation; otherwise false if the default NSWindow animation should be used.

This method lets the window delegate customize the animation when the window is about to exit fullscreen. If an you do not want to perform custom animation, you can omit the implementation of this method, or it can return nil.

Available in Mac OS X v10.7 and later.

8.12.11 didDecodeRestorableState(win as NSWindowMBS, state as NSCoderMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate the window is has extracted its restorable state from a given archiver.

Notes: win: The window extracting its restorable state from an archive.

state: The coder extracting the archive.

This method is invoked during the window's restoreStateWithCoder method.

Available in Mac OS X v10.7 and later.

8.12.12 draggingEnded(sender as NSDraggingInfoMBS)

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

Function: Implement this event to be notified when a drag operation ends in some other destination.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

This method might be used by a destination doing auto-expansion in order to collapse any auto-expands.

8.12.13 draggingEntered(sender as NSDraggingInfoMBS) as Integer

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

Function: Invoked when the dragged image enters destination bounds or frame; delegate returns dragging operation to perform.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered message.

Invoked when a dragged image enters the destination but only if the destination has registered for the paste-board data type involved in the drag operation. Specifically, this method is invoked when the mouse pointer enters the destination's bounds rectangle (if it is a view object) or its frame rectangle (if it is a window object).

This method must return a value that indicates which dragging operation the destination will perform when the image is released. In deciding which dragging operation to return, the method should evaluate the overlap between both the dragging operations allowed by the source (obtained from sender with the `draggingSourceOperationMask` method) and the dragging operations and pasteboard data types the destination itself supports.

If none of the operations is appropriate, this method should return `NSDragOperationNone` (this is the default response if the method is not implemented by the destination). A destination will still receive `draggingUpdated` and `draggingExited` even if `NSDragOperationNone` is returned by this method.

8.12.14 `draggingExited(sender as NSDraggingInfoMBS)`

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

Function: Invoked when the dragged image exits the destination's bounds rectangle (in the case of a view object) or its frame rectangle (in the case of a window object).

Notes: sender: The object sending the message; use it to get details about the dragging operation.

8.12.15 `draggingUpdated(sender as NSDraggingInfoMBS) as Integer`

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

Function: Invoked periodically as the image is held within the destination area, allowing modification of the dragging operation or mouse-pointer position.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in `NSDragOperation` in the `NSDraggingInfo` reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous `draggingEntered` message.

For this to be invoked, the destination must have registered for the pasteboard data type involved in the drag operation. The messages continue until the image is either released or dragged out of the window or view.

This method provides the destination with an opportunity to modify the dragging operation depending on the position of the mouse pointer inside of the destination view or window object. For example, you may have several graphics or areas of text contained within the same view and wish to tailor the dragging operation, or to ignore the drag event completely, depending upon which object is underneath the mouse pointer at the time when the user releases the dragged image and the `performDragOperation` method is invoked.

You typically examine the contents of the pasteboard in the `draggingEntered` method, where this examination is performed only once, rather than in the `draggingUpdated` method, which is invoked multiple times.

Only one destination at a time receives a sequence of draggingUpdated messages. If the mouse pointer is within the bounds of two overlapping views that are both valid destinations, the uppermost view receives these messages until the image is either released or dragged out.

8.12.16 encodeRestorableStateWithCoder(win as NSWindowMBS, coder as NSCoderMBS)

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

Function: Method called to save the restorable state.

Notes: The receiver is passed an NSCoderMBS that supports keyed encoding (but not decoding), and should encode its restorable state. If you override this method, you should call through to super. You should not otherwise invoke this method. If you encode an object that implements the NSUserInterfaceItemIdentification protocol, the object itself is not archived; only its identifier is stored. Thus, for example, a window may efficiently store its firstResponder as restorable state.

Called only if you called InstallRestoreEvents at least ones.

8.12.17 performDragOperation(sender as NSDraggingInfoMBS) as boolean

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

Function: Invoked after the released image has been removed from the screen, signaling the receiver to import the pasteboard data.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return if the destination accepts the data, it returns true; otherwise it returns false. The default is to return false.

For this method to be invoked, the previous prepareForDragOperation message must have returned true. The destination should implement this method to do the real work of importing the pasteboard data represented by the image.

If the sender object's animatesToDestination was set to true in prepareForDragOperation, then setup any animation to arrange space for the drag items to animate to. Also at this time, enumerate through the dragging items to set their destination frames and destination images.

8.12.18 prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean

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

Function: Invoked when the image is released, allowing the receiver to agree to or refuse drag operation.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns true if the receiver agrees to perform the drag operation and false if not.

This method is invoked only if the most recent draggingEntered or draggingUpdated message returned an acceptable drag-operation value.

If you want the drag items to animate from their current location on screen to their final location in your view, set the sender object's animatesToDestination property to true in your implementation of this method.

8.12.19 restoreStateWithCoder(win as NSWindowMBS, coder as NSCoderMBS)

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

Function: Method called to restore state.

Notes: The receiver is passed an NSCoder that supports keyed decoding (but not encoding). The receiver should decode any previously stored state. If you override this method, you should call through to super. You should not otherwise invoke this method.

Called only if you called InstallRestoreEvents at least ones.

8.12.20 shouldDragDocumentWithEvent(win as NSWindowMBS, evnt as NSEventMBS, dragImageLocation as NSPointMBS, pasteboard as Variant) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Asks the delegate whether a user can drag the document icon from the window's title bar.

Notes: win: The window containing the document icon the user wants to drag.

evnt: The left-mouse down event that triggered the dragging operation.

dragImageLocation: The location at which the user started the dragging operation.

pasteboard: The pasteboard containing the contents of the document, which the delegate can modify. This is a NSPasteboardMBS object.

Return true to allow the drag to proceed; false to prevent it. Before turning no the delegate can implement its own dragging behavior as described below.

Implementing this method allows an application to customize the process of dragging the window's document icon. implement its own dragging process, the delegate can perform the dragging operation and return false.

The delegate can prohibit the drag by returning false. Before returning false, the delegate may implement its own dragging behavior.

Available in Mac OS X v10.5 and later.

8.12.21 shouldPopUpDocumentPathMenu(win as NSWindowMBS, menu as NSMenuMBS) as boolean

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Asks the delegate whether the window displays the title pop-up menu in response to a Command-click or Control-click on its title.

Notes: win: The window whose title the user Command-clicked or Control-clicked.

menu: The menu the window will display, if allowed. By default, its items are the path components of the file represented by window.

Returns true to allow the display of the title pop-up menu; false to prevent it.

Available in Mac OS X v10.5 and later.

8.12.22 startCustomAnimationToEnterFullscreenWithDuration(win as NSWindowMBS, duration as Double)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked when the window is about to enter fullscreen mode.

Notes: win: The window to enter to full screen mode.

duration: The duration of the presentation change.

This method is called to start the window animation into fullscreen, including transitioning to a new space. You can implement this method to perform custom animation with the given duration to be in sync with the system animation.

Special Considerations

This method is called only if customWindowsToEnterFullscreenForWindow returned non-nil.

Available in Mac OS X v10.7 and later.

8.12.23 startCustomAnimationToExitFullscreenWithDuration(win as NSWindowMBS, duration as Double)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that the window is about to enter fullscreen mode.

Notes: win: The window to exit to fullscreen.

duration: The duration of the presentation change.

Special Considerations

This method is called only if `customWindowsToExitFullScreenForWindow` returned `non-nil`.

Available in Mac OS X v10.7 and later.

8.12.24 `updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)`

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

Function: Invoked when the dragging images should be changed.

Notes: sender: The object sending the message; use this object to get details about the dragging operation.

While a destination may change the dragging images at any time, it is recommended to wait until this method is called before updating the dragging images.

This allows the system to delay changing the dragging images until it is likely that the user will drop on this destination. Otherwise, the dragging images will change too often during the drag which would be distracting to the user.

8.12.25 `wantsPeriodicDraggingUpdates` as boolean

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

Function: Asks the destination object whether it wants to receive periodic `draggingUpdated` messages.

Notes: Return true if the destination wants to receive periodic `draggingUpdated` messages, false otherwise.

If the destination returns false, these messages are sent only when the mouse moves or a modifier flag changes. Otherwise the destination gets the default behavior, where it receives periodic `draggingUpdated` messages even if nothing changes.

8.12.26 `willEncodeRestorableState(win as NSWindowMBS, state as NSCoderMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate the window is about to add its restorable state to a given archiver.

Notes: win: The window adding its restorable state to an archive.

state: The coder creating the archive.

This method is invoked during the window's `encodeRestorableStateWithCoder` method.

Available in Mac OS X v10.7 and later.

8.12.27 `willPositionSheet(win as NSWindowMBS, sheet as NSWindowMBS, rect as NSRectMBS) as NSRectMBS`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window is about to show a sheet at the specified location, giving it the opportunity to return a custom location for the attachment of the sheet to the window.

Notes: `win`: The window containing the sheet to be animated.

`sheet`: The sheet to be shown.

`rect`: The default sheet location, just under the title bar of the window, aligned with the left and right edges of the window.

Return the custom location specified.

This method is also invoked whenever the user resizes window while sheet is attached.

This method is useful in many situations. If your window has a toolbar, for example, you can specify a location for the sheet that is just below it. If you want the sheet associated with a certain control or view, you could position the sheet so that it appears to originate from the object (through animation) or is positioned next to it.

Neither the `rect` parameter nor the returned `NSRect` value define the boundary of the sheet. They indicate where the top-left edge of the sheet is attached to the window. The origin is expressed in window coordinates; the default `origin.y` value is the height of the content view and the default `origin.x` value is 0. The `size.width` value indicates the width and behavior of the initial animation; if `size.width` is narrower than the sheet, the sheet genes out from the specified location, and if `size.width` is wider than the sheet, the sheet slides out. You cannot affect the size of the sheet through the `size.width` and `size.height` fields. It is recommended that you specify zero for the `size.height` value as this field may have additional meaning in a future release.

Available in Mac OS X v10.3 and later.

8.12.28 `willResizeForVersionBrowser(win as NSWindowMBS, maxPreferredFrameSize as NSSizeMBS, maxAllowedFrameSize as NSSizeMBS) as NSSizeMBS`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate the the window will resize for presentation during version browsing.

Notes: win: The window being presented in a version browser.

maxPreferredSize: The maximum size the version browser would prefer the window to be.

maxAllowedSize: The maximum allowed size for the window (the full screen frame minus the margins required to ensure the Versions controls are still visible).

Returns the size that the window should be.

Windows entering the version browser will be resized to the size returned by this method. If either dimension of the returned size is larger than the maxPreferredFrameSize, the window will also be scaled down to ensure it fits properly in the version browser.

If this method is not implemented, the version browser will use windowWillUseStandardFrame to determine the resulting window frame size.

Available in Mac OS X v10.7 and later.

8.12.29 willUseFullScreenContentSize(win as NSWindowMBS, proposedSize as NSSizeMBS) as NSSizeMBS

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked to allow the delegate to modify the fullscreen content size.

Notes: win: The window to enter to full screen mode.

proposedSize: The proposed window size.

Returns the window size to actually use when displaying content size.

Available in Mac OS X v10.7 and later.

8.12.30 willUseFullScreenPresentationOptions(win as NSWindowMBS, proposedOptions as Integer) as Integer

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Returns the presentation options the window will use when transitioning to fullscreen.

Notes: win: The window to enter to full screen mode.

proposedOptions: The proposed options. See NSApplicationPresentationOptions for the possible values.

Return the options the window should use when transitioning to fullscreen. These may be the same as the

proposedOptions or may be modified.

Available in Mac OS X v10.7 and later.

8.12.31 windowDidBecomeKey(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that the window has become the key window.

Notes: notification: A notification named NSWindowDidBecomeKeyNotification.

You can retrieve the window object in question by sending object to notification.

Available in Mac OS X v10.0 and later.

8.12.32 windowDidBecomeMain(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that the window has become main.

Notes: notification: A notification named NSWindowDidBecomeMainNotification.

You can retrieve the window object in question by sending object to notification.

Available in Mac OS X v10.0 and later.

8.12.33 windowDidChangeScreen(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window has changed screens.

Notes: notification: A notification named NSWindowDidChangeScreenNotification.

You can retrieve the NSWindow object in question by sending object to notification.

Available in Mac OS X v10.0 and later.

8.12.34 `windowDidChangeScreenProfile(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window has changed screen display profiles.

Notes: notification: A notification named `NSNotificationMBS.windowDidChangeScreenProfileNotification`.

You can retrieve the `NSNotificationMBS` object in question by sending `notification` to `notification`.

Available in Mac OS X v10.4 and later.

8.12.35 `windowDidDeminiaturize(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window has been deminimized.

Notes: notification: A notification named `NSNotificationMBS.windowDidDeminiaturizeNotification`.

You can retrieve the `NSNotificationMBS` object in question by sending `notification` to `notification`.

Available in Mac OS X v10.0 and later.

8.12.36 `windowDidEndLiveResize(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that a live resize operation on the window has ended.

Notes: notification: A notification named `NSNotificationMBS.windowDidEndLiveResizeNotification`.

You can retrieve the window object in question by sending `notification` to `notification`.

Available in Mac OS X v10.6 and later.

8.12.37 `windowDidEndSheet(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window has closed a sheet.

Notes: notification: A notification named `NSNotificationMBS.windowDidEndSheetNotification`.

You can retrieve the window object in question by sending object to notification.
Available in Mac OS X v10.1 and later.

8.12.38 windowDidEnterFullScreen(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: The window just entered fullscreen mode.

Notes: notification: A notification named NSWindowDidEnterFullScreenNotification.
Available in Mac OS X v10.7 and later.

8.12.39 windowDidEnterVersionBrowser(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate the the window just entered version browsing.

Notes: notification: An NSWindowDidEnterVersionBrowserNotification notification.

Available in Mac OS X v10.7 and later.

8.12.40 windowDidExitFullScreen(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: The window is about to enter fullscreen mode.

Notes: notification: A notification named NSWindowDidExitFullScreenNotification.
Available in Mac OS X v10.7 and later.

8.12.41 windowDidExitVersionBrowser(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate the the window is about to leave version browsing.

Notes: notification: An NSWindowDidExitVersionBrowserNotification notification.
Available in Mac OS X v10.7 and later.

8.12.42 windowDidExpose(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window has been exposed.

Notes: notification: A notification named `NSNotification`. You can retrieve the window object in question by sending `object` to `notification`. Available in Mac OS X v10.0 and later.

8.12.43 `windowDidFailToEnterFullScreen(win as NSWindowMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked if the window failed to enter fullscreen.

Notes: win: The window that failed to enter to full screen mode.

In some cases, the transition to enter fullscreen will fail, due to being in the midst of handling some other animation or user gesture. This method indicates that there was an error, and you should clean up any work you may have done to prepare to enter fullscreen.

This message is sent whether or not the delegate indicated a custom animation by returning non-nil from `customWindowsToEnterFullScreenForWindow`.

Available in Mac OS X v10.7 and later.

8.12.44 `windowDidFailToExitFullScreen(win as NSWindowMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked if the window failed to exit fullscreen.

Notes: win: The window that failed to exit to fullscreen.

In some cases, the transition to exit fullscreen will fail, due to being in the midst of handling some other animation or user gesture. This method indicates that there was an error, and you should clean up any work you may have done to prepare to enter fullscreen.

This message is sent whether or not the delegate indicated a custom animation by returning non-nil from `customWindowsToExitFullScreenForWindow`.

Available in Mac OS X v10.7 and later.

8.12.45 windowDidMiniaturize(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window has been minimized.

Notes: notification: A notification named NSWindowDidMiniaturizeNotification.

You can retrieve the NSWindow object in question by sending object to notification.
Available in Mac OS X v10.0 and later.

8.12.46 windowDidMove(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window has moved.

Notes: notification: A notification named NSWindowDidMoveNotification.

You can retrieve the NSWindow object in question by sending object to notification.
Available in Mac OS X v10.0 and later.

8.12.47 windowDidResignKey(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that the window has resigned key window status.

Notes: notification: A notification named NSWindowDidResignKeyNotification.

You can retrieve the window object in question by sending object to notification.
Available in Mac OS X v10.0 and later.

8.12.48 windowDidResignMain(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that the window has resigned main window status.

Notes: notification: A notification named NSWindowDidResignMainNotification.

You can retrieve the window object in question by sending object to notification.
Available in Mac OS X v10.0 and later.

8.12.49 `windowDidResize(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that the window has been resized.

Notes: notification: A notification named `NSWindowDidResizeNotification`.

You can retrieve the window object in question by sending `object` to `notification`.
Available in Mac OS X v10.0 and later.

8.12.50 `windowDidUpdate(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window received an update message.

Notes: notification: A notification named `NSWindowDidUpdateNotification`

You can retrieve the window object in question by sending `object` to `notification`.
Available in Mac OS X v10.0 and later.

8.12.51 `windowShouldClose as boolean`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the user has attempted to close a window or the window has received a `performClose` message.

Notes: Return true to allow sender to be closed; otherwise, false.

This method may not always be called during window closing. Specifically, this method is not called when a user quits an application.

Available in Mac OS X v10.0 and later.

8.12.52 `windowShouldZoom(win as NSWindowMBS, newFrame as NSRectMBS) as boolean`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Asks the delegate whether the specified window should zoom to the specified frame.

Notes: win: The window being zoomed.

newFrame: The rectangle to which the specified window is being zoomed.

Return true to allow window's frame to become newFrame; otherwise, false.

Available in Mac OS X v10.0 and later.

8.12.53 windowWillBeginSheet(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Notifies the delegate that the window is about to open a sheet.

Notes: notification: A notification named NSWindowWillBeginSheetNotification.

You can retrieve the window object in question by sending object to notification.

Available in Mac OS X v10.1 and later.

8.12.54 windowWillClose(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window is about to close.

Notes: notification: A notification named NSWindowWillCloseNotification.

You can retrieve the NSWindow object in question by sending object to notification.

Available in Mac OS X v10.0 and later.

8.12.55 windowWillEnterFullScreen(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: The window is about to enter fullscreen mode.

Notes: notification: A notification named NSWindowWillEnterFullScreenNotification.

Available in Mac OS X v10.7 and later.

8.12.56 windowWillEnterVersionBrowser(notification as NSNotificationMBS)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate the the window is about to enter version browsing.

Notes: notification: An `NSWindowWillEnterVersionBrowserNotification` notification.
Available in Mac OS X v10.7 and later.

8.12.57 `windowWillExitFullScreen(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: The window is about to exit fullscreen mode.

Notes: notification: A notification named `NSWindowWillExitFullScreenNotification`.
Available in Mac OS X v10.7 and later.

8.12.58 `windowWillExitVersionBrowser(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate the the window just left version browsing.

Notes: notification: An `NSWindowWillExitVersionBrowserNotification` notification.
Available in Mac OS X v10.7 and later.

8.12.59 `windowWillMiniaturize(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window is about to be minimized.

Notes: notification: A notification named `NSWindowWillMiniaturizeNotification`.
You can retrieve the `NSWindow` object in question by sending `object` to notification.
Available in Mac OS X v10.0 and later.

8.12.60 `windowWillMove(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window is about to move.

Notes: notification: A notification named `NSWindowWillMoveNotification`.
You can retrieve the `NSWindow` object in question by sending `object` to notification.
Available in Mac OS X v10.0 and later.

8.12.61 `windowWillResize(win as NSWindowMBS, newFrameSize as NSSizeMBS, newSize as NSSizeMBS) as NSSizeMBS`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window is being resized (whether by the user or through one of the setFrame... methods other than setFrame:display:).

Notes: The plugin first passes the event to Xojo runtime. In the newSize parameter you get the result from the runtime. Now you can decide. If you return nil, the newSize value is used.

win: The window being resized.

frameSize: The size to which the specified window is being resized.

newSize: The size Xojo wants to use.

Return a custom size to which the specified window will be resized.

The frameSize contains the size (in screen coordinates) sender will be resized to. To resize to a different size, simply return the desired size from this method; to avoid resizing, return the current size. sender's minimum and maximum size constraints have already been applied when this method is invoked.

While the user is resizing a window, the delegate is sent a series of windowWillResize messages as the window's outline is dragged. The window's outline is displayed at the constrained size as set by this method.

Available in Mac OS X v10.0 and later.

8.12.62 `windowWillReturnUndoManager(win as NSWindowMBS) as NSUndoManagerMBS`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Tells the delegate that the window's undo manager has been requested. Returns the appropriate undo manager for the window.

Notes: win: The window whose undo manager is being requested.

Return the appropriate undo manager for the specified window.

If this method is not implemented by the delegate, the window creates an NSUndoManager for window.

Available in Mac OS X v10.0 and later.

8.12.63 `windowWillStartLiveResize(notification as NSNotificationMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Informs the delegate that the window is about to be live resized.

Notes: notification: A notification named `NSWindowWillStartLiveResizeNotification`.

You can retrieve the window object in question by sending object to notification.
Available in Mac OS X v10.6 and later.

8.12.64 `windowWillUseStandardFrame(win as NSWindowMBS, newFrame as NSRectMBS) as NSRectMBS`

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Invoked by `NSWindow`'s `zoom:` method while determining the frame a window may be zoomed to.

Notes: win: The window whose frame size is being determined.

newFrame: The size of the current screen, which is the screen containing the largest part of the window's current frame, possibly reduced on the top, bottom, left, or right, depending on the current interface style. The frame is reduced on the top to leave room for the menu bar.

Return the specified window's standard frame.

The standard frame for a window should supply the size and location that are "best" for the type of information shown in the window, taking into account the available display or displays. For example, the best width for a window that displays a word-processing document is the width of a page or the width of the display, whichever is smaller. The best height can be determined similarly. On return from this method, the `zoom:` method modifies the returned standard frame, if necessary, to fit on the current screen.

Available in Mac OS X v10.0 and later.

8.13 class NSWorkspaceAuthorizationMBS

8.13.1 class NSWorkspaceAuthorizationMBS

Plugin Version: 19.3, Platform: macOS, Targets: All.

Function: The authorization granted to the app by the user.

Notes: To enable your app to prompt the user for these file permissions, you must have a Privileged File Operation entitlement. If you have an app on the Mac App Store or plan to submit your app for review, you can request this entitlement (see <https://developer.apple.com/go/?id=workspace-authorization>).

see also

<https://developer.apple.com/documentation/appkit/nsworkspaceauthorization>

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 19.3](#)
- [MBS Xojo Plugins, version 19.3pr3](#)

Xojo Developer Magazine

- [17.5, page 9: News](#)

8.13.2 Methods

8.13.3 Constructor

Plugin Version: 19.3, Platform: macOS, Targets: All.

Function: The private constructor.

8.13.4 Properties

8.13.5 Handle as Integer

Plugin Version: 19.3, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

8.13.6 Constants

Authorization types

Constant	Value	Description
<code>NSWorkspaceAuthorizationTypeCreateSymbolicLink</code>	0	Authorization for the app to create a symbolic link.
<code>NSWorkspaceAuthorizationTypeReplaceFile</code>	2	Authorization for the app to perform an atomic file write v target file's permissions.
<code>NSWorkspaceAuthorizationTypeSetAttributes</code>	1	Authorization for the app to change specific file attributes

8.14 class NSWorkspaceMBS

8.14.1 class NSWorkspaceMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: An NSWorkspace object responds to application requests to perform a variety of services.

Example:

```
// get icon image
dim n as NSImageMBS = NSWorkspaceMBS.iconForFile(SpecialFolder.desktop)
// set the size we want
n.setSize 512,512
// make a copy as picture
Backdrop = n.CopyPictureWithMask
```

Notes:

- Opening, manipulating, and obtaining information about files and devices
- Tracking changes to the file system, devices, and the user database
- Launching applications

Blog Entries

- [MBS Xojo Plugins, version 23.2pr5](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 20.2](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 19.3](#)
- [IconFamilyMBS class deprecated](#)
- [Custom menu checkmarks](#)
- [Adding custom icons for MacOS in Xojo](#)
- [\[ANN \] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.4](#)
- [Tip of the day: Windows Notifications](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 9.2](#)
- [MonkeyBread Software Releases the MBS Plugins 8.1](#)

Xojo Developer Magazine

- [17.5, page 9: News](#)
- [13.1, page 10: News](#)

8.14.2 Methods

8.14.3 `absolutePathForAppBundleWithIdentifier(bundleIdentifier as string) as string`

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

Function: Returns the absolute file-system path of an application bundle.

Notes: `bundleIdentifier`: The bundle identifier string. This value corresponds to the value in the `CFBundleIdentifier` key of the application's `Info.plist` file. For example, the bundle identifier of the `TextEdit` application is `com.apple.TextEdit`.

Returns the file system path to the application bundle identified by `bundleIdentifier`, or `""` if the bundle cannot be found.

8.14.4 `activateFileViewerSelectingFiles(Files() as folderitem)`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Activates the Finder, and opens one or more windows selecting the specified files.

Example:

```
dim file as FolderItem = SpecialFolder.Pictures.Child("mbs.jpg") // some file
```

```
dim w as new NSWorkspaceMBS
```

```
dim files() as FolderItem
```

```
files.Append file
```

```
// show in Finder
```

```
w.activateFileViewerSelectingFiles(files)
```

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

8.14.5 `activateFileViewerSelectingURLs(URLs() as string)`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Activates the Finder, and opens one or more windows selecting the specified files.

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

8.14.6 desktopImageOptionsForScreen(screen as NSScreenMBS) as dictionary

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns the desktop image options for the given screen.

Example:

```
dim w as new NSWorkspaceMBS
dim m as NSScreenMBS = NSScreenMBS.mainScreen
dim dic as Dictionary = w.desktopImageOptionsForScreen(m)

break // check in debugger
```

Notes: screen: The screen for which to get the desktop image options.

Returns a dictionary containing key-value pairs. Keys can be NSWorkspaceDesktopImageScalingKey, NSWorkspaceDesktopImageAllowClippingKey or NSWorkspaceDesktopImageFillColorKey.

Available in Mac OS X v10.6 and later.

8.14.7 desktopImageURLForScreen(screen as NSScreenMBS) as folderitem

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns the folderitem for the desktop image for the given screen.

Example:

```
dim w as new NSWorkspaceMBS
dim m as NSScreenMBS = NSScreenMBS.mainScreen
dim file as FolderItem = w.desktopImageURLForScreen(m)
```

```
MsgBox file.NativePath
```

Notes: screen: The screen for which to get the desktop image.

Returns the desktop image.

Available in Mac OS X v10.6 and later.

8.14.8 fileLabelColors as NSColorMBS()

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the corresponding array of file label colors for the file labels.

Example:

```

dim w as new NSWorkspaceMBS
dim labels() as string = w.fileLabels
dim colors() as NSColorMBS = w.fileLabelColors
dim lines() as string

dim u as Integer = UBound(Colors)
for i as Integer = 0 to u
dim co as NSColorMBS = colors(i)

lines.Append labels(i)+"": "+str(co.colorValue)
next

MsgBox Join(lines,EndOfLine)

```

Notes: This array has the same number of elements as fileLabels, and the color at a given index corresponds to the label at the same index.

You can listen for notifications named NSWorkspaceDidChangeFileLabelsNotification to be notified when file labels change which may result in changes to the order of the fileLabelColors.

Available in Mac OS X v10.6 and later.

8.14.9 fileLabels as string()

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the array of file labels as strings.

Example:

```

dim w as new NSWorkspaceMBS
dim labels() as string = w.fileLabels

MsgBox Join(labels, EndOfLine)

```

Notes: You can listen for notifications named NSWorkspaceDidChangeFileLabelsNotification to be notified when file labels change.

Available in Mac OS X v10.6 and later.

8.14.10 findApplications

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Examines all applications and updates the records of registered services and file types.

Example:

```
NSWorkspaceMBS.findApplications
```

8.14.11 frontmostApplication as NSRunningApplicationMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Gets the frontmost application, which is the application that will receive key events.

Example:

```
dim w as new NSWorkspaceMBS
dim n as NSRunningApplicationMBS = w.frontmostApplication
```

```
MsgBox n.localizedName
```

Notes: Requires Mac OS X 10.7.

8.14.12 fullPathForApplication(appname as string) as folderitem

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the full path for the specified application.

Example:

```
MsgBox NSWorkspaceMBS.fullPathForApplication("textedit").ShellPath
// shows /Applications/TextEdit.app
```

Notes: The full path for the application, or nil if the specified application was not found.

8.14.13 hideOtherApplications

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Hides all applications other than the sender.

Example:

```
NSWorkspaceMBS.hideOtherApplications
```

Notes: The user can hide all applications except the current one by Command-Option-clicking on an application's Dock icon.

8.14.14 iconForFile(file as folderitem) as NSImageMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an image containing the icon for the specified file.

Example:

```
dim f as FolderItem
f=SpecialFolder.Desktop.Child("test.txt")

dim n as NSImageMBS = NSWorkspaceMBS.iconForFile(f)
n.size = new NSSizeMBS(512,512)
Backdrop=n.CopyPictureWithMask
```

Notes: The returned image has an initial size of 32 pixels by 32 pixels.
Returns nil on any error.

8.14.15 iconForFiles(files() as folderitem) as NSImageMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an image containing the icon for the specified files.

Example:

```
dim w as new NSWorkspaceMBS
dim files() as FolderItem

dim folder as FolderItem = SpecialFolder.Pictures

// one file
files.Append folder.TrueItem(2)
canvas1.Backdrop = w.iconForFiles(files).CopyPictureWithMask

// two files
files.Append folder.TrueItem(3)
canvas2.Backdrop = w.iconForFiles(files).CopyPictureWithMask
```

```
// three files
files.Append folder.TrueItem(4)
canvas3.Backdrop = w.iconForFiles(files).CopyPictureWithMask
```

Notes: files: An array of folderitems, each of which contains the full path to a file.

Returns the icon associated with the group of files.

If fullPaths specifies one file, that file's icon is returned. If fullPaths specifies more than one file, an icon representing the multiple selection is returned.

8.14.16 iconForFileType(filetype as string) as NSImageMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an image containing the icon for files of the specified type.

Example:

```
const SizeWanted = 512
const typeWanted = "txt"

dim n as NSImageMBS = NSWorkspaceMBS.iconForFileType(typeWanted)
// set size we want
n.setSize SizeWanted*2, SizeWanted*2

// make picture
dim p as Picture = n.CopyPictureWithMask

// set DPI
p.VerticalResolution = 144
p.HorizontalResolution = 144
```

Notes: filetype: The file type, which may be either a filename extension or an encoded HFS file type.

The returned image has an initial size of 32 pixels by 32 pixels.
Returns nil on any error.

Running this in a thread can lead to crashes.

8.14.17 isFilePackageAtPath(item as folderitem) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Determines whether the specified path is a file package.

Example:

```
dim w as new NSWorkspaceMBS
dim file as FolderItem = SpecialFolder.Applications.Child("iTunes.app")

// shows true for iTunes
MsgBox str(w.isFilePackageAtPath(file))
```

Notes: Returns true if the path identifies a file package; otherwise, false if the path does not exist, is not a directory, or is not a file package.

8.14.18 launchApplication(appname as string) as boolean

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Launches the specified application.

Example:

```
if NSWorkspaceMBS.launchApplication("TextEdit") then
MsgBox "Ok"
else
MsgBox "failed"
end if
```

Notes: Returns true if the application was successfully launched or was already running; otherwise, false.

The appName parameter need not be specified with a full path and, in the case of an application wrapper, may be specified with or without the .app extension, as described in "Use of .app Extension".

See also:

- 8.14.19 launchApplication(appname as string, showicon as boolean, autolaunch as boolean) as boolean
420

8.14.19 launchApplication(appname as string, showicon as boolean, autolaunch as boolean) as boolean

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Launches the specified application using additional options.

Example:

```
if NSWorkspaceMBS.launchApplication("TextEdit",false,false) then
  MsgBox "Ok"
else
  MsgBox "failed"
end if
```

Notes: appName: The name of the application to open.

showIcon: If false, the application's icon is not placed on the screen. (The icon still exists, though.)

autolaunch: If true, the autolaunch default is set as though the specified application were autolaunched at startup.

This method is provided to enable daemon-like applications that lack a normal user interface. Its use is not generally encouraged.

Returns true if the application is successfully launched or already running, and false if it can't be launched.

See also:

- 8.14.18 launchApplication(appname as string) as boolean

420

8.14.20 launchApplicationAtFile(file as folderitem, options as UInt32 = 0, configuration as dictionary = nil) as NSRunningApplicationMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Launches the app at the specified file location.

Notes: file: The application folderitem.

options: Options to use when launching the application. see NSWorkspaceLaunch* constants.

configuration: A dictionary containing the configuration options. Possible key-value pairs are described NSWorkspaceLaunchConfiguration* functions.

error: The error is returned here.

Returns reference to newly started application.

Available in Mac OS X v10.6 and later.

See also:

- 8.14.21 launchApplicationAtFile(file as folderitem, options as UInt32, configuration as dictionary, byref error as NSErrorMBS) as NSRunningApplicationMBS

422

8.14.21 `launchApplicationAtFile`(file as folderitem, options as UInt32, configuration as dictionary, byref error as NSErrorMBS) as NSRunningApplicationMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Launches the app at the specified file location.

Example:

```
dim w as new NSWorkspaceMBS
dim file as FolderItem = SpecialFolder.Applications.Child("Address Book.app")

dim error as NSErrorMBS
dim configuration as new Dictionary
dim options as Integer

// today we start 32 bit version
configuration.Value(w.NSWorkspaceLaunchConfigurationArchitecture) = w.NSBundleExecutableArchitectureI386

// and hide all others
options = w.NSWorkspaceLaunchAndHideOthers

dim r as NSRunningApplicationMBS = w.launchApplicationAtFile(file, options, configuration, error)

if r = nil then
  MsgBox "Error: " + error.LocalizedDescription
else
  MsgBox "Started: " + r.localizedName
end if
```

Notes: file: The application folderitem.

options: Options to use when launching the application. see `NSWorkspaceLaunch*` constants.

configuration: A dictionary containing the configuration options. Possible key-value pairs are described `NSWorkspaceLaunchConfiguration*` functions.

error: The error is returned here.

Returns reference to newly started application.

Available in Mac OS X v10.6 and later.

See also:

- 8.14.20 `launchApplicationAtFile`(file as folderitem, options as UInt32 = 0, configuration as dictionary = nil) as NSRunningApplicationMBS 421

8.14.22 `launchApplicationAtURL(URL as string, options as UInt32 = 0, configuration as dictionary = nil) as NSRunningApplicationMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Launches the app at the specified URL.

Notes: url: The application URL.

options: Options to use when launching the application. see `NSWorkspaceLaunch*` constants.

configuration: A dictionary containing the configuration options. Possible key-value pairs are described `NSWorkspaceLaunchConfiguration*` functions.

error: The error is returned here.

Returns reference to newly started application.

Available in Mac OS X v10.6 and later.

See also:

- 8.14.23 `launchApplicationAtURL(URL as string, options as UInt32, configuration as dictionary, byref error as NSErrorMBS) as NSRunningApplicationMBS` 423

8.14.23 `launchApplicationAtURL(URL as string, options as UInt32, configuration as dictionary, byref error as NSErrorMBS) as NSRunningApplicationMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Launches the app at the specified URL.

Notes: url: The application URL.

options: Options to use when launching the application. see `NSWorkspaceLaunch*` constants.

configuration: A dictionary containing the configuration options. Possible key-value pairs are described `NSWorkspaceLaunchConfiguration*` functions.

error: The error is returned here.

Returns reference to newly started application.

Available in Mac OS X v10.6 and later.

See also:

- 8.14.22 `launchApplicationAtURL(URL as string, options as UInt32 = 0, configuration as dictionary = nil) as NSRunningApplicationMBS` 423

8.14.24 `launchAppWithBundleIdentifier(bundleIdentifier as string, options as Integer = &h00030000, AppleEventDescriptor as Variant = nil) as Boolean`

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

Function: Launches the application corresponding to the specified bundleIdentifier.

Notes: bundleIdentifier: A bundle identifier string. This value corresponds to the value in the CFBundleIdentifier key of the application's Info.plist file. For example, the bundle identifier of the TextEdit application is com.apple.TextEdit.

options: Options to use when launching the application. Values for this parameter are described in constants.
descriptor: Additional options specified in an AppleEvent-style descriptor. For example, you could use this parameter to specify additional documents to open when the application is launched.

Returns true if the application was found and launched; otherwise, false.

8.14.25 `localizedDescriptionForType(typeName as string) as string`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the localized description for the specified Uniform Type Identifier.

Example:

```
MsgBox NSWorkspaceMBS.localizedDescriptionForType("public.jpeg")
// "JPEG-Bild" in German
```

Notes: The localized description is suitable for displaying to the user.

Available in Mac OS X v10.5 and later.

8.14.26 `menuBarOwningApplication as NSRunningApplicationMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Gets the menu bar owning application, which is the application that currently owns and draws the menu bar.

Example:

```
dim w as new NSWorkspaceMBS
dim n as NSRunningApplicationMBS = w.menuBarOwningApplication
```

```
MsgBox n.localizedName
```

Notes: Requires Mac OS X 10.7.

8.14.27 `mountedLocalVolumePaths` as `string()`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the mount points of all local volumes, not just the removable ones returned by `mountedRemovableMedia`.

Example:

```
dim w as new NSWorkspaceMBS
dim paths() as string = w.mountedLocalVolumePaths
```

```
MsgBox Join(paths, EndOfLine)
```

Notes: Returns an array of strings, each of which contains the full pathname of the mount point for any local volumes.

8.14.28 `mountedRemovableMedia` as `string()`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the full pathnames of all currently mounted removable disks.

Example:

```
dim w as new NSWorkspaceMBS
dim paths() as string = w.mountedRemovableMedia
```

```
MsgBox Join(paths, EndOfLine)
```

Notes: Returns an array of strings, each of which contains the full pathname of a mounted removable disk.

If the computer provides an interrupt or other notification when the user inserts a disk into a drive, the Finder will mount the disk immediately. However, if no notification is given, the Finder won't be aware that a disk needs to be mounted. On such systems, an application should invoke either `mountNewRemovableMedia` or `checkForRemovableMedia` before invoking `mountedRemovableMedia`. Either of these methods cause the Finder to poll the drives to see if a disk is present. If a disk has been inserted but not yet mounted, these methods will cause the Finder to mount it.

The Disk button in an Open or Save panel invokes `mountedRemovableMedia` and `mountNewRemovableMedia` as part of its operation, so most applications won't need to invoke these methods directly.

8.14.29 `noteFileSystemChanged`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Informs the `NSWorkspace` object that the file system has changed.

Notes: The `NSWorkspace` object then gets the status of all the files and directories it is interested in and updates itself appropriately. This method is used by many objects that write or delete files.

The `NSDocument` and `NSSavePanel` objects use this method when saving a file. If you create a file directly, you should call `noteFileSystemChanged` so that the Finder can update the folder if it is open.

Available in Mac OS X v10.0 and later. Deprecated in Mac OS X v10.6.

See also:

- 8.14.30 `noteFileSystemChanged(path as folderitem)` 426

8.14.30 `noteFileSystemChanged(path as folderitem)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Informs the `NSWorkspace` object that the file system changed at the specified path.

Notes: The `NSWorkspace` object then gets the status of all the files and directories it is interested in and updates itself appropriately. This method is used by many objects that write or delete files.

See also:

- 8.14.29 `noteFileSystemChanged` 426

8.14.31 `notificationCenter as NSNotificationCenterMBS`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the notification center for workspace notifications.

8.14.32 `NSWorkspaceActiveSpaceDidChangeNotification as string`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when a Spaces change has occurred.

The notification object is the shared `NSWorkspace` instance. The notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.6 and later.

8.14.33 NSWorkspaceApplicationKey as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: This constant is supplied in the userInfo dictionary of various notifications.

Notes: The value corresponding to this key is an instance of NSRunningApplication that reflects the affected application.

Available in Mac OS X v10.6 and later.

8.14.34 NSWorkspaceCompressOperation as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Notes: Compress file. This operation always returns an error.

8.14.35 NSWorkspaceCopyOperation as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test")
dim tag as Integer
dim files(-1) as string
dim b as Boolean
dim source,dest as FolderItem
```

```
// copies a file from one folder to another folder
```

```
source=f.Parent
files.Append f.name
dest=SpecialFolder.Pictures
```

```
b=NSWorkspaceMBS.performFileOperation(NSWorkspaceMBS.NSWorkspaceCopyOperation, source, dest,
files, tag)
```

```
if b then
MsgBox "OK"
else
```

```
MsgBox "Failed"  
end if
```

Notes: Copy file to destination.

8.14.36 `NSWorkspaceDecompressOperation` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Notes: Decompress file. This operation always returns an error.

8.14.37 `NSWorkspaceDecryptOperation` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Notes: Decrypt file. This operation always returns an error.

8.14.38 `NSWorkspaceDesktopImageAllowClippingKey` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the options dictionary used with `SetDesktopImageURL`.

Notes: The value is a boolean, which affects the interpretation of Proportional scaling types. A false value will make the image fully visible, but there may be empty space on the sides or top and bottom. A true value will cause the image to fill the entire screen, but the image may be clipped. If this is not specified, false is assumed. Non-proportional scaling types ignore this value.

Available in Mac OS X v10.6 and later.

8.14.39 `NSWorkspaceDesktopImageFillColorKey` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the options dictionary used with `SetDesktopImageURL`.

Notes: The value is an `NSColor`, which is used to fill any empty space around the image. If not specified, a default value is used. Currently, only colors that use or can be converted to use `NSCalibratedRGBColorSpace` are supported, and any alpha value is ignored.

Available in Mac OS X v10.6 and later.

8.14.40 `NSWorkspaceDesktopImageScalingKey` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the options dictionary used with `SetDesktopImageURL`.

Notes: The value is an `Number` containing an `NSImageScaling` constant as declared in `NSCell`. If this is not specified, `NSImageScaleProportionallyUpOrDown=3` is used. `NSImageScaleProportionallyDown=0` is not currently supported.

Available in Mac OS X v10.6 and later.

8.14.41 `NSWorkspaceDestroyOperation` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Notes: Destroy file.

8.14.42 `NSWorkspaceDidActivateApplicationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder is about to activate an application.

The notification object is the shared `NSWorkspace` instance. In Mac OS X v10.6 and later the `userInfo` dictionary contains the `NSWorkspaceApplicationKey` key with a corresponding instance of `NSRunningApplication` that represents the affected application.

Available in Mac OS X v10.6 and later.

8.14.43 `NSWorkspaceDidChangeFileLabelsNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder file labels or colors change.

The notification object is the shared `NSWorkspace` instance. The notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.6 and later.

8.14.44 `NSWorkspaceDidDeactivateApplicationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder deactivated an application.

The notification object is the shared `NSWorkspace` instance. In Mac OS X v10.6 and later the `userInfo` dictionary contains the `NSWorkspaceApplicationKey` key with a corresponding instance of `NSRunningApplication` that represents the affected application.

Available in Mac OS X v10.6 and later.

8.14.45 `NSWorkspaceDidHideApplicationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder hid an application.

The notification object is the shared `NSWorkspace` instance. In Mac OS X v10.6 and later the `userInfo` dictionary contains the `NSWorkspaceApplicationKey` key with a corresponding instance of `NSRunningApplication` that represents the affected application.

Available in Mac OS X v10.6 and later.

8.14.46 `NSWorkspaceDidLaunchApplicationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when a new application has started up.

The notification object is the shared `NSWorkspace` instance. In Mac OS X v10.6 and later the `userInfo` dictionary contains the `NSWorkspaceApplicationKey` key with a corresponding instance of `NSRunningApplication` that represents the affected application.

8.14.47 `NSWorkspaceDidMountNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when a new device has been mounted.

The notification object is the shared `NSWorkspace` instance.

In Mac OS X v10.5 and earlier the `userInfo` dictionary contains a key `@”NSDevicePath”` that returns the path where the device was mounted, as a string.

Available in Mac OS X v10.0 and later.

8.14.48 `NSWorkspaceDidPerformFileOperationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when a file operation has been performed in the receiving application.

The notification object is the shared `NSWorkspace` instance. The `userInfo` dictionary contains a key `"NSOperationNumber"` with a number containing an integer indicating the type of file operation

Available in Mac OS X v10.0 and later.

8.14.49 `NSWorkspaceDidRenameVolumeNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when a volume changes its name and/or mount path. These typically change simultaneously, in which case only one notification is posted.

The notification object is the shared `NSWorkspace` instance.

Available in Mac OS X v10.6 and later.

8.14.50 `NSWorkspaceDidTerminateApplicationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when an application finishes executing.

The notification object is the shared `NSWorkspace` instance. In Mac OS X v10.6 and later the `userInfo` dictionary contains the `NSWorkspaceApplicationKey` key with a corresponding instance of `NSRunningApplication` that represents the affected application.

Available in Mac OS X v10.0 and later.

8.14.51 `NSWorkspaceDidUnhideApplicationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder unhid an application.

The notification object is the shared `NSWorkspace` instance. In Mac OS X v10.6 and later the `userInfo` dictionary contains the `NSWorkspaceApplicationKey` key with a corresponding instance of `NSRunningApplication` that represents the affected application.

Available in Mac OS X v10.6 and later.

8.14.52 `NSWorkspaceDidUnmountNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder did unmount a device.

This notification is delivered even if a volume was forcibly and immediately made unavailable, such as when a drive is simply unplugged.

The notification object is the shared `NSWorkspace` instance. The `userInfo` dictionary contains a key `"NSDevicePath"` that returns the path where the device was mounted, as a string.

Available in Mac OS X v10.0 and later.

8.14.53 `NSWorkspaceDidWakeNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the machine wakes from sleep.

The notification object is the shared `NSWorkspace` instance. The notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.3 and later.

8.14.54 `NSWorkspaceDuplicateOperation` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Notes: Duplicate file in source directory.

8.14.55 `NSWorkspaceEncryptOperation` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Notes: Encrypt file. This operation always returns an error.

8.14.56 `NSWorkspaceLaunchConfigurationAppleEvent` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the configuration dictionary for `launchApplication*` methods.

Notes: The value is the first `NSAppleEventDescriptor` to send to the new application. If an instance of the application is already running, this is sent to that application.

Available in Mac OS X v10.6 and later.

8.14.57 `NSWorkspaceLaunchConfigurationArchitecture` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the configuration dictionary for `launchApplication*` methods.

Notes: The value is a number containing an Mach-O Architecture constant. Ignored if a new instance of the application is not launched.

Available in Mac OS X v10.6 and later.

See `NSBundleExecutableArchitecture*` constants.

8.14.58 `NSWorkspaceLaunchConfigurationArguments` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the configuration dictionary for `launchApplication*` methods.

Notes: The value is an `NSArray` of `NSStrings`, passed to the new application in the `argv` parameter. Ignored if a new instance of the application is not launched.

Available in Mac OS X v10.6 and later.

8.14.59 `NSWorkspaceLaunchConfigurationEnvironment` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the configuration dictionary for `launchApplication*` methods.

Notes: The value is an dictionary, mapping `Strings` to `Strings`, containing environment variables to set for the new app. Ignored if a new instance of the application is not launched.

Available in Mac OS X v10.6 and later.

8.14.60 `NSWorkspaceLinkOperation` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test")
```

```
dim tag as Integer
```

```

dim files(-1) as string
dim b as Boolean
dim source,dest as FolderItem

// creates a hard link to a file in a folder

source=f.Parent
files.Append f.name
dest=SpecialFolder.Pictures

b=NSWorkspaceMBS.performFileOperation(NSWorkspaceMBS.NSWorkspaceLinkOperation, source, dest,
files, tag)

if b then
MsgBox "OK"
else
MsgBox "Failed"
end if

```

Notes: Create hard link to file in destination.

8.14.61 NSWorkspaceMoveOperation as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Example:

```

dim f as FolderItem = SpecialFolder.Desktop.Child("test")
dim tag as Integer
dim files(-1) as string
dim b as Boolean
dim source,dest as FolderItem

// moves a file from one folder to another folder

source=f.Parent
files.Append f.name
dest=SpecialFolder.Pictures

b=NSWorkspaceMBS.performFileOperation(NSWorkspaceMBS.NSWorkspaceMoveOperation, source, dest,
files, tag)

if b then
MsgBox "OK"

```

```

else
MsgBox "Failed"
end if

```

Notes: Move file to destination.

8.14.62 NSWorkspaceRecycleOperation as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the file operation modes.

Example:

```

dim f as FolderItem = SpecialFolder.Desktop.Child("test")
dim tag as Integer
dim files(-1) as string
dim b as Boolean
dim source,dest as FolderItem

```

```
source=f.Parent
```

```
files.Append f.name
```

```
b=NSWorkspaceMBS.performFileOperation(NSWorkspaceMBS.NSWorkspaceRecycleOperation, source, dest,
files, tag)
```

```

if b then
MsgBox "OK"
else
MsgBox "Failed"
end if

```

Notes: Move file to trash. The file is moved to the trash folder on the volume containing the file using the same semantics as NSWorkspaceMoveOperation. If a file with the same name currently exists in the trash folder, the new file is renamed. If no trash folder exists on the volume containing the file, the operation fails.

8.14.63 NSWorkspaceScreensDidSleepNotification as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the machine's screen goes to sleep.

The notification object is the shared `NSWorkspace` instance. The notification does not contain a `userInfo` dictionary.

Few applications are likely to be interested in this notification, but they may be useful for certain hardware-based drawing decisions, for example when using OpenGL.

Available in Mac OS X v10.6 and later.

8.14.64 `NSWorkspaceScreensDidWakeNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the machine's screens wake.

The notification object is the shared `NSWorkspace` instance. The notification does not contain a `userInfo` dictionary.

Few applications are likely to be interested in this notification, but they may be useful for certain hardware-based drawing decisions, for example when using OpenGL.

Available in Mac OS X v10.6 and later.

8.14.65 `NSWorkspaceSessionDidBecomeActiveNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted after a user session is switched in. This allows an application to re-enable some processing when a switched out session gets switched back in, for example.

The notification object is the shared `NSWorkspace` instance. The notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.3 and later.

8.14.66 `NSWorkspaceSessionDidResignActiveNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted before a user session is switched out. This allows an application to disable some processing

when its user session is switched out, and re-enable when that session gets switched back in, for example. The notification object is the shared `NSWorkspace` instance. The notification does not contain a `userInfo` dictionary.

If an application is launched in an inactive session, `NSWorkspaceSessionDidResignActiveNotification` is sent after `NSApplicationWillFinishLaunchingNotification` and before sending `NSApplicationDidFinishLaunchingNotification`.

Available in Mac OS X v10.3 and later.

8.14.67 `NSWorkspaceVolumeLocalizedNameKey` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the `userInfo` dictionary for the `NSWorkspaceDidRenameVolumeNotification` notification.

Notes: String containing the user-visible name of the volume.

Available in Mac OS X v10.6 and later.

8.14.68 `NSWorkspaceVolumeOldLocalizedNameKey` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the `userInfo` dictionary for the `NSWorkspaceDidRenameVolumeNotification` notification.

Notes: String containing the old user-visible name of the volume

Available in Mac OS X v10.6 and later.

8.14.69 `NSWorkspaceVolumeOldURLKey` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the `userInfo` dictionary for the `NSWorkspaceDidRenameVolumeNotification` notification.

Notes: URL containing the old mount path of the volume

Available in Mac OS X v10.6 and later.

8.14.70 `NSWorkspaceVolumeURLKey` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys for the userInfo dictionary for the `NSWorkspaceDidRenameVolumeNotification` notification.

Notes: URL containing the mount path of the volume.
Available in Mac OS X v10.6 and later.

8.14.71 `NSWorkspaceWillLaunchApplicationNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder is about to launch an application.

The notification object is the shared `NSWorkspace` instance. In Mac OS X v10.6 and later the userInfo dictionary contains the `NSWorkspaceApplicationKey` key with a corresponding instance of `NSRunningApplication` that represents the affected application.

Available in Mac OS X v10.0 and later.

8.14.72 `NSWorkspaceWillPowerOffNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the user has requested a logout or that the machine be powered off.

The notification object is the shared `NSWorkspace` instance. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.0 and later.

8.14.73 `NSWorkspaceWillSleepNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted before the machine goes to sleep. An observer of this message can delay sleep for up to 30 seconds while handling this notification.

The notification object is the shared `NSWorkspace` instance. The notification does not contain a userInfo dictionary.

Available in Mac OS X v10.3 and later.

8.14.74 NSWorkspaceWillUnmountNotification as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Posted when the Finder is about to unmount a device.

This notification will not be delivered if a volume was forcibly and immediately made unavailable, such as when a FireWire drive is simply unplugged, because there is no chance to deliver it before the volume becomes unavailable.

The notification object is the shared NSWorkspace instance. The userInfo dictionary contains a key "NSDevicePath" that returns the path where the device was mounted, as a string.

Available in Mac OS X v10.0 and later.

8.14.75 openFile(file as folderitem) as boolean

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Opens the specified file specified using the default application associated with its type.

Example:

```
dim f as FolderItem
```

```
f=SpecialFolder.Desktop.Child("test.txt")
```

```
if NSWorkspaceMBS.openFile(f) then
  MsgBox "Ok"
else
  MsgBox "failed"
end if
```

Notes: Returns true if the file was successfully opened; otherwise, false.

The sending application is deactivated before the request is sent.

See also:

- 8.14.76 openFile(file as folderitem, appname as string) as boolean 439
- 8.14.77 openFile(file as folderitem, appname as string, Deactivate as boolean) as boolean 440

8.14.76 openFile(file as folderitem, appname as string) as boolean

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Opens a file using the specified application.

Example:

```

dim f as FolderItem
f=SpecialFolder.Desktop.Child("test.txt")

if NSWorkspaceMBS.openFile(f,"BBEdit") then
MsgBox "Ok"
else
MsgBox "failed"
end if

```

Notes: Returns true if the file was successfully opened; otherwise, false.

The `appName` parameter need not be specified with a full path and, in the case of an application wrapper, may be specified with or without the `.app` extension, as described in "Use of `.app` Extension". The sending application is deactivated before the request is sent.

See also:

- 8.14.75 `openFile(file as folderitem) as boolean` 439
- 8.14.77 `openFile(file as folderitem, appName as string, Deactivate as boolean) as boolean` 440

8.14.77 `openFile(file as folderitem, appName as string, Deactivate as boolean) as boolean`

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Opens the specified file and optionally deactivates the sending application.

Example:

```

dim f as FolderItem = SpecialFolder.Desktop.Child("test.file")

if NSWorkspaceMBS.openFile(f,"BBEdit",true) then
MsgBox "Ok"
else
MsgBox "failed"
end if

```

Notes: `appName`: The name of the application to use when opening the file.

`flag`: If true, the sending application is deactivated before the request is sent, allowing the opening application to become the active application.

Returns true if the file was successfully opened; otherwise, false.

The `appName` parameter need not be specified with a full path and, in the case of an application wrapper, may be specified with or without the `.app` extension, as described in "Use of `.app` Extension". If `appName` is `nil`, the default application for the file's type is used.

See also:

- 8.14.75 `openFile(file as folderitem) as boolean` 439
- 8.14.76 `openFile(file as folderitem, appName as string) as boolean` 439

8.14.78 `openURL(url as string) as boolean`

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Opens the location at the specified URL.

Example:

```
if NSWorkspaceMBS.openURL("http://www.apple.com") then
  MsgBox "Ok"
else
  MsgBox "failed"
end if
```

Notes: Returns true if the location was successfully opened; otherwise, false.

See also:

- 8.14.79 `openURL(url as string, bundleIdentifier as string, options as Integer = &h00030000, AppleEventDescriptor as Variant = nil) as Boolean` 441

8.14.79 `openURL(url as string, bundleIdentifier as string, options as Integer = &h00030000, AppleEventDescriptor as Variant = nil) as Boolean`

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

Function: Opens one or more files from an array of URLs.

Notes: `url`: A URL for the application to open.

`bundleIdentifier`: A bundle identifier string or "" to use the default system bindings. This value corresponds to the value in the `CFBundleIdentifier` key of the application's `Info.plist` file. For example, the bundle identifier of the `TextEdit` application is `com.apple.TextEdit`.

`options`: Options to use when launching the application. Values for this parameter are described in constants.

`descriptor`: Additional options specified in an `AppleEvent`-style descriptor. For example, you could use this parameter to specify additional documents to open when the application is launched.

Returns true if the application was found and launched; otherwise, false.

See also:

- 8.14.78 `openURL(url as string)` as boolean

8.14.80 `performFileOperation(operation as string, source as folderitem, destination as folderitem, files() as string, byref tag as Integer)` as boolean

Plugin Version: 9.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Performs a file operation on a set of files in a particular directory.

Notes: `operation`: The file operation to perform. The possible values for this parameter are described in "Constants."

`source`: The full path to the directory containing the files on which to operate.

`destination`: The full path to the destination directory of the operation.

`files`: An array of folderitems specifying the names of the files and directories to be manipulated. Each string must not contain any path information other than the name of the file or directory. In other words, all of the files and directories must be located in the source directory and not in one of its subdirectories.

`tag`: On input, an integer variable; on return, this variable contains a negative integer if the operation fails, 0 if the operation was performed synchronously and succeeded, or a positive integer if the operation was performed asynchronously. If the value is a positive integer, the value is a tag that identifies the requested file operation.

Return Value: True if the operation succeeded; otherwise, false.

Discussion:

Some operations—such as moving, copying, and linking files—require a destination directory to be specified. If not, `destination` should be the empty string (""). Before this method returns, it posts an `NSWorkspaceDidPerformFileOperationNotification` to the `NSWorkspace` object's notification center.

8.14.81 `preferredFilenameExtensionForType(typeName as string)` as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the preferred filename extension for the specified Uniform Type Identifier.

Example:

```
MsgBox NSWorkspaceMBS.preferredFilenameExtensionForType("public.jpeg")
// "jpeg" in German
```

Notes: The appropriate filename extension for `typeName`, or "" if no extension could be determined.

8.14.82 requestAuthorization(type as integer, tag as variant = nil)

Plugin Version: 19.3, Platform: macOS, Targets: Desktop only.

Function: Requests authorization to perform a privileged file operation.

Notes: type: The type of file operation to perform.

Calls requestAuthorizationCompleted event later.

Tag value is passed through as is to event.

Requires the application to be code-signed.

8.14.83 selectFile(file as folderitem) as boolean

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Selects the file in the Finder.

Example:

```
dim f as FolderItem
```

```
f=SpecialFolder.Desktop.Child("test.txt")
```

```
if NSWorkspaceMBS.selectFile(f) then  
  MsgBox "Ok"  
else  
  MsgBox "failed"  
end if
```

Notes: Returns true on success.

For Windows, please use WinOpenFolderAndSelectItemsMBS function.

8.14.84 setDesktopImageURL(file as folderitem, screen as NSScreenMBS, options as dictionary, byref error as NSErrorMBS) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Sets the desktop image for the given screen to the image at the specified URL.

Notes: file: The image file. The file must not be nil.

screen: The screen to set the desktop image on.

options: The options dictionary may contain any of the Desktop Image Dictionary Keys (NSWorkspaceDesktopImageScalingKey, NSWorkspaceDesktopImageAllowClippingKey or NSWorkspaceDesktopImageFillColorKey), which control how the image is scaled on the screen.

error: A error that is returned by-reference if setting the image fails.

Returns true if the image was set as the desktop, otherwise false. If false is returned, the error parameter provides additional information.

You should not present a user interface for picking the options. Instead, choose appropriate defaults and allow the user to adjust them in the System Preference Pane.

Available in Mac OS X v10.6 and later.

8.14.85 setIcon(image as NSImageMBS, file as folderitem, flags as Integer = 0) as boolean

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the icon for the file or directory at the specified path.

Example:

```
dim f as FolderItem
```

```
dim p as Picture
```

```
dim img as NSImageMBS
```

```
f=SpecialFolder.Desktop.Child("test.txt")
```

```
p=New Picture(128,128,32)
```

```
p.Graphics.ForeColor=rgb(0,128,0)
```

```
p.Graphics.fillrect 0,0,128,128
```

```
p.Graphics.ForeColor=rgb(255,0,0)
```

```
p.Graphics.filloval 0,0,128,128
```

```
img=new NSImageMBS(p)
```

```
if NSWorkspaceMBS.setIcon(img,f,0) then
```

```
  MsgBox "Ok. Icon may not be visible directly. Maybe you make a copy of the file to see it directly?"
```

```
else
```

```
  MsgBox "failed"
```

```
end if
```

Notes: image: The image to use as the icon for the file or directory.

file: The full path of the file or directory.

flags: The icon representations to generate from the image. You specify this value by combining the appropriate `NSWorkspaceIconCreationOptions` constants, listed in `Constants`, using the C bitwise OR operator. Specify 0 if you want to generate icons in all available icon representation formats.

Returns true if the icon was set; otherwise, false.

The image can be an arbitrary image, with or without transparency. This image is automatically scaled (as needed) to generate the icon representations. The file or folder must exist and be writable by the user.

It is recommended that applications include the `NSExclude10_4ElementsIconCreationOption` option for compatibility with pre-Mac OS X v10.3 Finder. Icons that include the high resolution elements prevent custom icons from being displayed on earlier systems.

Before setting icon, make sure you close all `BinaryStream`, `TextOutputStream` or other classes which may have the file open.

See also:

- 8.14.86 `setIcon(image as NSImageMBS, path as string, flags as Integer = 0)` as boolean 445

8.14.86 `setIcon(image as NSImageMBS, path as string, flags as Integer = 0)` as boolean

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the icon for the file or directory at the specified path.

Notes: image: The image to use as the icon for the file or directory.

path: The full path of the file or directory.

flags: The icon representations to generate from the image. You specify this value by combining the appropriate `NSWorkspaceIconCreationOptions` constants, listed in `Constants`, using the C bitwise OR operator. Specify 0 if you want to generate icons in all available icon representation formats.

Returns true if the icon was set; otherwise, false.

The image can be an arbitrary image, with or without transparency. This image is automatically scaled (as needed) to generate the icon representations. The file or folder must exist and be writable by the user.

It is recommended that applications include the `NSExclude10_4ElementsIconCreationOption` option for compatibility with pre-Mac OS X v10.3 Finder. Icons that include the high resolution elements prevent custom icons from being displayed on earlier systems.

Before setting icon, make sure you close all Binarystream, Textoutputstream or other classes which may have the file open.

See also:

- 8.14.85 setIcon(image as NSImageMBS, file as folderitem, flags as Integer = 0) as boolean 444

8.14.87 showSearchResultsForQueryString(queryString as string) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Displays a Spotlight search results window in Finder for the specified query string.

Notes: Returns true if the communication with Finder was successful, otherwise false.

Finder becomes the active application, if possible. The user can further refine the search via the Finder user interface.

Available in Mac OS X v10.6 and later.

8.14.88 typeOfFile(File as folderitem, byref error as NSErrorMBS) as string

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

Function: Returns the uniform type identifier of the specified file, if it can be determined.

Notes: file: The absolute path of the file.

Error: If the Uniform Type Identifier of the file at absolutePath can't be determined, outError contains an NSError object that describes why.

Returns a string containing the uniform type identifier of the file at absoluteFilePath. If no UTI can be determined the return value is "".

If the file at the specified path is a symbolic link, the type of the symbolic link is returned.

See also:

- 8.14.89 typeOfFile(Path as string, byref error as NSErrorMBS) as string 446

8.14.89 typeOfFile(Path as string, byref error as NSErrorMBS) as string

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

Function: Returns the uniform type identifier of the specified file, if it can be determined.

Notes: file: The absolute path of the file.

Error: If the Uniform Type Identifier of the file at absolutePath can't be determined, outError contains an

NSError object that describes why.

Returns a string containing the uniform type identifier of the file at `absoluteFilePath`. If no UTI can be determined the return value is "".

If the file at the specified path is a symbolic link, the type of the symbolic link is returned.

See also:

- 8.14.88 `typeOfFile(File as folderitem, byref error as NSErrorMBS) as string` 446

8.14.90 `unmountAndEjectDevice(item as folderitem, byref e as NSErrorMBS) as boolean`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Unmounts and ejects the device at the specified path.

Example:

```
dim w as new NSWorkspaceMBS
dim disk as FolderItem = Volume(VolumeCount-1)
```

```
MsgBox disk.Name
```

```
dim e as NSErrorMBS
if w.unmountAndEjectDevice(disk, e) then
  MsgBox "OK"
else
  MsgBox "Error: " + e.localizedDescription
end if
```

Notes: Returns true if the volume was unmounted and ejected successfully, otherwise false, for example, if the volume is not ejectable.

error: If the operation fails, this error contains more information about the failure.

8.14.91 `URLForApplicationToOpenURL(url as string) as string`

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

Function: Returns the URL to the default application that would be used to open the given URL.

Notes: url: The URL of the file to open.

Returns the URL of the default application that would open the specified url. Returns "" if no application

is able to open the url, or if the file url does not exist.

This is the programmatic equivalent of double clicking a document in the Finder.

8.14.92 `URLForApplicationWithBundleIdentifier(bundleIdentifier as string) as string`

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

Function: Returns the URL for the application with the specified identifier.

Notes: `bundleIdentifier`: A bundle identifier specifying an application.

Returns the URL of the application, or "" if no application has the bundle identifier.

This uses various (currently unspecified) heuristics in case multiple apps have the same bundle ID.

8.14.93 Properties

8.14.94 `accessibilityDisplayShouldDifferentiateWithoutColor` as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Get the current accessibility display option for differentiate without color.

Notes: If this is true, UI should not convey information using color alone and instead should use shapes or glyphs to convey information. You may listen for `NSWorkspaceAccessibilityDisplayOptionsDidChangeNotification` to be notified when this changes.

(Read only property)

8.14.95 `accessibilityDisplayShouldIncreaseContrast` as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A Boolean value that indicates whether the app should present a high contrast user interface.

Notes: When this method returns true, present a high contrast UI. For example, use a less subtle color palette or bolder lines.

(Read only property)

8.14.96 `accessibilityDisplayShouldInvertColors` as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Get the current accessibility display option for invert colors.

Notes: If this property's value is true then the display will be inverted. In these cases it may be needed for UI drawing to be adjusted in order to display optimally when inverted. You may listen for `NSWorkspaceAccessibilityDisplayOptionsDidChangeNotification` to be notified when this changes.

Available on MacOS 10.12 or newer.
(Read only property)

8.14.97 `accessibilityDisplayShouldReduceMotion` as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Get the current accessibility display option for reduce motion.

Notes: If this property's value is true, UI should avoid large animations, especially those that simulate the third dimension. You may listen for `NSWorkspaceAccessibilityDisplayOptionsDidChangeNotification` to be notified when this changes.

Available on MacOS 10.12 or newer.
(Read only property)

8.14.98 `accessibilityDisplayShouldReduceTransparency` as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Get the current accessibility display option for reduce transparency.

Notes: If this property's value is true, UI (mainly window) backgrounds should not be semi-transparent; they should be opaque. You may listen for `NSWorkspaceAccessibilityDisplayOptionsDidChangeNotification` to be notified when this changes.

(Read only property)

8.14.99 `isSwitchControlEnabled` as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates if Switch Control is currently running.

Notes: (Read only property)

8.14.100 `isVoiceOverEnabled` as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates if VoiceOver is currently running.

Notes: (Read only property)

8.14.101 Events

8.14.102 requestAuthorizationCompleted(type as Integer, authorization as NSWorkspaceAuthorizationMBS, error as NSErrorMBS, tag as variant)

Plugin Version: 19.3, Platform: macOS, Targets: .

Function: The request for authorization completed.

Notes: type: The type of file operation requested.

authorization: The authorization granted for this app. Use it when creating a new NSFileManagerMBS with fileManagerWithAuthorization function.

error: nil if the app is authorized; otherwise, a pointer to the authorization error.

The completion event to call when the authorization request is completed.

8.14.103 Constants

Constants

Constant	Value	Description
NSExclude10_4ElementsIconCreationOption	4	One of the possible constants you can use with SetIconCreationOptions.
NSExcludeQuickDrawElementsIconCreationOption	2	One of the possible constants you can use with SetIconCreationOptions.
NSWorkspaceLaunchAndHide	&h00100000	One of the constants for the launch functions.
NSWorkspaceLaunchAndHideOthers	&h00200000	One of the constants for the launch functions.
NSWorkspaceLaunchAndPrint	2	One of the constants for the launch functions.
NSWorkspaceLaunchAsync	&h00010000	One of the constants for the launch functions.
NSWorkspaceLaunchDefault	&h00010000	One of the constants for the launch functions.
NSWorkspaceLaunchInhibitingBackgroundOnly	&h00000080	One of the constants for the launch functions.
NSWorkspaceLaunchNewInstance	&h00080000	One of the constants for the launch functions.
NSWorkspaceLaunchWithErrorPresentation	&h00000040	One of the constants for the launch functions.
NSWorkspaceLaunchWithoutActivation	&h00000200	One of the constants for the launch functions.
NSWorkspaceLaunchWithoutAddingToRecents	&h00000100	One of the constants for the launch functions.

Architectures

Constant	Value	Description
NSBundleExecutableArchitectureARM64	&h0100000c	Specifies the 64-bit ARM architecture.
NSBundleExecutableArchitectureI386	&h00000007	Intel 32 bit.
NSBundleExecutableArchitecturePPC	&h00000012	PPC 32 bit.
NSBundleExecutableArchitecturePPC64	&h01000012	PPC 64 bit.
NSBundleExecutableArchitectureX86_64	&h01000007	Intel 64 bit.

Authorization Types

Constant	Value	Description
NSWorkspaceAuthorizationTypeCreateSymbolicLink	0	Authorization for the app to create a symbolic link.
NSWorkspaceAuthorizationTypeReplaceFile	2	Authorization for the app to perform an atomic file write with target file's permissions.
NSWorkspaceAuthorizationTypeSetAttributes	1	Authorization for the app to change specific file attributes.

Chapter 9

Cocoa Controls

9.1 class CustomNSSearchFieldMBS

9.1.1 class CustomNSSearchFieldMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The class for a custom searchfield.

Notes: Some events will not fire as Searchfield eats them itself.
Subclass of the NSSearchFieldMBS class.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 17.0pr1](#)
- [MBS Real Studio Plugins, version 13.1pr11](#)
- [MBS Real Studio Plugins, version 12.5pr1](#)
- [MBS Real Studio Plugins, version 12.4pr9](#)

9.1.2 Methods

9.1.3 Constructor

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The dummy constructor doing nothing.

See also:

- 9.1.4 Constructor(Handle as Integer) 454
- 9.1.5 Constructor(left as Double, top as Double, width as Double, height as Double) 454

9.1.4 Constructor(Handle as Integer)

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

Function: The constructor.

See also:

- 9.1.3 Constructor 453
- 9.1.5 Constructor(left as Double, top as Double, width as Double, height as Double) 454

9.1.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The constructor for a new custom NSView object.

See also:

- 9.1.3 Constructor 453
- 9.1.4 Constructor(Handle as Integer) 454

9.1.6 Destructor

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: The destructor.

9.1.7 Events

9.1.8 acceptsFirstMouse(e as NSEventMBS) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Overridden by subclasses to return true if the receiver should be sent a mouseDown event for an initial mouse-down event, false if not.

Notes: The receiver can either return a value unconditionally or use the location of event *e* to determine whether or not it wants the event. The default implementation ignores the event and returns false.

Implement this event in a subclass to allow instances to respond to click-through. This allows the user to click on a view in an inactive window, activating the view with one click, instead of clicking first to make the window active and then clicking the view. Most view objects refuse a click-through attempt, so the event simply activates the window. Many control objects, however, such as instances of NSButton and NSSlider, do

accept them, so the user can immediately manipulate the control without having to release the mouse button.

9.1.9 acceptsFirstResponder as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Whether to accept first responder.

Notes: Return true if your control can have the focus and false if not.

9.1.10 becomeFirstResponder as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Called when the object gets focus.

Notes: Return true to accept.

9.1.11 beginGestureWithEvent(e as NSEventMBS) as boolean

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

Function: Informs the receiver that the user has begun a touch gesture.

Notes: e: An event object representing the gesture beginning.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.1.12 canBecomeKeyView as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Returns whether the receiver can become key view.

Notes: Returns true if the receiver can become key view, false otherwise.

9.1.13 Close

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: The event called when the custom view is destroyed.

9.1.14 `concludeDragOperation(sender as NSDraggingInfoMBS)`

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

Function: Invoked when the dragging operation is complete, signaling the receiver to perform any necessary clean-up.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

For this method to be invoked, the previous `performDragOperation` must have returned true.

The destination implements this method to perform any tidying up that it needs to do, such as updating its visual representation now that it has incorporated the dragged data. This message is the last message sent from sender to the destination during a dragging session.

If the sender object's `animatesToDestination` property was set to true in `prepareForDragOperation`, then the drag image is still visible. At this point you should draw the final visual representation in the view. When this method returns, the drag image is removed from the screen. If your final visual representation matches the visual representation in the drag, this is a seamless transition.

9.1.15 `draggingEnded(sender as NSDraggingInfoMBS)`

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

Function: Implement this event to be notified when a drag operation ends in some other destination.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

This method might be used by a destination doing auto-expansion in order to collapse any auto-expands.

9.1.16 `draggingEntered(sender as NSDraggingInfoMBS) as Integer`

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

Function: Invoked when the dragged image enters destination bounds or frame; delegate returns dragging operation to perform.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in `NSDragOperation` in the `NSDraggingInfo` reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous `draggingEntered` message.

Invoked when a dragged image enters the destination but only if the destination has registered for the paste-

board data type involved in the drag operation. Specifically, this method is invoked when the mouse pointer enters the destination's bounds rectangle (if it is a view object) or its frame rectangle (if it is a window object).

This method must return a value that indicates which dragging operation the destination will perform when the image is released. In deciding which dragging operation to return, the method should evaluate the overlap between both the dragging operations allowed by the source (obtained from sender with the `draggingSourceOperationMask` method) and the dragging operations and pasteboard data types the destination itself supports.

If none of the operations is appropriate, this method should return `NSDragOperationNone` (this is the default response if the method is not implemented by the destination). A destination will still receive `draggingUpdated` and `draggingExited` even if `NSDragOperationNone` is returned by this method.

9.1.17 `draggingExited(sender as NSDraggingInfoMBS)`

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

Function: Invoked when the dragged image exits the destination's bounds rectangle (in the case of a view object) or its frame rectangle (in the case of a window object).

Notes: sender: The object sending the message; use it to get details about the dragging operation.

9.1.18 `draggingSessionEndedAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS, operation as Integer)`

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

Function: Invoked when the dragging session has completed.

Notes: session: The dragging session.

screenPoint: The point where the drag ended, in screen coordinates.

operation: The drag operation. See constants for drag operation types.

Available in OS X v10.7 and later.

9.1.19 `draggingSessionMovedToPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)`

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

Function: Invoked when the drag moves on the screen.

Notes: session: The dragging session.

screenPoint: The point where the drag moved to, in screen coordinates.

Available in OS X v10.7 and later.

9.1.20 draggingSessionSourceOperationMaskForDraggingContext(session as NSDraggingSessionMBS, context as Integer) as Integer

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

Function: Declares the types of operations the source allows to be performed. (required)

Notes: session: The dragging session.

context: The dragging context. See NSDraggingContext constants for the supported values.

Return the appropriate dragging operation as defined in constants.

In the future Apple may provide more specific "within" values in the future. To account for this, for unrecognized localities, return the operation mask for the most specific context that you are concerned with.

9.1.21 draggingSessionWillBeginAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)

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

Function: Invoked when the drag will begin.

Notes: session: The dragging session.

screenPoint: The point where the drag will begin, in screen coordinates.

Available in OS X v10.7 and later.

9.1.22 draggingUpdated(sender as NSDraggingInfoMBS) as Integer

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

Function: Invoked periodically as the image is held within the destination area, allowing modification of the dragging operation or mouse-pointer position.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is

the value returned by the previous `draggingEntered` message.

For this to be invoked, the destination must have registered for the pasteboard data type involved in the drag operation. The messages continue until the image is either released or dragged out of the window or view.

This method provides the destination with an opportunity to modify the dragging operation depending on the position of the mouse pointer inside of the destination view or window object. For example, you may have several graphics or areas of text contained within the same view and wish to tailor the dragging operation, or to ignore the drag event completely, depending upon which object is underneath the mouse pointer at the time when the user releases the dragged image and the `performDragOperation` method is invoked.

You typically examine the contents of the pasteboard in the `draggingEntered` method, where this examination is performed only once, rather than in the `draggingUpdated` method, which is invoked multiple times.

Only one destination at a time receives a sequence of `draggingUpdated` messages. If the mouse pointer is within the bounds of two overlapping views that are both valid destinations, the uppermost view receives these messages until the image is either released or dragged out.

9.1.23 `endGestureWithEvent(e as NSEventMBS)` as boolean

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

Function: Informs the receiver that the user has ended a touch gesture.

Notes: `e`: An event object representing the gesture end.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.1.24 `ignoreModifierKeysForDraggingSession(session as NSDraggingSessionMBS)` as boolean

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

Function: Returns whether the modifier keys will be ignored for this dragging session.

Notes: `session`: The dragging session.

Return true if the modifier keys will be ignored, false otherwise.

Available in OS X v10.7 and later.

9.1.25 isOpaque as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Whether this view is opaque.

9.1.26 keyDown(e as NSEventMBS) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the key events.

Notes: Return true if you handled this event.

Please return true in `becomeFirstResponder` and `acceptsFirstResponder`, so your `nview` can become first responder and receive key events.

9.1.27 keyUp(e as NSEventMBS) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the key events.

Notes: Return true if you handled this event.

Please return true in `becomeFirstResponder` and `acceptsFirstResponder`, so your `nview` can become first responder and receive key events.

9.1.28 magnifyWithEvent(e as NSEventMBS) as boolean

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

Function: Informs the receiver that the user has begun a pinch gesture.

Notes: `e`: An event object representing the magnify gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.1.29 menuForEvent(e as NSEventMBS, defaultMenu as NSMenuMBS) as NSMenuMBS

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Overridden by subclasses to return a context-sensitive pop-up menu for a given mouse-down event.

Notes: theEvent: An object representing a mouse-down event.
defaultMenu: The menu as constructed by super class.

The receiver can use information in the mouse event, such as its location over a particular element of the receiver, to determine what kind of menu to return. For example, a text object might display a text-editing menu when the cursor lies over text and a menu for changing graphics attributes when the cursor lies over an embedded image.

The default implementation returns the default menu.

9.1.30 mouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.31 mouseDownCanMoveWindow as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: This event is called so you can decide what happens with mouse down.

Notes: Return true if you do not need to handle a mouse down and it can pass through to superviews; False if you need to handle the mouse down.

This allows iApp-type applications to determine the region by which a window can be moved. By default, this method returns false if the view is opaque; otherwise, it returns true. Subclasses can override this method to return a different value.

9.1.32 mouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.33 mouseEntered(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.34 `mouseExited(e as NSEventMBS, x as Double, y as Double)` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.35 `mouseMoved(e as NSEventMBS, x as Double, y as Double)` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.36 `mouseUp(e as NSEventMBS, x as Double, y as Double)` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.37 `Open`

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: The event called when the custom `NSSearchField` is created.

9.1.38 `otherMouseDown(e as NSEventMBS, x as Double, y as Double)` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.1.39 otherMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.1.40 otherMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.1.41 performDragOperation(sender as NSDraggingInfoMBS) as boolean

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

Function: Invoked after the released image has been removed from the screen, signaling the receiver to import the pasteboard data.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return if the destination accepts the data, it returns true; otherwise it returns false. The default is to return false.

For this method to be invoked, the previous `prepareForDragOperation` message must have returned true. The destination should implement this method to do the real work of importing the pasteboard data represented by the image.

If the sender object's `animatesToDestination` was set to true in `prepareForDragOperation`, then setup any animation to arrange space for the drag items to animate to. Also at this time, enumerate through the dragging items to set their destination frames and destination images.

9.1.42 prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean

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

Function: Invoked when the image is released, allowing the receiver to agree to or refuse drag operation.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns true if the receiver agrees to perform the drag operation and false if not.

This method is invoked only if the most recent draggingEntered or draggingUpdated message returned an acceptable drag-operation value.

If you want the drag items to animate from their current location on screen to their final location in your view, set the sender object's animatesToDestination property to true in your implementation of this method.

9.1.43 pressureChange(e as NSEventMBS) as boolean

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

Function: Informs the current object that a pressure change occurred on a system that supports pressure sensitivity.

Notes: This method is invoked automatically in response to user actions. event is the event that initiated the change in pressure.

Available in OS X v10.10.3 and later.

9.1.44 resignFirstResponder as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Focus is going away.

Notes: Return true to accept.

9.1.45 rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.46 rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.47 rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.1.48 rotateWithEvent(e as NSEventMBS) as boolean

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

Function: Informs the receiver that the user has begun a rotation gesture.

Notes: e: An event object representing the rotate gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.1.49 scrollWheel(e as NSEventMBS) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Informs the subclass that the mouse's scroll wheel has moved.

Notes: e: An object encapsulating information about the wheel-scrolling event.

The default implementation simply passes this message to the next responder.

Return true to not pass the event.

9.1.50 swipeWithEvent(e as NSEventMBS) as boolean

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

Function: Informs the receiver that the user has begun a swipe gesture.

Notes: e: An event object representing the swipe gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.1.51 updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)

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

Function: Invoked when the dragging images should be changed.

Notes: sender: The object sending the message; use this object to get details about the dragging operation.

While a destination may change the dragging images at any time, it is recommended to wait until this method is called before updating the dragging images.

This allows the system to delay changing the dragging images until it is likely that the user will drop on this destination. Otherwise, the dragging images will change too often during the drag which would be distracting to the user.

9.1.52 `viewDidMoveToWindow`

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Informs the receiver that it has been added to a new view hierarchy.

Notes: The default implementation does nothing; subclasses can implement this event to perform whatever actions are necessary.

`window` may return `nil` when this method is invoked, indicating that the receiver does not currently reside in any window. This occurs when the receiver has just been removed from its superview or when the receiver has just been added to a superview that does not itself have a window. Overrides of this method may choose to ignore such cases if they are not of interest.

9.1.53 `wantsPeriodicDraggingUpdates` as boolean

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

Function: Asks the destination object whether it wants to receive periodic `draggingUpdated` messages.

Notes: Return `true` if the destination wants to receive periodic `draggingUpdated` messages, `false` otherwise.

If the destination returns `false`, these messages are sent only when the mouse moves or a modifier flag changes. Otherwise the destination gets the default behavior, where it receives periodic `draggingUpdated` messages even if nothing changes.

9.2 class CustomNSTextFieldCellMBS

9.2.1 class CustomNSTextFieldCellMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The class to customize cells.

Notes: Subclass of the NSTextFieldCellMBS class.

Blog Entries

- [MBS Xojo Plugins, version 17.2pr1](#)

9.2.2 Methods

9.2.3 Constructor

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The constructor.

9.2.4 superDrawWithFrame(frame as NSRectMBS, view as NSViewMBS)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Calls drawWithFrame on super class.

Notes: This is for calling in DrawWithFrame event.

9.2.5 Events

9.2.6 cellSize(size as NSSizeMBS) as NSSizeMBS

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Return a custom cell size.

Notes: If not implemented, we call through to super.cellSize.

We provide super.cellSize in size parameter.

9.2.7 Clone(clonedCell as NSTextFieldCellMBS) as CustomNSTextFieldCellMBS

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Requests a clone of the object.

Notes: This event is called when the system needs a clone of the object.

Please create a new object, keep a reference and return it.

SuperClone provides the cloned object, which we use together with the xoyo object you return.

9.2.8 didDrawWithFrame(cellFrame as NSRectMBS, controlView as NSViewMBS)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Event called after drawWithFrame run.

9.2.9 drawWithFrame(cellFrame as NSRectMBS, controlView as NSViewMBS) as boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Event to replace drawWithFrame.

Notes: If false is returned or not implemented, we call super.drawWithFrame.

9.2.10 fieldEditorForView(controlView as NSViewMBS) as NSTextViewMBS

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Returns a custom field editor for editing in the view.

Notes: ,Ã~Control,Ã~View: The view containing cells that require a custom field editor.

Returns a custom field editor. The field editor must have field,Ã~Editor set to true.

This is an override point for NSCell subclasses designed to use their own custom field editors. This message is sent to the selected cell of ,Ã~Control,Ã~View using the NSWindow method in field,Ã~Editor,Ã~.

Returning non-nil from this method indicates skipping the standard field editor querying processes including window,Ã~Will,Ã~Return,Ã~Field,Ã~Editor delegation.

The default implementation returns nil.

9.2.11 imageRectForBounds(rect as NSRectMBS) as NSRectMBS

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event for imageRectForBounds method.

9.2.12 selectWithFrame(rect as NSRectMBS, controlView as NSViewMBS, text as NSTextMBS, theDelegate as Variant, selStart as Integer, selLength as Integer) as boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event for selectWithFrame method.

Notes: If you return false, we call super.selectWithFrame.

9.2.13 setUpFieldEditorAttributes(textObj as NSTextMBS, superFieldEditor as NSTextMBS) as NSTextMBS

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Sets up the field editor. You never invoke this method directly; by overriding it, however, you can customize the field editor.

Notes: When you override this method, you should generally invoke the implementation of super and return the text,ÄObj argument. For information on field editors, see Using the Window,Äs Field Editor.

9.2.14 titleRectForBounds(rect as NSRectMBS) as NSRectMBS

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event for titleRectForBounds method.

Notes: If you return nil, we call super.titleRectForBounds.

9.3 class CustomNSTextFieldMBS

9.3.1 class CustomNSTextFieldMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The class for a custom text field.

Notes: Some events will not fire as text field eats them itself.

Subclass of the NSTextFieldMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 12.5pr3](#)

9.3.2 Methods

9.3.3 Constructor

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The constructor for a new custom NSTextField object.

See also:

- 9.3.4 Constructor(Handle as Integer) 470
- 9.3.5 Constructor(left as Double, top as Double, width as Double, height as Double) 470

9.3.4 Constructor(Handle as Integer)

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

Function: The constructor.

Notes: Pass a valid handle to a NSTextField to initialize.

See also:

- 9.3.3 Constructor 470
- 9.3.5 Constructor(left as Double, top as Double, width as Double, height as Double) 470

9.3.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The constructor for a new custom NSTextField object.

See also:

9.3. CLASS CUSTOMNSTEXTFIELDMBS	471
• 9.3.3 Constructor	470
• 9.3.4 Constructor(Handle as Integer)	470

9.3.6 Destructor

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: The destructor.

9.3.7 Events

9.3.8 acceptsFirstMouse(e as NSEventMBS) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Overridden by subclasses to return true if the receiver should be sent a mouseDown event for an initial mouse-down event, false if not.

Notes: The receiver can either return a value unconditionally or use the location of event e to determine whether or not it wants the event. The default implementation ignores the event and returns false.

Implement this event in a subclass to allow instances to respond to click-through. This allows the user to click on a view in an inactive window, activating the view with one click, instead of clicking first to make the window active and then clicking the view. Most view objects refuse a click-through attempt, so the event simply activates the window. Many control objects, however, such as instances of NSButton and NSSlider, do accept them, so the user can immediately manipulate the control without having to release the mouse button.

9.3.9 acceptsFirstResponder as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Whether to accept first responder.

Notes: Return true if your control can have the focus and false if not.

9.3.10 becomeFirstResponder as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Called when the object gets focus.

Notes: Return true to accept.

9.3.11 beginGestureWithEvent(e as NSEventMBS) as boolean

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

Function: Informs the receiver that the user has begun a touch gesture.

Notes: e: An event object representing the gesture beginning.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.3.12 canBecomeKeyView as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Returns whether the receiver can become key view.

Notes: Returns true if the receiver can become key view, false otherwise.

9.3.13 Close

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: The event called when the custom view is destroyed.

9.3.14 concludeDragOperation(sender as NSDraggingInfoMBS)

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

Function: Invoked when the dragging operation is complete, signaling the receiver to perform any necessary clean-up.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

For this method to be invoked, the previous performDragOperation must have returned true.

The destination implements this method to perform any tidying up that it needs to do, such as updating its visual representation now that it has incorporated the dragged data. This message is the last message sent from sender to the destination during a dragging session.

If the sender object's animatesToDestination property was set to true in prepareForDragOperation, then the drag image is still visible. At this point you should draw the final visual representation in the view. When this method returns, the drag image is removed from the screen. If your final visual representation matches the visual representation in the drag, this is a seamless transition.

9.3.15 draggingEnded(sender as NSDraggingInfoMBS)

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

Function: Implement this event to be notified when a drag operation ends in some other destination.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

This method might be used by a destination doing auto-expansion in order to collapse any auto-expands.

9.3.16 draggingEntered(sender as NSDraggingInfoMBS) as Integer

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

Function: Invoked when the dragged image enters destination bounds or frame; delegate returns dragging operation to perform.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered message.

Invoked when a dragged image enters the destination but only if the destination has registered for the pasteboard data type involved in the drag operation. Specifically, this method is invoked when the mouse pointer enters the destination's bounds rectangle (if it is a view object) or its frame rectangle (if it is a window object).

This method must return a value that indicates which dragging operation the destination will perform when the image is released. In deciding which dragging operation to return, the method should evaluate the overlap between both the dragging operations allowed by the source (obtained from sender with the draggingSourceOperationMask method) and the dragging operations and pasteboard data types the destination itself supports.

If none of the operations is appropriate, this method should return NSDragOperationNone (this is the default response if the method is not implemented by the destination). A destination will still receive draggingUpdated and draggingExited even if NSDragOperationNone is returned by this method.

9.3.17 draggingExited(sender as NSDraggingInfoMBS)

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

Function: Invoked when the dragged image exits the destination's bounds rectangle (in the case of a view object) or its frame rectangle (in the case of a window object).

Notes: sender: The object sending the message; use it to get details about the dragging operation.

9.3.18 `draggingSessionEndedAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS, operation as Integer)`

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

Function: Invoked when the dragging session has completed.

Notes: session: The dragging session.

screenPoint: The point where the drag ended, in screen coordinates.

operation: The drag operation. See constants for drag operation types.

Available in OS X v10.7 and later.

9.3.19 `draggingSessionMovedToPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)`

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

Function: Invoked when the drag moves on the screen.

Notes: session: The dragging session.

screenPoint: The point where the drag moved to, in screen coordinates.

Available in OS X v10.7 and later.

9.3.20 `draggingSessionSourceOperationMaskForDraggingContext(session as NSDraggingSessionMBS, context as Integer) as Integer`

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

Function: Declares the types of operations the source allows to be performed. (required)

Notes: session: The dragging session.

context: The dragging context. See `NSDraggingContext` constants for the supported values.

Return the appropriate dragging operation as defined in constants.

In the future Apple may provide more specific "within" values in the future. To account for this, for unrecognized localities, return the operation mask for the most specific context that you are concerned with.

9.3.21 draggingSessionWillBeginAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)

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

Function: Invoked when the drag will begin.

Notes: session: The dragging session.

screenPoint: The point where the drag will begin, in screen coordinates.

Available in OS X v10.7 and later.

9.3.22 draggingUpdated(sender as NSDraggingInfoMBS) as Integer

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

Function: Invoked periodically as the image is held within the destination area, allowing modification of the dragging operation or mouse-pointer position.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered message.

For this to be invoked, the destination must have registered for the pasteboard data type involved in the drag operation. The messages continue until the image is either released or dragged out of the window or view.

This method provides the destination with an opportunity to modify the dragging operation depending on the position of the mouse pointer inside of the destination view or window object. For example, you may have several graphics or areas of text contained within the same view and wish to tailor the dragging operation, or to ignore the drag event completely, depending upon which object is underneath the mouse pointer at the time when the user releases the dragged image and the performDragOperation method is invoked.

You typically examine the contents of the pasteboard in the draggingEntered method, where this examination is performed only once, rather than in the draggingUpdated method, which is invoked multiple times.

Only one destination at a time receives a sequence of draggingUpdated messages. If the mouse pointer is within the bounds of two overlapping views that are both valid destinations, the uppermost view receives these messages until the image is either released or dragged out.

9.3.23 `endGestureWithEvent(e as NSEventMBS)` as boolean

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

Function: Informs the receiver that the user has ended a touch gesture.

Notes: e: An event object representing the gesture end.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.3.24 `ignoreModifierKeysForDraggingSession(session as NSDraggingSessionMBS)` as boolean

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

Function: Returns whether the modifier keys will be ignored for this dragging session.

Notes: session: The dragging session.

Return true if the modifier keys will be ignored, false otherwise.

Available in OS X v10.7 and later.

9.3.25 `isOpaque` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Whether this view is opaque.

9.3.26 `keyDown(e as NSEventMBS)` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the key events.

Notes: Return true if you handled this event.

Please return true in `becomeFirstResponder` and `acceptsFirstResponder`, so your `nview` can become first responder and receive key events.

9.3.27 `keyUp(e as NSEventMBS)` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the key events.

Notes: Return true if you handled this event.

Please return true in `becomeFirstResponder` and `acceptsFirstResponder`, so your `nview` can become first responder and receive key events.

9.3.28 `magnifyWithEvent(e as NSEventMBS)` as boolean

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

Function: Informs the receiver that the user has begun a pinch gesture.

Notes: `e`: An event object representing the magnify gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.3.29 `menuForEvent(e as NSEventMBS, defaultMenu as NSMenuMBS)` as NSMenuMBS

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Overridden by subclasses to return a context-sensitive pop-up menu for a given mouse-down event.

Notes: `theEvent`: An object representing a mouse-down event.

`defaultMenu`: The menu as constructed by super class.

The receiver can use information in the mouse event, such as its location over a particular element of the receiver, to determine what kind of menu to return. For example, a text object might display a text-editing menu when the cursor lies over text and a menu for changing graphics attributes when the cursor lies over an embedded image.

The default implementation returns the default menu.

9.3.30 `mouseDown(e as NSEventMBS, x as Double, y as Double)` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.31 `mouseDownCanMoveWindow` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: This event is called so you can decide what happens with mouse down.

Notes: Return true if you do not need to handle a mouse down and it can pass through to superviews; False if you need to handle the mouse down.

This allows iApp-type applications to determine the region by which a window can be moved. By default, this method returns false if the view is opaque; otherwise, it returns true. Subclasses can override this method to return a different value.

9.3.32 mouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.33 mouseEntered(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.34 mouseExited(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.35 mouseMoved(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.36 mouseUp(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.37 Open

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: The event called when the custom NSSearchField is created.

9.3.38 otherMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.3.39 otherMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.3.40 otherMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.3.41 performDragOperation(sender as NSDraggingInfoMBS) as boolean

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

Function: Invoked after the released image has been removed from the screen, signaling the receiver to import the pasteboard data.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return if the destination accepts the data, it returns true; otherwise it returns false. The default is to return false.

For this method to be invoked, the previous `prepareForDragOperation` message must have returned true. The destination should implement this method to do the real work of importing the pasteboard data represented by the image.

If the sender object's `animatesToDestination` was set to true in `prepareForDragOperation`, then setup any animation to arrange space for the drag items to animate to. Also at this time, enumerate through the dragging items to set their destination frames and destination images.

9.3.42 `prepareForDragOperation(sender as NSDraggingInfoMBS)` as boolean

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

Function: Invoked when the image is released, allowing the receiver to agree to or refuse drag operation.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns true if the receiver agrees to perform the drag operation and false if not.

This method is invoked only if the most recent `draggingEntered` or `draggingUpdated` message returned an acceptable drag-operation value.

If you want the drag items to animate from their current location on screen to their final location in your view, set the sender object's `animatesToDestination` property to true in your implementation of this method.

9.3.43 `pressureChange(e as NSEventMBS)` as boolean

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

Function: Informs the current object that a pressure change occurred on a system that supports pressure sensitivity.

Notes: This method is invoked automatically in response to user actions. `event` is the event that initiated the change in pressure.

Available in OS X v10.10.3 and later.

9.3.44 `resignFirstResponder` as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Focus is going away.

Notes: Return true to accept.

9.3.45 rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.46 rightMouseDownDragged(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.47 rightMouseDownUp(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.3.48 rotateWithEvent(e as NSEventMBS) as boolean

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

Function: Informs the receiver that the user has begun a rotation gesture.

Notes: e: An event object representing the rotate gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.3.49 scrollWheel(e as NSEventMBS) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Informs the subclass that the mouse's scroll wheel has moved.

Notes: e: An object encapsulating information about the wheel-scrolling event.

The default implementation simply passes this message to the next responder.
Return true to not pass the event.

9.3.50 `swipeWithEvent(e as NSEventMBS)` as boolean

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

Function: Informs the receiver that the user has begun a swipe gesture.

Notes: `e`: An event object representing the swipe gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.3.51 `updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)`

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

Function: Invoked when the dragging images should be changed.

Notes: `sender`: The object sending the message; use this object to get details about the dragging operation.

While a destination may change the dragging images at any time, it is recommended to wait until this method is called before updating the dragging images.

This allows the system to delay changing the dragging images until it is likely that the user will drop on this destination. Otherwise, the dragging images will change too often during the drag which would be distracting to the user.

9.3.52 `viewDidMoveToWindow`

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Informs the receiver that it has been added to a new view hierarchy.

Notes: The default implementation does nothing; subclasses can implement this event to perform whatever actions are necessary.

Window may return nil when this method is invoked, indicating that the receiver does not currently reside in any window. This occurs when the receiver has just been removed from its superview or when the receiver has just been added to a superview that does not itself have a window. Overrides of this method may choose to ignore such cases if they are not of interest.

9.3.53 wantsPeriodicDraggingUpdates as boolean

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

Function: Asks the destination object whether it wants to receive periodic draggingUpdated messages.

Notes: Return true if the destination wants to receive periodic draggingUpdated messages, false otherwise.

If the destination returns false, these messages are sent only when the mouse moves or a modifier flag changes. Otherwise the destination gets the default behavior, where it receives periodic dragging-updated messages even if nothing changes.

9.4 class CustomNSTextViewMBS

9.4.1 class CustomNSTextViewMBS

Plugin Version: 17.2, Platform: macOS, Targets: Desktop only.

Function: The class for a custom text view.

Notes: Some events will not fire as text view eats them itself.

Subclass of the NSTextViewMBS class.

Blog Entries

- [MBS Xojo Plugins, version 17.2pr1](#)

9.4.2 Methods

9.4.3 Constructor

Plugin Version: 17.2, Platform: macOS, Targets: Desktop only.

Function: The dummy constructor doing nothing.

See also:

- 9.4.4 Constructor(Handle as Integer) 484
- 9.4.5 Constructor(left as Double, top as Double, width as Double, height as Double) 484

9.4.4 Constructor(Handle as Integer)

Plugin Version: 17.2, Platform: macOS, Targets: Desktop only.

Function: The constructor.

See also:

- 9.4.3 Constructor 484
- 9.4.5 Constructor(left as Double, top as Double, width as Double, height as Double) 484

9.4.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 17.2, Platform: macOS, Targets: Desktop only.

Function: The constructor for a new custom NSView object.

See also:

- 9.4.3 Constructor 484

- 9.4.4 Constructor(Handle as Integer)

9.4.6 Destructor

Plugin Version: 17.2, Platform: macOS, Targets: Desktop only.

Function: The destructor.

9.4.7 Events

9.4.8 acceptsFirstMouse(e as NSEventMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Overridden by subclasses to return true if the receiver should be sent a mouseDown event for an initial mouse-down event, false if not.

Notes: The receiver can either return a value unconditionally or use the location of event e to determine whether or not it wants the event. The default implementation ignores the event and returns false.

Implement this event in a subclass to allow instances to respond to click-through. This allows the user to click on a view in an inactive window, activating the view with one click, instead of clicking first to make the window active and then clicking the view. Most view objects refuse a click-through attempt, so the event simply activates the window. Many control objects, however, such as instances of NSButton and NSSlider, do accept them, so the user can immediately manipulate the control without having to release the mouse button.

9.4.9 acceptsFirstResponder as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Whether to accept first responder.

Notes: Return true if your control can have the focus and false if not.

9.4.10 becomeFirstResponder as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Called when the object gets focus.

Notes: Return true to accept.

9.4.11 `beginGestureWithEvent(e as NSEventMBS)` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the receiver that the user has begun a touch gesture.

Notes: `e`: An event object representing the gesture beginning.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.4.12 `canBecomeKeyView` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Returns whether the receiver can become key view.

Notes: Returns true if the receiver can become key view, false otherwise.

9.4.13 `Close`

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: The event called when the custom view is destroyed.

9.4.14 `concludeDragOperation(sender as NSDraggingInfoMBS)`

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the dragging operation is complete, signaling the receiver to perform any necessary clean-up.

Notes: `sender`: The object sending the message; use it to get details about the dragging operation.

For this method to be invoked, the previous `performDragOperation` must have returned true.

The destination implements this method to perform any tidying up that it needs to do, such as updating its visual representation now that it has incorporated the dragged data. This message is the last message sent from sender to the destination during a dragging session.

If the sender object's `animatesToDestination` property was set to true in `prepareForDragOperation`, then the drag image is still visible. At this point you should draw the final visual representation in the view. When this method returns, the drag image is removed from the screen. If your final visual representation matches the visual representation in the drag, this is a seamless transition.

9.4.15 draggingEnded(sender as NSDraggingInfoMBS)

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Implement this event to be notified when a drag operation ends in some other destination.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

This method might be used by a destination doing auto-expansion in order to collapse any auto-expands.

9.4.16 draggingEntered(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the dragged image enters destination bounds or frame; delegate returns dragging operation to perform.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered message.

Invoked when a dragged image enters the destination but only if the destination has registered for the pasteboard data type involved in the drag operation. Specifically, this method is invoked when the mouse pointer enters the destination's bounds rectangle (if it is a view object) or its frame rectangle (if it is a window object).

This method must return a value that indicates which dragging operation the destination will perform when the image is released. In deciding which dragging operation to return, the method should evaluate the overlap between both the dragging operations allowed by the source (obtained from sender with the draggingSourceOperationMask method) and the dragging operations and pasteboard data types the destination itself supports.

If none of the operations is appropriate, this method should return NSDragOperationNone (this is the default response if the method is not implemented by the destination). A destination will still receive draggingUpdated and draggingExited even if NSDragOperationNone is returned by this method.

9.4.17 draggingExited(sender as NSDraggingInfoMBS)

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the dragged image exits the destination’s bounds rectangle (in the case of a view object) or its frame rectangle (in the case of a window object).

Notes: sender: The object sending the message; use it to get details about the dragging operation.

9.4.18 `draggingSessionEndedAtPoint`(session as `NSDraggingSessionMBS`, screenPoint as `NSPointMBS`, operation as `Integer`)

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the dragging session has completed.

Notes: session: The dragging session.

screenPoint: The point where the drag ended, in screen coordinates.

operation: The drag operation. See constants for drag operation types.

Available in OS X v10.7 and later.

9.4.19 `draggingSessionMovedToPoint`(session as `NSDraggingSessionMBS`, screenPoint as `NSPointMBS`)

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the drag moves on the screen.

Notes: session: The dragging session.

screenPoint: The point where the drag moved to, in screen coordinates.

Available in OS X v10.7 and later.

9.4.20 `draggingSessionSourceOperationMaskForDraggingContext`(session as `NSDraggingSessionMBS`, context as `Integer`) as `Integer`

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Declares the types of operations the source allows to be performed. (required)

Notes: session: The dragging session.

context: The dragging context. See `NSDraggingContext` constants for the supported values.

Return the appropriate dragging operation as defined in constants.

In the future Apple may provide more specific "within" values in the future. To account for this, for unrecognized localities, return the operation mask for the most specific context that you are concerned with.

9.4.21 draggingSessionWillBeginAtPoint(session as NSDraggingSessionMBS, screenPoint as NSPointMBS)

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the drag will begin.

Notes: session: The dragging session.

screenPoint: The point where the drag will begin, in screen coordinates.

Available in OS X v10.7 and later.

9.4.22 draggingUpdated(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked periodically as the image is held within the destination area, allowing modification of the dragging operation or mouse-pointer position.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered message.

For this to be invoked, the destination must have registered for the pasteboard data type involved in the drag operation. The messages continue until the image is either released or dragged out of the window or view.

This method provides the destination with an opportunity to modify the dragging operation depending on the position of the mouse pointer inside of the destination view or window object. For example, you may have several graphics or areas of text contained within the same view and wish to tailor the dragging operation, or to ignore the drag event completely, depending upon which object is underneath the mouse pointer at the time when the user releases the dragged image and the performDragOperation method is invoked.

You typically examine the contents of the pasteboard in the draggingEntered method, where this examination is performed only once, rather than in the draggingUpdated method, which is invoked multiple times.

Only one destination at a time receives a sequence of draggingUpdated messages. If the mouse pointer is within the bounds of two overlapping views that are both valid destinations, the uppermost view receives these messages until the image is either released or dragged out.

9.4.23 endGestureWithEvent(e as NSEventMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the receiver that the user has ended a touch gesture.

Notes: e: An event object representing the gesture end.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.4.24 ignoreModifierKeysForDraggingSession(session as NSDraggingSessionMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Returns whether the modifier keys will be ignored for this dragging session.

Notes: session: The dragging session.

Return true if the modifier keys will be ignored, false otherwise.

Available in OS X v10.7 and later.

9.4.25 isOpaque as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Whether this view is opaque.

9.4.26 keyDown(e as NSEventMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the key events.

Notes: Return true if you handled this event.

Please return true in `becomeFirstResponder` and `acceptsFirstResponder`, so your `nview` can become first responder and receive key events.

9.4.27 keyUp(e as NSEventMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the key events.

Notes: Return true if you handled this event.

Please return true in `becomeFirstResponder` and `acceptsFirstResponder`, so your `nview` can become first responder and receive key events.

9.4.28 `magnifyWithEvent(e as NSEventMBS)` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the receiver that the user has begun a pinch gesture.

Notes: `e`: An event object representing the magnify gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.4.29 `menuForEvent(e as NSEventMBS, defaultMenu as NSMenuMBS)` as NSMenuMBS

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Overridden by subclasses to return a context-sensitive pop-up menu for a given mouse-down event.

Notes: `theEvent`: An object representing a mouse-down event.

`defaultMenu`: The menu as constructed by super class.

The receiver can use information in the mouse event, such as its location over a particular element of the receiver, to determine what kind of menu to return. For example, a text object might display a text-editing menu when the cursor lies over text and a menu for changing graphics attributes when the cursor lies over an embedded image.

The default implementation returns the default menu.

9.4.30 `mouseDown(e as NSEventMBS, x as Double, y as Double)` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.31 `mouseDownCanMoveWindow` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: This event is called so you can decide what happens with mouse down.

Notes: Return true if you do not need to handle a mouse down and it can pass through to superviews; False if you need to handle the mouse down.

This allows iApp-type applications to determine the region by which a window can be moved. By default, this method returns false if the view is opaque; otherwise, it returns true. Subclasses can override this method to return a different value.

9.4.32 mouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.33 mouseEntered(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.34 mouseExited(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.35 mouseMoved(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.36 mouseUp(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.37 Open

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: The event called when the custom NSSearchField is created.

9.4.38 otherMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.4.39 otherMouseDragged(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.4.40 otherMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

Third mouse button.

9.4.41 performDragOperation(sender as NSDraggingInfoMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked after the released image has been removed from the screen, signaling the receiver to import the pasteboard data.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return if the destination accepts the data, it returns true; otherwise it returns false. The default is to return false.

For this method to be invoked, the previous `prepareForDragOperation` message must have returned true. The destination should implement this method to do the real work of importing the pasteboard data represented by the image.

If the sender object's `animatesToDestination` was set to true in `prepareForDragOperation`, then setup any animation to arrange space for the drag items to animate to. Also at this time, enumerate through the dragging items to set their destination frames and destination images.

9.4.42 `prepareForDragOperation(sender as NSDraggingInfoMBS)` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the image is released, allowing the receiver to agree to or refuse drag operation.

Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns true if the receiver agrees to perform the drag operation and false if not.

This method is invoked only if the most recent `draggingEntered` or `draggingUpdated` message returned an acceptable drag-operation value.

If you want the drag items to animate from their current location on screen to their final location in your view, set the sender object's `animatesToDestination` property to true in your implementation of this method.

9.4.43 `pressureChange(e as NSEventMBS)` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the current object that a pressure change occurred on a system that supports pressure sensitivity.

Notes: This method is invoked automatically in response to user actions. `event` is the event that initiated the change in pressure.

Available in OS X v10.10.3 and later.

9.4.44 `resignFirstResponder` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Focus is going away.

Notes: Return true to accept.

9.4.45 rightMouseDown(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.46 rightMouseDownDragged(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.47 rightMouseUp(e as NSEventMBS, x as Double, y as Double) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: One of the mouse events.

Notes: Return true if you handled this event.

9.4.48 rotateWithEvent(e as NSEventMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the receiver that the user has begun a rotation gesture.

Notes: e: An event object representing the rotate gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.4.49 scrollWheel(e as NSEventMBS) as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the subclass that the mouse's scroll wheel has moved.

Notes: e: An object encapsulating information about the wheel-scrolling event.

The default implementation simply passes this message to the next responder.
Return true to not pass the event.

9.4.50 `swipeWithEvent(e as NSEventMBS)` as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the receiver that the user has begun a swipe gesture.

Notes: `e`: An event object representing the swipe gesture.

The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

Return true if you handled this event.

9.4.51 `updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)`

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Invoked when the dragging images should be changed.

Notes: `sender`: The object sending the message; use this object to get details about the dragging operation.

While a destination may change the dragging images at any time, it is recommended to wait until this method is called before updating the dragging images.

This allows the system to delay changing the dragging images until it is likely that the user will drop on this destination. Otherwise, the dragging images will change too often during the drag which would be distracting to the user.

9.4.52 `viewDidMoveToWindow`

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Informs the receiver that it has been added to a new view hierarchy.

Notes: The default implementation does nothing; subclasses can implement this event to perform whatever actions are necessary.

Window may return nil when this method is invoked, indicating that the receiver does not currently reside in any window. This occurs when the receiver has just been removed from its superview or when the receiver has just been added to a superview that does not itself have a window. Overrides of this method may choose to ignore such cases if they are not of interest.

9.4.53 wantsPeriodicDraggingUpdates as boolean

Plugin Version: 17.2, Platform: macOS, Targets: .

Function: Asks the destination object whether it wants to receive periodic draggingUpdated messages.

Notes: Return true if the destination wants to receive periodic draggingUpdated messages, false otherwise.

If the destination returns false, these messages are sent only when the mouse moves or a modifier flag changes. Otherwise the destination gets the default behavior, where it receives periodic dragging-updated messages even if nothing changes.

9.5 control DesktopNSPathControlControlMBS

9.5.1 control DesktopNSPathControlControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The Xojo control to host a NSPathControl.

Notes: We know the fact, that the control wrapping the segmented control has control twice in the name. Suggestions to rename are welcome, but unlikely to happen as it would break code.

Blog Entries

- [News from the MBS Xojo Plugins Version 24.0](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 24.0](#)
- [NSSegmentedControl and NSPathControl](#)
- [MBS Xojo Plugins, version 24.0pr7](#)

Xojo Developer Magazine

- [22.2, page 9: News](#)

9.5.2 Properties

9.5.3 View as NSPathControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The reference to the underlying NSPathControlMBS.

Notes: (Read only property)

9.5.4 Events

9.5.5 Action

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event is called when user clicks on a date/time and changes something.

9.5.6 BoundsChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.5.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.5.8 DoubleClick

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Called when a double click happened.

Notes: May call Action event before.

9.5.9 FocusLost

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control lost focus.

In older Xojo versions, this event is named LostFocus.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.5.10 FocusReceived

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control itself got focus.

In older Xojo versions, this event is named GotFocus.

Notes:

This only fires if the control itself got focus and not a sub control.

9.5.11 FrameChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.5.12 MenuBarSelected

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In older Xojo versions, this event is named EnableMenuItems.

9.5.13MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ãs region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.5.14 MouseDrag(x as Integer, y as Integer)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.5.15 MouseUp(x As Integer, y As Integer)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.5.16 ScaleFactorChanged(NewFactor as double)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.5.17 willDisplayOpenPanel(openPanel as Variant)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Implement this method to customize the Open panel shown by a pop-up-style path.

Notes: This method is called before the Open panel is shown but after its allowed file types are set to the cell's allowed types. At this time, you can further customize the Open panel as required. This method is called only when the style is set to NSPathStylePopUp. Implementation of this method is optional.

openPanel is a NSOpenPanelMBS object.

9.5.18 willPopUpMenu(menu as Variant)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Implement this method to customize the menu of a pop-up-style path.

Notes: This event is called before the pop-up menu is shown. At this time, you can further customize the menu as required, adding and removing items. This method is called only when the style is set to NSPathStylePopUp. Implementation of this method is optional.

Menu is a NSMenuMBS object.

9.5.19 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.6 control DesktopNSSearchFieldControlMBS

9.6.1 control DesktopNSSearchFieldControlMBS

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

Function: The Xojo control for a NSSearchField.

Notes: This control embeds a special NSSearchField subclass.

Designed for Xojo 2021r3 and newer.

Please use view property to access the underlying object and set properties.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 24.0](#)
- [MBS Xojo Plugins, version 23.6pr1](#)
- [MBS Xojo Plugins, version 23.2pr1](#)
- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo Plugins, version 21.5pr1](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr8](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr2](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr10](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr7](#)

9.6.2 Properties

9.6.3 View as NSSearchFieldMBS

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

Function: The view used in the control.

Notes: Use this object to set more options on the control.

(Read only property)

9.6.4 Events

9.6.5 Action

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The action event run, when e.g. return key is pressed.

9.6.6 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.6.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.6.8 FocusLost

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control lost focus.

In older Xojo versions, this event is named LostFocus.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.6.9 FocusReceived

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control itself got focus.

In older Xojo versions, this event is named GotFocus.

Notes:

This only fires if the control itself got focus and not a sub control.

9.6.10 FrameChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.6.11 MenuBarSelected

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In older Xojo versions, this event is named EnableMenuItems.

9.6.12MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ã region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.6.13 MouseDrag(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.6.14 MouseUp(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.6.15 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.6.16 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when a control with editable text begins an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidBeginEditingNotification`.

This event is invoked when the user begins editing text in a control such as a text field or a form field. The control posts a `NSControlTextDidBeginEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also delivered for inspection.

9.6.17 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when the text in the receiving control changes.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidChangeNotification`.

This event is invoked when text in a control such as a text field or form changes. The control posts a `NSControlTextDidChangeNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is provided as parameter for inspection.

9.6.18 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when a control with editable text ends an editing session.

Notes: Notification: The notification object. The name of the notification is always NSControlTextDidEndEditingNotification.

This event is invoked when the user stops editing text in a control such as a text field or form. The control posts a NSControlTextDidEndEditingNotification notification, and if the control' subclass implements this event, it is automatically registered to receive the notification. The field editor is also provided for inspection.

9.6.19 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called to decide whether text editing should be allowed.

Notes: Return true to allow text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.6.20 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called to decide whether ending text editing should be allowed.

Notes: Return true to allow end of text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.6.21 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.7 control DesktopNSSecureTextFieldControlMBS

9.7.1 control DesktopNSSecureTextFieldControlMBS

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

Function: The Xojo control for a NSSecureTextField.

Notes: This control embeds a special NSSecureTextField subclass.

Designed for Xojo 2021r3 and newer.

Please use view property to access the underlying object and set properties.

Blog Entries

- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr8](#)

9.7.2 Properties

9.7.3 echosBullets as Boolean

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

Function: Whether the receiver echoes a bullet character rather than each character typed.

Example:

```
dim t as DesktopNSSecureTextFieldControlMBS // your textfield
t.echosBullets = true
```

Notes: If true, bullets are echoed. If false, the cursor is moved for each character typed, but nothing is displayed.

(Read and Write property)

9.7.4 View as NSSecureTextFieldMBS

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

Function: The view used in the control.

Notes: Use this object to set more options on the control.

(Read only property)

9.7.5 Events

9.7.6 Action

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The control's action was triggered.

Notes: The text changed.

9.7.7 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.7.8 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.7.9 FocusLost

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control lost focus.

In older Xojo versions, this event is named LostFocus.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.7.10 FocusReceived

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control itself got focus.

In older Xojo versions, this event is named GotFocus.

Notes:

This only fires if the control itself got focus and not a sub control.

9.7.11 FrameChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.7.12 MenuBarSelected

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In older Xojo versions, this event is named EnableMenuItems.

9.7.13MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control's region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.7.14 MouseDrag(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the

mouse has really moved.

9.7.15 MouseUp(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.7.16 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.7.17 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when a control with editable text begins an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidBeginEditingNotification`.

This event is invoked when the user begins editing text in a control such as a text field or a form field. The control posts a `NSControlTextDidBeginEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also delivered for inspection.

9.7.18 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when the text in the receiving control changes.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidChangeNotification`.

This event is invoked when text in a control such as a text field or form changes. The control posts a `NSControlTextDidChangeNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is provided as parameter for inspection.

9.7.19 `TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when a control with editable text ends an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidEndEditingNotification`.

This event is invoked when the user stops editing text in a control such as a text field or form. The control posts a `NSControlTextDidEndEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also provided for inspection.

9.7.20 `textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called to decide whether text editing should be allowed.

Notes: Return true to allow text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.7.21 `textShouldEndEditing(fieldEditor as NSTextMBS) as boolean`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called to decide whether ending text editing should be allowed.

Notes: Return true to allow end of text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.7.22 `willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)`

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.8 control DesktopNSSegmentedControlControlMBS

9.8.1 control DesktopNSSegmentedControlControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The Xojo control to host a NSSegmentedControl.

Notes: We know the fact, that the control wrapping the segmented control has control twice in the name. Suggestions to rename are welcome, but unlikely to happen as it would break code.

Blog Entries

- [News from the MBS Xojo Plugins Version 24.0](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 24.0](#)
- [NSSegmentedControl and NSPathControl](#)
- [MBS Xojo Plugins, version 24.0pr7](#)

9.8.2 Properties

9.8.3 View as NSSegmentedControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The reference to the underlying NSSegmentedControlMBS.

Notes: (Read only property)

9.8.4 Events

9.8.5 Action

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event is called when user clicks on a date/time and changes something.

9.8.6 BoundsChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.8.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.8.8 FocusLost

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control lost focus.

In older Xojo versions, this event is named LostFocus.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.8.9 FocusReceived

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control itself got focus.

In older Xojo versions, this event is named GotFocus.

Notes:

This only fires if the control itself got focus and not a sub control.

9.8.10 FrameChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.8.11 MenuBarSelected

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In older Xojo versions, this event is named `EnableMenuItems`.

9.8.12 `MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ãs region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the `MouseDown`. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the `MouseDown` and `MouseUp` events.

If you return False, the system handles the `MouseDown` so the above event handlers do not get called.

9.8.13 `MouseDown(x as Integer, y as Integer)`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of times per second), it is your responsibility to determine if the mouse has really moved.

9.8.14 `MouseUp(x As Integer, y As Integer)`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.8.15 `ScaleFactorChanged(NewFactor as double)`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.8.16 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.9 control DesktopNSSplitViewControlMBS

9.9.1 control DesktopNSSplitViewControlMBS

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Our control to wrap a NSSplitViewMBS object.

Blog Entries

- [MBS Xojo Plugins, version 23.5pr7](#)
- [News from the MBS Xojo Plugins Version 23.4](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.4](#)
- [NSSplitView Control for Xojo](#)
- [MBS Xojo Plugins, version 23.4pr5](#)

9.9.2 Properties

9.9.3 View as NSSplitViewMBS

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: The reference to the underlying NSSplitView.

Notes: (Read only property)

9.9.4 Events

9.9.5 BoundsChanged

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.9.6 canCollapseSubview(subview as NSViewMBS) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to determine whether the user can collapse and expand the specified subview.

Notes: subview: The subview to collapse.

Returns true if subview collapses when the user drags a divider beyond the halfway mark between its minimum size and its edge; otherwise, false.

The subview expands when the user drags the divider beyond the halfway mark between its minimum size and its edge.

To specify the minimum size, define the `constrainMaxCoordinate` and `constrainMinCoordinate` events. A subview can collapse only if you also define `constrainMinCoordinate`.

A collapsed subview isn't visible, but the split view object retains it with the same size as before the collapse.

If the events are implemented, the subviews can't collapse.

9.9.7 `constrainMaxCoordinate(proposedMaximumPosition as double, dividerIndex as Integer) as Double`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to constrain the maximum coordinate limit of a divider when the user drags it.

Notes: `proposedMax`: The proposed maximum coordinate limit of the subview in the split view, flipped coordinate system.

`dividerIndex`: Specifies the divider the user is moving, with the first divider being 0 and increasing from top to bottom (or left to right).

Returns the maximum coordinate limit of the divider.

The delegate receives this message before the split view begins tracking the mouse to position a divider. You can further constrain the limits, but you can't extend the divider limits.

If the split bars are horizontal (views are one on top of the other), `proposedMax` is the bottom limit. If the split bars are vertical (views are side by side), `proposedMax` is the right limit. The initial value of `proposedMax` is the bottom (or right side) of the subview after the divider.

9.9.8 `constrainMinCoordinate(proposedMinimumPosition as double, dividerIndex as Integer) as Double`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to constrain the minimum coordinate limit of a divider when the user drags it.

Notes: `proposedMin`: The proposed minimum coordinate limit of the subview in the split view, flipped coordinate system.

`dividerIndex`: Specifies the divider the user is moving, with the first divider being 0 and increasing from top

to bottom (or left to right).

Returns the minimum coordinate limit of the divider.

The delegate receives this message before the split view begins tracking the cursor to position a divider. You can further constrain the limits, but you can,Äô extend the divider limits.

If the split bars are horizontal (views are one on top of the other), proposedMin is the top limit. If the split bars are vertical (views are side by side), proposedMin is the left limit. The initial value of proposedMin is the top (or left side) of the subview before the divider.

9.9.9 constrainSplitPosition(proposedPosition as double, dividerIndex as Integer) as Double

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to constrain the divider to certain positions.

Notes: proposedPosition: The cursor,Äôs current position, and the proposed position of the divider.

dividerIndex: The index of the divider the user is moving, with the first divider being 0 and increasing from top to bottom (or left to right).

Returns the position for constraining the divider.

If the delegate implements this method, the split view calls it repeatedly as the user moves the divider.

If a subview,Äôs height must be a multiple of a certain number, use this method to return the multiple nearest to proposedPosition.

9.9.10 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.9.11 drawDivider(graphics as NSGraphicsMBS, Rect as NSRectMBS) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Draws a divider between two of the split view,Äôs subviews.

Example:

[Function](#) drawDivider(graphics as NSGraphicsMBS, Rect as NSRectMBS) [Handles](#) drawDivider as Boolean

```
// draw a red box with blue border
graphics.setFillColor NSColorMBS.redColor
graphics.setStrokeColor NSColorMBS.blueColor

Dim r As New NSRectMBS(rect.X, rect.y + rect.Height/2 - 5, 10, 10)
graphics.fillRect r
graphics.strokeRect r

Return True
End Function
```

Notes: Rect: The entire divider rectangle in the split view,Âs flipped coordinate system.

If you override this method to use a custom style for the divider, you may need to change the size of the divider.

9.9.12 FocusLost

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The control lost focus.

In older Xojo versions, this event is named LostFocus.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.9.13 FocusReceived

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The control itself got focus.

In older Xojo versions, this event is named GotFocus.

Notes:

This only fires if the control itself got focus and not a sub control.

9.9.14 FrameChanged

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.9.15 MenuBarSelected

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In older Xojo versions, this event is named EnableMenuItems.

9.9.16MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ãs region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.9.17 MouseDrag(x as Integer, y as Integer)

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.9.18 MouseUp(x As Integer, y As Integer)

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.9.19 resizeModeSubviewsWithOldSize(oldSize as NSSizeMBS)

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to specify custom sizing behavior for the subviews of the split view.

Notes: oldSize: The size of the split view before the user resizes it.

If the delegate implements this method, it receives this message after the split view resizes.

Resize the subviews so that the sum of the sizes of the subviews plus the sum of the thickness of the dividers equals the size of the new frame of the NSSplitView. You can get the thickness of a divider through the dividerThickness method.

If you implement this delegate method to resize subviews on your own, the NSSplitViewMBS doesn't perform any error checking for you. However, you can invoke adjustSubviews() to perform the default sizing behavior.

9.9.20 ScaleFactorChanged(NewFactor as double)

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.9.21 shouldAdjustSizeOfSubview(view as NSViewMBS) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to specify whether to resize the subview.

Notes: subview: The subview to resize.

If adjustSubviews() can change the size of the subview, return true; otherwise, false. By returning false, you lock the size of the split view subview while the split view resizes.

Regardless of the value that this method returns, `adjustSubviews()` may change the origin of the subview. Nonresizable subviews may resize to prevent an invalid subview layout. If a split view has no delegate, or if its delegate doesn't respond to this message, the split view behaves as if this method returns true.

9.9.22 `shouldHideDivider(dividerIndex as Integer) as Boolean`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to determine whether the user can drag a divider or adjust it off the edge of the split view.

Notes: `dividerIndex`: The zero-based index of the divider.

Return true if the user can drag the divider off the edge of the split view, resulting in it not being visible.

If a split view has no delegate, or if its delegate doesn't respond to this message, the split view behaves as if it has a delegate that returns false when it receives this message.

9.9.23 `splitViewDidResizeSubviews`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Notifies the delegate when the split view resizes its subviews.

Notes: The control invokes this event after the split view resizes two of its subviews in response to the repositioning of a divider.

9.9.24 `splitViewWillResizeSubviews`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Notifies the delegate when the split view is about to resize its subviews.

Notes: The control invokes this method before the split view resizes two of its subviews in response to the repositioning of a divider.

9.9.25 `willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)`

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.10 control DesktopNSTextFieldControlMBS

9.10.1 control DesktopNSTextFieldControlMBS

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

Function: The Xojo control for a NSTextField.

Notes: This control embeds a special NSTextField subclass.

Designed for Xojo 2021r3 and newer.

Please use view property to access the underlying object and set properties.

Blog Entries

- [MBS Xojo Plugins, version 23.2pr1](#)
- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr2](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr7](#)

9.10.2 Properties

9.10.3 View as NSTextFieldMBS

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

Function: The view used in the control.

Notes: Use this object to set more options on the control.

(Read only property)

9.10.4 Events

9.10.5 Action

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The control's action was triggered.

Notes: The text changed.

9.10.6 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.10.7 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.10.8 FocusLost

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control lost focus.

In older Xojo versions, this event is named LostFocus.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.10.9 FocusReceived

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control itself got focus.

In older Xojo versions, this event is named GotFocus.

Notes:

This only fires if the control itself got focus and not a sub control.

9.10.10 FrameChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.10.11 MenuBarSelected

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In older Xojo versions, this event is named EnableMenuItems.

9.10.12MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ã region at the location passed in to x, y.
Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.
Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.10.13 MouseDrag(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.
Notes: Mouse location is local to the control passed in to x, y.
As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.10.14 MouseUp(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was released.
Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.10.15 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.10.16 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when a control with editable text begins an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidBeginEditingNotification`.

This event is invoked when the user begins editing text in a control such as a text field or a form field. The control posts a `NSControlTextDidBeginEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also delivered for inspection.

9.10.17 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when the text in the receiving control changes.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidChangeNotification`.

This event is invoked when text in a control such as a text field or form changes. The control posts a `NSControlTextDidChangeNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is provided as parameter for inspection.

9.10.18 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when a control with editable text ends an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidEndEditingNotification`.

dEndEditingNotification.

This event is invoked when the user stops editing text in a control such as a text field or form. The control posts a `NSControlTextDidEndEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also provided for inspection.

9.10.19 `textShouldBeginEditing(fieldEditor as NSTextMBS)` as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called to decide whether text editing should be allowed.

Notes: Return true to allow text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.10.20 `textShouldEndEditing(fieldEditor as NSTextMBS)` as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called to decide whether ending text editing should be allowed.

Notes: Return true to allow end of text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.10.21 `willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)`

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.11 control DesktopNSTextViewControlMBS

9.11.1 control DesktopNSTextViewControlMBS

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

Function: The Xojo control for a NSTextView.

Notes: This control embeds a special NSTextView subclass.

Designed for Xojo 2021r3 and newer.

Please use view property to access the underlying object and set properties.

Blog Entries

- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo / Real Studio Plugins, version 16.5pr6](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr8](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr1](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr7](#)

9.11.2 Properties

9.11.3 AcceptTabs as Boolean

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

Function: Whether the control should accept tab keys.

Notes: If true, the plugin will not forward the tab keydown/keyup events to Xojo, because Xojo would do switch to next control.

(Read and Write property)

9.11.4 Scrollview as Variant

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

Function: The scrollview for this textview.

Notes: (Read only property)

9.11.5 View as NSTextViewMBS

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

Function: The view used in the control.

Notes: Use this object to set more options on the control.
(Read only property)

9.11.6 Events

9.11.7 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.11.8 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.11.9 FocusLost

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control lost focus.

In older Xojo versions, this event is named LostFocus.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.11.10 FocusReceived

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control itself got focus.

In older Xojo versions, this event is named GotFocus.

Notes:

This only fires if the control itself got focus and not a sub control.

9.11.11 FrameChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.11.12 MenuBarSelected

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In older Xojo versions, this event is named EnableMenuItems.

9.11.13MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ãs region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.11.14 MouseDrag(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.11.15 `MouseUp(x as Integer, y as Integer)`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the `x` and `y` parameters to determine if the mouse button was released within the control's boundaries.

9.11.16 `ScaleFactorChanged(NewFactor as Double)`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.11.17 `shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string) as boolean`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when a text view needs to determine if text in a specified range should be changed.

Notes: `affectedCharRange`: The range of characters to be replaced.

`replacementString`: The characters that will replace the characters in `affectedCharRange`; nil if only text attributes are being changed.

Return true to allow the replacement, or false to reject the change.

9.11.18 `textDidBeginEditing`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Informs you that the text object has begun editing (that the user has begun changing it).

9.11.19 `textDidChange`

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Informs you that the text object has changed its characters or formatting attributes.

9.11.20 textDidEndEditing

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Informs you that the text object has finished editing (that it has resigned first responder status).

9.11.21 textShouldBeginEditing as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when a text object begins to change its text, this method requests permission to begin editing.

Notes: If the delegate returns false, the text object proceeds to make changes. If the delegate returns true, the text object abandons the editing operation. This method is also invoked when the user drags and drops a file onto the text object.

9.11.22 textShouldEndEditing as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked from a text object's implementation of resignFirstResponder, this method requests permission to end editing.

Notes: If the delegate returns false, the text object proceeds to finish editing and resign first responder status. If the delegate returns true, the text object selects all of its text and remains the first responder.

9.11.23 textViewDidChangeSelection

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Sent when the selection changes in the text view.

9.11.24 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.12 class DesktopSearchField

9.12.1 class DesktopSearchField

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

Function: The built-in search field control in Xojo.

Notes: Only for Xojo 2020r2 or newer.

9.12.2 Methods

9.12.3 NSSearchFieldMBS as NSSearchFieldMBS

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

Function: Queries search field object for the control.

Notes: Only for Xojo 2021r3 or newer.

9.13 class DesktopSegmentedButton

9.13.1 class DesktopSegmentedButton

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

Function: Build in Segmented Control class in Xojo.

9.13.2 Methods

9.13.3 NSSegmentedControlMBS as NSSegmentedControlMBS

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

Function: Creates a NSSegmentedControlMBS object for the given control.

Example:

```
SegmentedControl1.NSSegmentedControlMBS.selectedSegment = 0
```

Notes: This way you can manipulate Cocoa controls directly.

9.14 class DesktopTabPanel

9.14.1 class DesktopTabPanel

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

Function: Extends the TabPanel control inside Xojo.

9.14.2 Methods

9.14.3 NSTableViewMBS as NSTableViewMBS

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

Function: Creates a NSTableViewMBS object for the given control.

Example:

```
MsgBox TabPanel1.NSTableViewMBS.className
```

Notes: This way you can manipulate Cocoa controls directly.

9.15 class DesktopTextArea

9.15.1 class DesktopTextArea

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

Function: The built in textarea class in Xojo.

Example:

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

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

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

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

if f<>Nil then

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

  b.Write s

end if
```

Notes: Requires Xojo 2021r3 or newer.

9.15.2 Methods

9.15.3 NSTextFieldMBS as NSTextFieldMBS

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

Function: Creates a NSTextFieldMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

Xojo uses a NSTextField for text areas without style and without multiline.

9.15.4 NSTextViewMBS as NSTextViewMBS

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

Function: Creates a NSTextViewMBS object for the given control.

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: This way you can manipulate Cocoa controls directly.

Xojo uses a NSTextField for text areas without style and without multiline.

9.15.5 Properties

9.15.6 RTFDataMBS as Memoryblock

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

Function: Get or set the textview content as RTF data.

Example:

```
dim rtf as MemoryBlock = TextArea1.RTFDataMBS
TextArea2.RTFDataMBS = rtf
```

Notes: Works only for Cocoa and uses RTF parser/generator from Apple.

See also RTFDataMBS in StyledText, WinRTFDataMBS in TextArea and DesktopTextArea controls and RTFData in NSTextViewMBS class.

(Read and Write computed property)

9.16 class NSActionCellMBS

9.16.1 class NSActionCellMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: An NSActionCell defines an active area inside a control (an instance of NSControl or one of its subclasses).

Notes: As an NSControl's active area, an NSActionCell does three things: it usually performs display of text or an icon; it provides the NSControl with a target and an action; and it handles mouse (cursor) tracking by properly highlighting its area and sending action messages to its target based on cursor movement.

Subclass of the NSCellMBS class.

Blog Entries

- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)

Xojo Developer Magazine

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

9.16.2 Methods

9.16.3 Constructor(image as NSImageMBS)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with an image.

Example:

```
dim pic as Picture = LogoMBS(500)
dim n as NSImageMBS = new NSImageMBS(pic)
dim c as new NSActionCellMBS(n)
```

```
Backdrop = c.image.CopyPictureWithMask
```

```
Title = c.classPath
```

See also:

- [9.16.4 Constructor\(text as string\)](#)

9.16.4 Constructor(text as string)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with a text.

Example:

```
dim c as new NSActionCellMBS("Hello")
MsgBox c.StringValue
```

See also:

- 9.16.3 Constructor(image as NSImageMBS)

9.17 class NSButtonCellMBS

9.17.1 class NSButtonCellMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The NSButtonCell class is a subclass of NSActionCell used to implement the user interfaces of push buttons, checkboxes (switches), and radio buttons.

Notes: It can also be used for any other region of a view that's designed to send a message to a target when clicked. The NSButton subclass of NSControl uses a single NSButtonCell.

The NSButtonCell class implements the user interface of NSButton.

Setting the integer, float, double, or object value of an NSButtonCell object results in a call to setState with the value converted to integer. In the case of setObjectValue, nil is equivalent to 0, and a non-nil object that doesn't respond to intValue sets the state to 1. Otherwise, the state is set to the object's intValue. Similarly, querying the integer, float, double, or object value of an NSButtonCell returns the current state in the requested representation. In the case of objectValue, this is an NSNumber containing true for on, false for off, and integer value -1 for the mixed state.

For more information on the behavior of NSButtonCell, see the NSButton and NSMatrix class specifications, and Button Programming Topics.

Subclass of the NSActionCellMBS class.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 14.2pr8](#)
- [MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.0](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr6](#)

Xojo Developer Magazine

- [12.2, page 10: News](#)

9.17.2 Methods

9.17.3 Constructor(image as NSImageMBS)

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with an image.

Example:

```
dim pic as Picture = LogoMBS(500)
dim n as NSImageMBS = new NSImageMBS(pic)
```

```
dim c as new NSButtonCellMBS(n)
```

```
Backdrop = c.image.CopyPictureWithMask
```

```
Title = c.classPath
```

See also:

- 9.17.4 Constructor(text as string) 544

9.17.4 Constructor(text as string)

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with a text.

Example:

```
dim c as new NSButtonCellMBS("Hello")
```

```
MsgBox c.StringValue
```

See also:

- 9.17.3 Constructor(image as NSImageMBS) 543

9.17.5 Properties

9.17.6 alternateImage as NSImageMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The image the button displays in its alternate state and, if necessary, redraws its contents.

Notes: Note that some button types don't display an alternate image.

(Read and Write property)

9.17.7 alternateTitle as String

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The title the button displays when it's in its alternate state.

Notes: Note that some button types don't display an alternate title.

(Read and Write property)

9.17.8 attributedAlternateTitle as NSAttributedStringMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The string the button displays when it's in its alternate state to the given attributed string.

Notes: Note that some button types don't display an alternate title.

Graphics attributes that are set on the cell (backgroundColor, alignment, font, etc.) are overridden when corresponding properties are set for the attributed string.

(Read and Write property)

9.17.9 attributedTitle as NSAttributedStringMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The string the button displays when it's in its normal state to the given attributed string and redraws the button.

Notes: The title is always shown on buttons that don't use their alternate contents when highlighting or displaying their alternate state.

Graphics attributes configured for the cell (backgroundColor, alignment, font, etc.) are overridden when corresponding properties are set for the attributed string.

(Read and Write property)

9.17.10 backgroundColor as NSColorMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The background color for the button.

Notes: (Read and Write property)

9.17.11 imageDimsWhenDisabled as Boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Whether to dim image when button is disabled.

Notes: When disabled, the image and text of an NSButtonCell are normally dimmed with gray. Radio buttons and switches use (imageDimsWhenDisabled = false) so only their text is dimmed.

(Read and Write property)

9.17.12 `imagePosition` as Integer

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The position of the receiver's image relative to its title.

Notes: (Read and Write property)

9.17.13 `imageScaling` as Integer

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The scale factor for the receiver's image.

Notes: Available in OS X v10.5 and later.

(Read and Write property)

9.17.14 `showsBorderOnlyWhileMouseInside` as Boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Whether to show border only while mouse is inside.

Notes: (Read and Write property)

9.17.15 `sound` as Variant

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: The sound that's played when the user presses the receiver.

Notes: The sound is played during a mouse-down event, such as `NSLeftMouseDown`.

Value is a `NSSoundMBS` object.

(Read and Write property)

9.18 class NSCellMBS

9.18.1 class NSCellMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The NSCell class provides a mechanism for displaying text or images in an NSView object without the overhead of a full NSView subclass.

Notes: It's used heavily by most of the NSControl classes to implement their internal workings.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr4](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr6](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr7](#)
- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)

Xojo Developer Magazine

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

9.18.2 Methods

9.18.3 acceptsFirstResponder as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver accepts first responder status.

Notes: The default value is true if the receiver is enabled. Subclasses may override this method to return a different value.

9.18.4 calcDrawInfo(theRect as NSRectMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Recalculates the cell geometry.

Notes: Objects (such as controls) that manage NSCell objects generally maintain a flag that informs them if any of their cells have been modified in such a way that the location or size of the cell should be recomputed. If so, calcSize method of NSControl is automatically invoked prior to the display of the cell, and that method invokes the calcDrawInfo method of the cell.

The default implementation of this method does nothing.

9.18.5 `cellSize` as `NSSizeMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the minimum size needed to display the receiver.

Notes: Returns the size of the cell, or the size (10000, 10000) if the receiver is not a text or image cell. If the cell is an image cell but no image has been set, returns `NSZeroSize`.

This method takes into account of the size of the image or text within a certain offset determined by the border type of the cell.

9.18.6 `cellSizeForBounds(theRect as NSRectMBS)` as `NSSizeMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the minimum size needed to display the receiver, constraining it to the specified rectangle.

Notes: `aRect`: The size of the cell, or the size of the `aRect` parameter if the cell is not a text or image cell. If the cell is an image cell but no image has been set, returns an empty size.

This method takes into account of the size of the image or text within a certain offset determined by the border type of the cell. If the receiver is of text type, the text is resized to fit within `aRect` (as much as `aRect` is within the bounds of the cell).

9.18.7 `compare(otherCell as NSCellMBS)` as `Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Compares the string values of the receiver another cell, disregarding case.

Notes: `otherCell`: The cell to compare against the receiver. This parameter must be of type `NSCell`; if it is not, this method raises `NSBadComparisonException`.

This value must not be `nil`. If the value is `nil`, the behavior is undefined and may change in future versions of Mac OS X.

Returns `NSOrderedAscending` if the string value of the receiver precedes the string value of `otherCell` in lexical ordering, `NSOrderedSame` if the string values are equivalent in lexical value, and `NSOrderedDescending` if the string value of the receiver follows the string value of `otherCell` in lexical ordering.

9.18.8 `Constructor(image as NSImageMBS)`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with an image.

Example:

```
dim pic as Picture = LogoMBS(500)
dim n as NSImageMBS = new NSImageMBS(pic)
dim c as new NSCellMBS(n)
```

```
Backdrop = c.image.CopyPictureWithMask
Title = c.classPath
```

See also:

- 9.18.9 Constructor(text as string)

549

9.18.9 Constructor(text as string)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with a text.

Example:

```
dim c as new NSCellMBS("Hello")
MsgBox c.StringValue
```

See also:

- 9.18.8 Constructor(image as NSImageMBS)

548

9.18.10 defaultFocusRingType as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the default type of focus ring for the receiver.

Notes: Use this constants:

NSFocusRingTypeDefault	= 0	The default focus ring type for NSView or NSCell.
NSFocusRingTypeNone	= 1	No focus ring. If you set the focus ring type to this value, NSView and NSCell will not draw any focus ring.
NSFocusRingTypeExterior	= 2	The standard Aqua focus ring.

9.18.11 defaultMenu as NSMenuMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the default menu for instances of the receiver.

Notes: Returns the default menu. The NSCell implementation of this method returns nil.

9.18.12 drawingRectForBounds(theRect as NSRectMBS) as NSRectMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the rectangle within which the receiver draws itself.

Notes: theRect: The bounding rectangle of the receiver.

Returns the rectangle in which the receiver draws itself. This rectangle is slightly inset from the one in theRect.

9.18.13 highlightColorWithFrame(theRect as NSRectMBS, controlView as NSViewMBS) as NSColorMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Redraws the receiver with the specified highlight setting.

Notes: theRect: The bounding rectangle of the receiver.

controlView: The control that manages the cell.

Returns the color the receiver uses when drawing the selection highlight.

You should not assume that a cell would necessarily want to draw itself with the value returned from selectedControlColor. A cell may wish to draw with different a selection highlight color depending on such things as the key state of its controlView.

9.18.14 imageRectForBounds(theRect as NSRectMBS) as NSRectMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the rectangle in which the receiver draws its image.

Notes: theRect: The bounding rectangle of the receiver.

The rectangle in which the receiver draws its image. This rectangle is slightly offset from the one in theRect.

9.18.15 isEntryAcceptable(aString as string) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns whether a string representing a numeric or date value is formatted in a suitable way for the cell's entry type.

Notes: This method is being deprecated in favor of a new class of formatter objects. For more information, see `NSFormatter`. This documentation is provided only for developers who need to modify older applications

9.18.16 mnemonic as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the character in the receiver's title that appears underlined for use as a mnemonic.

Notes: A string containing the mnemonic character, or an empty string if no mnemonic character is set.

9.18.17 nextState as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's next state.

Notes: If the receiver has three states, it cycles through them in this order: on, off, mixed, on, and so forth. If the receiver has two states, it toggles between them.

9.18.18 performClick

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Simulates a single mouse click on the receiver.

Notes: This method performs the receiver's action on its target. The receiver must be enabled to perform the action. If the receiver's control view is valid, that view is used as the sender; otherwise, the value in sender is used.

The receiver of this message must be a cell of type `NSActionCell`. This method raises an exception if the action message cannot be successfully sent.

9.18.19 prefersTrackingUntilMouseUp as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether tracking stops when the cursor leaves the cell.

Notes: The default implementation returns false. Subclasses may override this method to return a different value.

9.18.20 `sendActionOn(mask as Integer) as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets the conditions on which the receiver sends action messages to its target.

Notes: `mask`: A bit mask containing the conditions for sending the action. The only conditions that are actually checked are associated with the `NSLeftMouseDownMask`, `NSLeftMouseUpMask`, `NSLeftMouseDraggedMask`, and `NSPeriodicMask` bits.

Returns a bit mask containing the previous settings. This bit mask uses the same values as specified in the `mask` parameter.

You use this method during mouse tracking when the mouse button changes state, the mouse moves, or if the cell is marked to send its action continuously while tracking. Because of this, the only bits checked in `mask` are `NSLeftMouseDownMask`, `NSLeftMouseUpMask`, `NSLeftMouseDraggedMask`, and `NSPeriodicMask`, which are declared in the `NSEvent` class reference.

You can use the `setContinuous` method to turn on the flag corresponding to `NSPeriodicMask` or `NSLeftMouseDraggedMask`, whichever is appropriate to the given subclass of `NSCell`.

9.18.21 `setNextState`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Changes the state of the receiver to its next state.

Notes: If the receiver has three states, it cycles through them in this order: on, off, mixed, on, and so forth. If the receiver has two states, it toggles between them.

9.18.22 `setTitleWithMnemonic(stringWithAmpersand as string)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets the title of the receiver with one character in the string denoted as an access key.

Notes: `stringWithAmpersand`: The new title of the cell. One character in the string should be preceded by an ampersand (&) character. The character that follows becomes the mnemonic character for the title. Mnemonics are not supported in Mac OS X.

9.18.23 titleRectForBounds(theRect as NSRectMBS) as NSRectMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the rectangle in which the receiver draws its title text.

Notes: If the receiver is a text-type cell, this method resizes the drawing rectangle for the title (theRect) inward by a small offset to accommodate the cell border. If the receiver is not a text-type cell, the method does nothing.

9.18.24 wantsNotificationForMarkedText as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the field editor initiated by the receiver should post text change notifications.

Notes: Returns true if the field editor initiated by the receiver should post text change notifications (NSTextDidChangeNotification) while editing marked text; otherwise, they are delayed until the marked text confirmation.

9.18.25 Properties

9.18.26 alignment as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The alignment of text in the receiver..

Notes: The alignment of text in the receiver (one of the following constants: NSLeftTextAlignment, NSRightTextAlignment, NSCenterTextAlignment, NSJustifiedTextAlignment, NSNaturalTextAlignment).

The default value is NSNaturalTextAlignment.

(Read and Write property)

9.18.27 allowsEditingTextAttributes as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver allows user editing of textual attributes.

Notes: (Read and Write property)

9.18.28 allowsMixedState as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver supports three states.

Notes: Value is true if the receiver supports all three states (on, off, and mixed), otherwise false (the receiver supports only the on and off states).

(Read and Write property)

9.18.29 allowsUndo as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver assumes responsibility for undo operations.

Notes: By default, the `NSTextFieldCell` class uses this feature to handle undo operations for edited text. Other controls set a value that is appropriate for their implementation.

Available in Mac OS X v10.4 and later.

(Read and Write property)

9.18.30 attributedStringValue as NSAttributedStringMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The value of the receiver's cell as an attributed string using the receiver's formatter object (if one exists).

Notes: The textual attributes are the default paragraph style, the receiver's font and alignment, and whether the receiver is enabled and scrollable.

For Mac OS X v10.3 and later: If you use a class that responds to the selector `attributedStringValue` for the object value of a cell, then the cell will use that method to fetch the string to draw rather than using `stringValue`.

(Read and Write property)

9.18.31 backgroundColor as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the background style for the receiver.

Notes: The background describes the surface the cell is drawn onto in `drawWithFrame`. A control typically sets this before it asks the cell to draw. A cell may draw differently based on background characteristics. For example, a tableview drawing a cell in a selected row might set `cell.backgroundColor=NSBackgroundColorDark`. A text cell might decide to render its text white as a result. A rating-style level indicator might draw its stars white instead of gray.

Available in Mac OS X v10.5 and later.

(Read and Write property)

9.18.32 baseWritingDirection as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The initial writing direction used to determine the actual writing direction for text.

Notes: The default value is NSWritingDirectionNatural.

The Text system uses this value as a hint for calculating the actual direction for displaying Unicode characters. You should not need to call this method directly.

(Read and Write property)

9.18.33 Bezeled as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver has a bezeled border.

Notes: (Read and Write property)

9.18.34 Bordered as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver has a plain border.

Notes: (Read and Write property)

9.18.35 className as string

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The name of this NSCell class.

Example:

```
dim c as new NSActionCellMBS("Hello")
MsgBox c.className // shows "NSActionCell"
```

Notes: (Read only property)

9.18.36 classPath as string

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The path of this NSCell class.

Example:

```
dim c as new NSActionCellMBS("Hello")
MsgBox c.classPath // shows "NSActionCell:NSCell:NSObject"
```

Notes: Useful for debugging to know what super classes the view has.
(Read only property)

9.18.37 Continuous as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver's cell sends its action message continuously to its target during mouse tracking.

Notes: (Read and Write property)

9.18.38 controlSize as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The size of the receiver.

Notes: Can be NSRegularControlSize, NSMiniControlSize or NSSmallControlSize.
(Read and Write property)

9.18.39 controlTint as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's control tint.

Notes: Can be NSGraphiteControlTint, NSBlueControlTint, NSClearControlTint or NSDefaultControlTint.
(Read and Write property)

9.18.40 controlView as NSViewMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's control view.

Notes: The control view represents the control currently being rendered by the cell.

(Read and Write property)

9.18.41 doubleValue as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The value of the receiver's cell as a double-precision floating-point number.

Notes: The value of the cell interpreted as a double-precision floating-point number. If the receiver is not a text-type cell or the cell value is not scannable, returns 0.

(Read and Write property)

9.18.42 Editable as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver is editable.

Notes: (Read and Write property)

9.18.43 Enabled as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver is enabled or disabled.

Notes: (Read and Write property)

9.18.44 floatValue as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The value of the receiver's cell as a single-precision floating-point number.

Notes: Returns the value of the cell interpreted as a single-precision floating-point number. If the receiver is not a text-type cell or the cell value is not scannable, returns 0.

(Read and Write property)

9.18.45 font as NSFontMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The font used to display text in the receiver.

Notes: The receiver's current font, or nil if the receiver is not a text-type cell.

(Read and Write property)

9.18.46 Handle as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The internal NSCell reference.

Notes: (Read and Write property)

9.18.47 hasValidObjectValue as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver has a valid object value.

Notes: A valid object value is one that the receiver's formatter can "understand." Objects are always assumed to be valid unless they are rejected by the formatter.

(Read only property)

9.18.48 Highlighted as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver is highlighted.

Notes: (Read and Write property)

9.18.49 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)

9.18.50 image as NSImageMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The image displayed by the receiver (if any).

Notes: The image displayed by the receiver, or nil if the receiver is not an image-type cell.
(Read and Write property)

9.18.51 importsGraphics as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the text of the receiver can contain imported graphics.

Notes: true if the receiver's text is in the RTFD format and supports imported graphics, otherwise false.
(Read and Write property)

9.18.52 interiorBackgroundStyle as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the interior background style for the receiver.

Notes: The interior background style describes the surface drawn onto in `drawInteriorWithFrame:inView:.` This is often the same as the `backgroundStyle`, but a button that draws a bezel would have a different `interiorBackgroundStyle`.

This is both an override point and a useful method to call. In a custom button with a custom bezel you can override this method to describe that surface. A cell that has custom interior drawing might query this method to help pick an image that looks good on the cell. Calling this method gives you some independence from changes in framework art style.

Available in Mac OS X v10.5 and later.

(Read only property)

9.18.53 intValue as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's value as an integer.

Notes: The value of the cell interpreted as an integer. If the receiver is not a text-type cell or the cell value is not scannable, returns 0.

On Mac OS X v10.5 and later, you should use integerValue instead.
(Read and Write property)

9.18.54 isOpaque as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver is opaque (nontransparent).

Notes: (Read only property)

9.18.55 keyEquivalent as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the key equivalent to clicking the cell.

Notes: Subclasses can override this method to return a string with a valid character for the key equivalent.
(Read only property)

9.18.56 lineBreakMode as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The line break mode currently used when drawing text.

Notes: The line break mode the receiver currently uses when drawing text (one of the following constants: `NSLineBreakByWordWrapping`, `NSLineBreakByCharWrapping`, `NSLineBreakByClipping`, `NSLineBreakByTruncatingHead`, `NSLineBreakByTruncatingTail`, or `NSLineBreakByTruncatingMiddle`).

Available in Mac OS X v10.4 and later.

(Read and Write property)

9.18.57 menu as NSMenuMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's contextual menu.

Notes: The receiver's contextual menu, or nil if no menu is assigned.

(Read and Write property)

9.18.58 mnemonicLocation as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The position of the underlined mnemonic character in the receiver's title.

Notes: A zero-based index into the receiver's title string indicating the position of the character. If there is no mnemonic character, this method returns `NSNotFound`.

Mnemonics are not supported in Mac OS X.

(Read and Write property)

9.18.59 refusesFirstResponder as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver should not become the first responder.

Notes: To find out whether the receiver can become first responder at this time, use the method `acceptsFirstResponder`.

(Read and Write property)

9.18.60 Scrollable as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver scrolls excess text past the cell's bounds.

Notes: (Read and Write property)

9.18.61 Selectable as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the text of the receiver can be selected.

Notes: (Read and Write property)

9.18.62 sendsActionOnEndEditing as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver's `NSControl` object sends its action message whenever the user finishes editing the cell's text.

Notes: If this method returns true, the receiver's `NSControl` object sends its action message when the user does one of the following:

- Presses the Return key
- Presses the Tab key to move out of the field
- Clicks another text field

If it returns false, the cell's `NSControl` object sends its action message only when the user presses the Return key.

(Read and Write property)

9.18.63 `showsFirstResponder` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver should draw some indication of its first responder status.

Notes: The `NSCell` class itself does not draw a first-responder indicator. Subclasses may use the returned value to determine whether or not they should draw one, however.

(Read and Write property)

9.18.64 `state` as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's state.

Notes: Cells can have two or three states. If the receiver has two states, it returns either `NSOffState` (the normal or unpressed state) or `NSOnState` (the alternate or pressed state). If it has three, it may also return `NSMixedState`, indicating the feature is in effect somewhere.

To check whether the receiver uses the mixed state, use the method `allowsMixedState`.

Note that the value `state` returns may not be the same value you passed into `setState`.

(Read and Write property)

9.18.65 `stringValue` as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The value of the receiver's cell as a string.

Notes: If no formatter exists and the cell's value is a string, this method returns the value as a plain, attributed, or localized formatted string. If the value is not a string or cannot be converted to one, this method returns an empty string.

For Mac OS X v10.3 and later: If you use a class that responds to the selector `attributedStringValue` for the object value of a cell, the cell uses that method to fetch the string to draw rather than the `stringValue`

method.

(Read and Write property)

9.18.66 tag as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The tag identifying the receiver.

Notes: The tag value. The NSCell implementation of this method returns -1.

Tags allow you to identify particular cells. Tag values are not used internally; they are only changed by external invocations of `setTag:`. You typically set tag values in Interface Builder and use them at runtime in your application. When you set the tag of a control with a single cell in Interface Builder, it sets the tags of both the control and the cell to the same value as a convenience.

(Read and Write property)

9.18.67 title as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's title.

Notes: Subclasses (such as `NSButtonCell`) may override this method to return a different value.

(Read and Write property)

9.18.68 truncatesLastVisibleLine as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether the receiver truncates and adds the ellipsis character to the last visible line if the text doesn't fit into the cell bounds.

Notes: The line break mode must be either `NSLineBreakByWordWrapping` or `NSLineBreakByCharWrapping`. Otherwise, this setting is ignored.

Available in Mac OS X v10.5 and later.

(Read and Write property)

9.18.69 type as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The type of the cell, changing it to a text cell, image cell, or null cell.

Notes: If the cell is already the same type as the one specified in the `aType` parameter, this method does nothing.

If `aType` is `NSTextCellType`, this method converts the receiver to a cell of that type, giving it a default title and setting the font to the system font at the default size. If `aType` is `NSImageCellType`, the cell type is not changed until you set a new non-nil image.

(Read and Write property)

9.18.70 `userInterfaceLayoutDirection` as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: The layout direction of the user interface.

Notes: This property specifies the general user interface layout flow directions. For subclasses that have multiple visual components in a single cell instance, this property should specify the directionality or flow of components.

`NSUserInterfaceLayoutDirectionLeftToRight = 0`

`NSUserInterfaceLayoutDirectionRightToLeft = 1`

Available in OS X v10.6 and later.

(Read and Write property)

9.18.71 `usesSingleLineMode` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether the text cell restricts layout and rendering of its content to a single line.

Notes: If true, the cell ignores the return value from `wraps`, interprets `NSLineBreakByWordWrapping` and `NSLineBreakByCharWrapping` returned by `lineBreakMode` as `NSLineBreakByClipping`, and configures the field editor to ignore key binding commands that insert paragraph and line separators.

The field editor bound to a single line cell filters paragraph and line separator insertion from user actions. Cells in the single line mode use the fixed baseline layout. The text baseline position is determined solely by the control size regardless of content font style or size.

(Read and Write property)

9.18.72 `wraps` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether text in the receiver wraps when its length exceeds the frame of the cell.

Notes: If the text of the receiver is an attributed string value you must explicitly set the paragraph style line break mode. Calling this method with the value true is equivalent to calling the `setLineBreakMode:`

method with the value `NSLineBreakByWordWrapping`.
(Read and Write property)

9.18.73 `cellAttribute(aParameter as Integer) as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The value for the specified cell attribute.

Notes: `aParameter`: The cell attribute whose value you want to get. Attributes include the receiver's current state and whether it is disabled, editable, or highlighted.

Returns the value for the cell attribute specified by `aParameter`.
(Read and Write computed property)

9.18.74 `focusRingType as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The type of focus ring currently set for the receiver.

Notes: You can disable a view's focus ring drawing by overriding this method so it always returns `NSFocusRingTypeNone`, or by calling `setFocusRingType:` with `NSFocusRingTypeNone`. You should only disable a view from drawing its focus ring if you want to draw your own focus ring, or if there isn't sufficient space to display a focus ring in the default location.

Available in Mac OS X v10.3 and later.

(Read and Write computed property)

9.18.75 Constants

Constants

Constant	Value	Description
<code>NSAnyType</code>	0	One of the constants specify how a cell formats numeric data. Any value is allowed. Deprecated in Mac OS X v10.4 and later.
<code>NSBlueControlTint</code>	1	One of the constants specify a cell's tint. Aqua control tint Available in Mac OS X v10.3 and later.
<code>NSCellAllowsMixedState</code>	16	One of the constants specify how a button behaves when pressed and how displays its state. Lets the cell's state be <code>NSMixedState</code> , as well as <code>NSOffState</code> and <code>NSOnState</code> .
<code>NSCellChangesContents</code>	14	One of the constants specify how a button behaves when pressed and how displays its state. If the cell's state is <code>NSMixedState</code> or <code>NSOnState</code> , displays the cell's alternate image.
<code>NSCellDisabled</code>	0	One of the constants specify how a button behaves when pressed and how displays its state. Does not let the user manipulate the cell.
<code>NSCellEditable</code>	3	One of the constants specify how a button behaves when pressed and how displays its state. Lets the user edit the cell's contents.
<code>NSCellHasImageHorizontal</code>	12	One of the constants specify how a button behaves when pressed and how displays its state. Controls the position of the cell's image: places the image on the right of any text in the cell. Together, <code>NSCellHasImageOnLeftOrBottom</code> , <code>NSCellHasImageHorizontal</code> , and <code>NSCellHasOverlappingImage</code> control the position of the cell's image and text. To place the image above, set none of them. To place the image below, set <code>NSCellHasImageOnLeftOrBottom</code> . To place the image to the right, set <code>NSCellHasImageHorizontal</code> . To place the image to the left, set <code>NSCellHasImageHorizontal</code> and <code>NSCellHasImageOnLeftOrBottom</code> . To place the image directly over, set <code>NSCellHasOverlappingImage</code> .
<code>NSCellHasImageOnLeftOrBottom</code>	13	One of the constants specify how a button behaves when pressed and how displays its state. Controls the position of the cell's image: places the image on the left of or below any text in the cell. See <code>NSCellHasImageHorizontal</code> for more details.
<code>NSCellHasOverlappingImage</code>	11	One of the constants specify how a button behaves when pressed and how displays its state. Controls the position of the cell's image: places the image over any text in the cell. See <code>NSCellHasImageHorizontal</code> for more details.
<code>NSCellHighlighted</code>	5	One of the constants specify how a button behaves when pressed and how displays its state. Draws the cell with a highlighted appearance. (Deprecated. Use <code>Highlighted</code> instead.)
<code>NSCellHitContentArea</code>	1	One of the constants are used by <code>hitTestForEvent</code> to determine the effect of an event. A content area in the cell. Available in Mac OS X v10.5 and later.
<code>NSCellHitEditableTextArea</code>	2	One of the constants are used by <code>hitTestForEvent</code> to determine the effect of an event. An editable text area of the cell. Available in Mac OS X v10.5 and later.
<code>NSCellHitNone</code>	0	One of the constants are used by <code>hitTestForEvent</code> to determine the effect of an event. An empty area, or did not hit in the cell. Available in Mac OS X v10.5 and later.
<code>NSCellHitTrackableArea</code>	4	One of the constants are used by <code>hitTestForEvent</code> to determine the effect of an event.

Background Styles

Constant	Value	Description
NSBackgroundStyleDark	1	The background is a dark color. Light content contrasts well with this background. Available in Mac OS X v10.5 and later.
NSBackgroundStyleLight	0	The background is a light color. Dark content contrasts well with this background. Available in Mac OS X v10.5 and later.
NSBackgroundStyleLowered	3	The background is intended to appear lower than the content drawn on it. Content might need to be embossed. Available in Mac OS X v10.5 and later.
NSBackgroundStyleRaised	2	The background is intended to appear higher than the content drawn on it. Content might need to be inset. Available in Mac OS X v10.5 and later.

Scaling Modes

Constant	Value	Description
NSImageScaleAxesIndependently	1	Scale each dimension to exactly fit destination. This setting does not preserve the aspect ratio of the image. Available in Mac OS X v10.5 and later.
NSImageScaleNone	2	Do not scale the image. Available in Mac OS X v10.5 and later.
NSImageScaleProportionallyDown	0	If it is too large for the destination, scale the image down while preserving aspect ratio. Available in Mac OS X v10.5 and later.
NSImageScaleProportionallyUpOrDown	3	Scale the image to its maximum possible dimensions while both staying within the destination area and preserving its aspect ratio. Available in Mac OS X v10.5 and later.

9.19 class NSControlMBS

9.19.1 class NSControlMBS

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The Cocoa class for a NSControl.

Notes: You can embed this view in a CustomNSViewMBS to get more events for mouse and keyboard. Subclass of the NSViewMBS class.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.1](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr7](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr2](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr1](#)
- [MBS Real Studio Plugins, version 13.0pr10](#)
- [MBS Real Studio Plugins, version 12.5pr11](#)
- [MBS Real Studio Plugins, version 12.5pr8](#)
- [MBS Real Studio Plugins, version 12.5pr6](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS Plugins 10.3 Release Notes](#)

Xojo Developer Magazine

- [20.4, page 10: News](#)
- [20.3, page 10: News](#)

9.19.2 Methods

9.19.3 calcSize

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Recomputes any internal sizing information for the receiver, if necessary.

Notes: This method uses the calcDrawInfo method of its cell to perform the calculations. Most controls maintain a flag that informs them if any of their cells have been modified in such a way that the location or size of the cell should be recomputed. If such a modification happens, this method is automatically invoked before the control is displayed. You should never need to invoke it yourself.

9.19.4 ConnectActionEvent

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Connects the action event.

Notes: If you want to use addhandler with this class and the action event, you need to call ConnectActionEvent after addhandler to actually have the plugin put things in place for handling the event.

9.19.5 Constructor

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new control with size 100/100 and position 0/0

Example:

```
dim t as new NSControlMBS
```

Notes: On success the handle property is not zero.

See also:

- 9.19.6 Constructor(Handle as Integer) 569
- 9.19.7 Constructor(left as Double, top as Double, width as Double, height as Double) 570

9.19.6 Constructor(Handle as Integer)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSControl handle.

Example:

```
dim t as new NSControlMBS(0, 0, 100, 100)
```

```
dim v as new NSControlMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSControl and the plugin retains this handle.

See also:

- 9.19.5 Constructor 569
- 9.19.7 Constructor(left as Double, top as Double, width as Double, height as Double) 570

9.19.7 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new control with the given size and position.

Example:

```
dim x as new NSControlMBS(0, 0, 100, 20)
```

Notes: On success the handle property is not zero.

See also:

- 9.19.5 Constructor 569
- 9.19.6 Constructor(Handle as Integer) 569

9.19.8 Destructor

Plugin Version: 13.0, Platform: macOS, Targets: Desktop only.

Function: The destructor.

9.19.9 EnableEvents

Plugin Version: 13.4, Platform: macOS, Targets: Desktop only.

Function: Enables events after you assigned methods to them with AddHandler.

9.19.10 performClick

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Simulates a single mouse click on the receiver.

Notes: This method calls the performClick method of the receiver's cell. This method raises an exception if the action message cannot be successfully sent.

9.19.11 selectCell(Cell as NSCellMBS)

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Selects the specified cell and redraws the control as needed.

Notes: Cell: The cell to select. The cell must belong to the receiver.

If the cell is already selected (or does not belong to the receiver), this method does nothing. If the cell belongs to the receiver and is not selected, this method changes its state to `NSOnState` and redraws the cell.

9.19.12 `setNeedsDisplay`

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Marks the receiver as needing redisplay (assuming automatic display is enabled).

Notes: This method also recalculates the dimensions of the control as needed.

9.19.13 `sizeToFit`

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Resizes the receiver's frame so that it is the minimum size needed to contain its cell.

Notes: If you want a multiple-cell custom subclass of `NSControl` to size itself to fit its cells, you must override this method. This method neither redisplay the receiver nor marks it as needing display. You must do this yourself with either `redisplay` or `setNeedsDisplay` method.

9.19.14 `validateEditing`

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Validates changes to any user-typed text.

Notes: Validation sets the object value of the cell to the current contents of the cell's editor (the `NSText` object used for editing), storing it as a simple string or an attributed string object based on the attributes of the editor.

9.19.15 Properties

9.19.16 `ActionSelector` as String

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The name of the selector.

Notes: The new action-message selector to associate with the receiver's cell. Specify `NULL` to prevent action messages from being sent to the receiver's target.

(Read and Write property)

9.19.17 alignment as Integer

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: The alignment mode of the text in the view's cell.

Example:

```
dim n as NSControlMBS // your control
n.alignment=3
```

Notes: One of the following constants: `NSLeftTextAlignment`, `NSRightTextAlignment`, `NSCenterTextAlignment`, `NSJustifiedTextAlignment`, or `NSNaturalTextAlignment`. The default value is `NSNaturalTextAlignment`.

Constants:

`NSLeftTextAlignment = 0`
Text is visually left aligned.

`NSRightTextAlignment = 1`
Text is visually right aligned.

`NSCenterTextAlignment = 2`
Text is visually center aligned.

`NSJustifiedTextAlignment = 3`
Text is justified.

`NSNaturalTextAlignment = 4`
Use the natural alignment of the text's script.
(Read and Write property)

9.19.18 attributedStringValue as NSAttributedStringMBS

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: The value of the view's cell as an attributed string.

Notes: The value of the cell interpreted as an attributed string, or an empty attributed string if the receiver

has no cell.

If the control contains many cells (for example, `NSMatrix`), then the value of the currently selected cell is returned. If the control is in the process of editing the affected cell, then it invokes the `validateEditing` method before extracting and returning the value.

(Read and Write property)

9.19.19 `baseWritingDirection` as Integer

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: The initial writing direction used to determine the actual writing direction for text.

Notes: One of the following values: `NSWritingDirectionNatural`, `NSWritingDirectionLeftToRight`, or `NSWritingDirectionRightToLeft`. The default value is `NSWritingDirectionNatural`.

The Text system uses this value as a hint for calculating the actual direction for displaying Unicode characters. You should not need to call this method directly.

(Read and Write property)

9.19.20 `cell` as Variant

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The receiver's cell object.

Notes: Cocoa controls often have a frame control and inside a `Cell` which implements the raw functionality. This way you can for example have a table which embeds such cell controls inside the table cells.

Use this method with great care as it can irrevocably damage the affected control; specifically, you should only use this method in initializers for subclasses of `NSControl`.

(Read and Write property)

9.19.21 `controlSize` as Integer

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: The size of the cell.

Notes: Use this property to change the rendered size of the cell and its control.

Changing the cell's control size does not change the font used by the cell. Use the `systemFontSize` class method of `NSFontMBS` to obtain an appropriate font based on the new control size.

If you set this property for Xojo's controls, you may need to set it again in Resized events as it may reset to the default.

Use constants:

ControlSizeRegular	0	Regular
ControlSizeSmall	1	Small size
ControlSizeMini	2	Mini size
ControlSizeLarge	3	Large size

(Read and Write property)

9.19.22 currentEditor as NSTextMBS

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Returns the current field editor for the control.

Notes: Returns the field editor for the current control, or nil if the receiver does not have a field editor.

When the receiver is a control displaying editable text (for example, a text field) and it is the first responder, it has a field editor, which is returned by this method. 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.

(Read only property)

9.19.23 doubleValue as Double

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The double value of the control.

Notes: (Read and Write property)

9.19.24 font as NSFontMBS

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: The font used to draw text in the receiver's cell.

Notes: If the cell is being edited, the text in the cell is redrawn in the new font, and the cell's editor (the NSText object used globally for editing) is updated with the new font object.

(Read and Write property)

9.19.25 ignoresMultiClick as boolean

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the receiver ignores multiple clicks made in rapid succession.

Example:

```
dim n as NSControlMBS // your control
n.ignoresMultiClick=True
```

Notes: True if the view ignores multiple clicks; otherwise, false.

(Read and Write property)

9.19.26 integerValue as Integer

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: The value of the receiver's cell as an Integer value.

Notes: If the control contains many cells (for example, NSMatrix), then the value of the currently selected cell is returned. If the control is in the process of editing the affected cell, then it invokes the validateEditing method before extracting and returning the value.

Available in OS X v10.5 and later.

(Read and Write property)

9.19.27 intValue as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The integer value of the control.

Example:

```
dim n as new nsbuttonMBS(0,0,100,100)
```

```
n.intValue=1
MsgBox str(n.intValue) // shows 1
n.intValue=0
MsgBox str(n.intValue) // shows 0
```

Notes: (Read and Write property)

9.19.28 isContinuous as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver’s cell sends its action message continuously to its target during mouse tracking.

Notes: True if the action message should be sent continuously; otherwise, false.
(Read and Write property)

9.19.29 isEnabled as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver reacts to mouse events.

Notes: True if the view responds to mouse events; otherwise, false.
(Read and Write property)

9.19.30 refusesFirstResponder as boolean

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver refuses the first responder role.

Notes: By default, the user can advance the focus of keyboard events between controls by pressing the Tab key; when this focus—or first responder status—is indicated for a control (by the insertion point or, for nontext controls, a faint rectangle), the user can activate the control by pressing the Space bar.
(Read and Write property)

9.19.31 selectedCell as NSCellMBS

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver’s selected cell.

Notes: The default implementation of this method simply returns the control’s associated cell (or nil if no cell has been set). Subclasses of NSControl that manage multiple cells (such as NSMatrix and NSForm) must override this method to return the cell selected by the user.
(Read only property)

9.19.32 selectedTag as Integer

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: The tag of the views's selected cell.

Notes: (Read only property)

9.19.33 stringValue as string

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The string value of the control.

Notes: (Read and Write property)

9.19.34 tag as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The tag identifying the object.

Notes: You can set this property to the value you need.
(Read and Write property)

9.19.35 Events

9.19.36 Action

Plugin Version: 8.0, Platform: macOS, Targets: .

Function: The control's action was triggered.

Notes: For a button if it was pressed.

9.19.37 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

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

Function: Sent when a control with editable text begins an editing session.

Notes: Notification: The notification object. The name of the notification is always NSControlTextDidBeginEditingNotification.

This event is invoked when the user begins editing text in a control such as a text field or a form field. The control posts a `NSControlTextDidBeginEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also delivered for inspection.

See `TextDidEndEditing` for an explanation of why you may not always get one invocation of `TextDidBeginEditing` for each invocation of `TextDidEndEditing`.

9.19.38 `TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)`

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

Function: Sent when the text in the receiving control changes.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidChangeNotification`.

This event is invoked when text in a control such as a text field or form changes. The control posts a `NSControlTextDidChangeNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is provided as parameter for inspection.

9.19.39 `TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)`

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

Function: Sent when a control with editable text ends an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidEndEditingNotification`.

This event is invoked when the user stops editing text in a control such as a text field or form. The control posts a `NSControlTextDidEndEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also provided for inspection.

Warning: In some cases, such as when editing within an instance of `NSOutlineView`, this method may be invoked without a previous invocation of `TextDidBeginEditing`. You will only get the `TextDidBeginEditing` notification if the user actually types something, but you can get the `TextDidEndEditing` notification if the user just double-clicks the field and then clicks outside the field, without typing.

9.19.40 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean

Plugin Version: 8.0, Platform: macOS, Targets: .

Function: The event called to decide whether text editing should be allowed.

Notes: Return true to allow text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.19.41 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean

Plugin Version: 8.0, Platform: macOS, Targets: .

Function: The event called to decide whether ending text editing should be allowed.

Notes: Return true to allow end of text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.19.42 Constants

Constants

Constant	Value	Description
ControlSizeLarge	3	One of the constants to specify a cell,Ãs size. A size larger than the default control size.
ControlSizeMini	2	One of the constants to specify a cell,Ãs size. The smallest control size.
ControlSizeRegular	0	One of the constants to specify a cell,Ãs size. The default control size.
ControlSizeSmall	1	One of the constants to specify a cell,Ãs size. A size smaller than the default control size. This constant is for controls that can,Ãt be resized in one direction, such as push buttons, radio buttons, checkboxes, sliders, scroll bars, pop-up buttons, tabs, and progress indicators. Use a small system font with a small control.

9.20 class NSImageCellMBS

9.20.1 class NSImageCellMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: An NSImageCell object displays a single image (encapsulated in an NSImage object) in a frame.

Notes: This class provides methods for choosing the frame and for aligning and scaling the image to fit the frame.

The object value of an NSImageCell object must be an NSImage object, so if you use the setObjectValue: method of NSCell, be sure to supply an NSImage object as an argument. Because an NSImage object does not need to be converted for display, do not use the NSCell methods relating to formatters.

An NSImageCell object is usually associated with some kind of control object—an NSImageView, an NSMatrix, or an NSTableView.

Subclass of the NSCellMBS class.

Blog Entries

- [MBS Xojo Plugins, version 21.2pr1](#)

9.20.2 Methods

9.20.3 Constructor(image as NSImageMBS)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with an image.

Example:

```
dim pic as Picture = LogoMBS(500)
dim n as NSImageMBS = new NSImageMBS(pic)
dim c as new NSImageCellMBS(n)
```

```
Backdrop = c.image.CopyPictureWithMask
```

```
Title = c.classPath
```

See also:

- 9.20.4 Constructor(text as string)

9.20.4 Constructor(text as string)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with a text.

Example:

```
dim c as new NSImageCellMBS("Hello")
MsgBox c.StringValue
```

See also:

- 9.20.3 Constructor(image as NSImageMBS)

580

9.20.5 Properties

9.20.6 imageAlignment as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The alignment of the receiver's image relative to its frame.

Notes: For a list of possible values, see NSImageAlign* constants. The default value is NSImageAlignCenter.

(Read and Write property)

9.20.7 imageFrameStyle as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The style of the frame that borders the image.

Notes: Value is one of the frame style constants. For a list of frame styles, see NSImageFrame* constants.

(Read and Write property)

9.20.8 imageScaling as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The scaling mode used to fit the receiver's image into the frame.

Notes: Value is one of the image scaling constants. For a list of possible values, see NSScale* constants.

(Read and Write property)

9.20.9 Constants

Constants

Constant	Value	Description
<code>NSImageAlignBottom</code>	5	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Align the image with the bottom edge of the cell.
<code>NSImageAlignBottomLeft</code>	6	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Align the image with the bottom and left edges of the cell.
<code>NSImageAlignBottomRight</code>	7	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Align the image with the bottom and right edges of the cell.
<code>NSImageAlignCenter</code>	0	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Center the image in the cell.
<code>NSImageAlignLeft</code>	4	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Align the image with the left edge of the cell.
<code>NSImageAlignRight</code>	8	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Position the image along the right edge of the cell.
<code>NSImageAlignTop</code>	1	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Position the image along the top edge of the cell.
<code>NSImageAlignTopLeft</code>	2	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Align the image with the top and left edges of the cell.
<code>NSImageAlignTopRight</code>	3	One of the constants to specify the location of the image in the frame using the <code>imageAlignment</code> property. Align the image with the top and right edges of the cell.
<code>NSImageFrameButton</code>	4	One of the frame constants for the <code>imageFrameStyle</code> property. A convex bezel that makes the image stand out in relief, like a button
<code>NSImageFrameGrayBezel</code>	2	One of the frame constants for the <code>imageFrameStyle</code> property. A gray, concave bezel that makes the image look sunken
<code>NSImageFrameGroove</code>	3	One of the frame constants for the <code>imageFrameStyle</code> property. A thin groove that looks etched around the image
<code>NSImageFrameNone</code>	0	One of the frame constants for the <code>imageFrameStyle</code> property. An invisible frame
<code>NSImageFramePhoto</code>	1	One of the frame constants for the <code>imageFrameStyle</code> property. A thin black outline and a dropped shadow

Scale Methods

Constant	Value	Description
NSImageScaleAxesIndependently	1	Scale each dimension to exactly fit destination. This setting does not preserve the aspect ratio of the image.
NSImageScaleNone	2	Do not scale the image.
NSImageScaleProportionallyDown	0	If it is too large for the destination, scale the image down while preserving aspect ratio.
NSImageScaleProportionallyUpOrDown	3	Scale the image to its maximum possible dimensions while both staying within the destination area and preserving its aspect ratio.
NSScaleNone	2	Do not scale the image.
NSScaleProportionally	0	If it is too large for the destination, scale the image down while preserving aspect ratio.
NSScaleToFit	1	Scale each dimension to exactly fit destination. This setting does not preserve the aspect ratio of the image.

9.21 class NSMenuItemCellMBS

9.21.1 class NSMenuItemCellMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: NSMenuItemCell is a class that handles the measurement and display of a single menu item in its encompassing frame.

Notes: Instances of NSMenuItemCell work in conjunction with an NSMenuView object to control the overall appearance of the menu.

Subclass of the NSButtonCellMBS class.

Blog Entries

- [MBS Xojo Plugins, version 17.1pr5](#)

9.21.2 Methods

9.21.3 calcSize

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Calculates the minimum required width and height of the receiver,Äôs menu item.

Notes: The calculated values are cached for future use. This method also calculates the sizes of individual components of the cell,Äôs menu item and caches those values.

This method is invoked automatically when necessary. You should not need to invoke it directly.

9.21.4 Constructor(image as NSImageMBS)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

Notes: Don't use it.

See also:

- [9.21.5 Constructor\(text as string\)](#) 584

9.21.5 Constructor(text as string)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The constructor.

See also:

- [9.21.4 Constructor\(image as NSImageMBS\)](#) 584

9.21.6 Properties

9.21.7 menuItem as NSMenuItemMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The menu item object associated with the cell.

Notes: (Read and Write property)

9.21.8 needsDisplay as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the menu item needs to be displayed.

Notes: Set this property to true when you want the menu item to be drawn.

(Read and Write property)

9.21.9 needsSizing as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the size of the menu needs to be calculated.

Notes: When the value of this property is true, the next attempt to obtain size information about the menu cause the calcSize method to be called. When the value of the property is false, the size information is obtained from the currently cached values.

Subclasses that drastically change the way a menu item is drawn can change the value of this property to update the menu item information. Other parts of your application should not need to change this property directly. The cell checks this value of this property as necessary when the content of its menu item changes.

(Read and Write property)

9.21.10 tag as Integer

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The integer tag of the selected menu item.

Notes: If no item is selected, the value in this property is 0.

(Read and Write property)

9.22 class `NSPathComponentCellMBS`

9.22.1 class `NSPathComponentCellMBS`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The `NSPathComponentCell` class displays a component of a path.

Notes: An `NSPathComponentCell` object manages a collection of `NSPathComponentCell` objects, in conjunction with an `NSPathComponentCell` object, to represent a path.

Subclass of the `NSPathComponentCellMBS` class.

Blog Entries

- [NSSegmentedControl and NSPathComponentCell](#)
- [MBS Real Studio Plugins, version 12.0pr6](#)

9.22.2 Methods

9.22.3 Constructor(`text as string`)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new `Cell` object with a text.

9.22.4 Properties

9.22.5 File as `folderitem`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The value of the portion of the path from the root through the component represented by the receiver.

Notes: (Read and Write computed property)

9.22.6 Image as `NSImageMBS`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The image displayed for this component cell.

Notes: Generally, a 16-by-16-point image fits best when the path style is `NSPathComponentCellStandard` or `NSPathComponentCellStylePopUp`, and a 14-by-14-point image is best when the path style is `NSPathComponentCellNavigationBar`.

Available in Mac OS X v10.5 and later.

(Read and Write computed property)

9.22.7 URL as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The value of the portion of the path from the root through the component represented by the receiver.

Notes: (Read and Write computed property)

9.23 control NSPathControlControlMBS

9.23.1 control NSPathControlControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The Xojo control to host a NSPathControl.

Notes: We know the fact, that the control wrapping the segmented control has control twice in the name. Suggestions to rename are welcome, but unlikely to happen as it would break code.

Blog Entries

- [News from the MBS Xojo Plugins Version 24.0](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 24.0](#)
- [NSSegmentedControl and NSPathControl](#)
- [MBS Xojo Plugins, version 24.0pr7](#)

Xojo Developer Magazine

- [22.2, page 9: News](#)

9.23.2 Properties

9.23.3 View as NSPathControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The reference to the underlying NSPathControlMBS.

Notes: (Read only property)

9.23.4 Events

9.23.5 Action

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event is called when user clicks on a date/time and changes something.

9.23.6 BoundsChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.23.7 Close

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control is about to close.

In Xojo version 2021r3 and newer this event is named Closing.

9.23.8 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

9.23.9 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Called when a menuitem is chosen.

Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

9.23.10 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.23.11 DoubleClick

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Called when a double click happened.

Notes: May call Action event before.

9.23.12 EnableMenuItems

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In Xojo version 2021r3 and newer this event is named MenuBarSelected.

9.23.13 FrameChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.23.14 GotFocus

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control itself got focus.

In Xojo version 2021r3 and newer this event is named FocusReceived.

Notes:

This only fires if the control itself got focus and not a sub control.

9.23.15 LostFocus

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control lost focus.

In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.23.16 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ãs region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.23.17 MouseDrag(x as Integer, y as Integer)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.23.18 MouseUp(x As Integer, y As Integer)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.23.19 Open

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control is about to was created and you can initialize it.

In Xojo version 2021r3 and newer this event is named Opening.

Example:

```
Sub Open() Handles Open
  // set small font
```

```
Dim p As NSPathControlMBS = Me.View
p.Font = NSFontMBS.systemFontOfSize(10)
```

```
// disable focus ring
p.focusRingType = p.NSFocusRingTypeNone
End Sub
```

9.23.20 ScaleFactorChanged(NewFactor as double)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.23.21 willDisplayOpenPanel(openPanel as Variant)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Implement this method to customize the Open panel shown by a pop-up-style path.

Notes: This method is called before the Open panel is shown but after its allowed file types are set to the cell's allowed types. At this time, you can further customize the Open panel as required. This method is called only when the style is set to NSPathStylePopUp. Implementation of this method is optional.

openPanel is a NSOpenPanelMBS object.

9.23.22 willPopUpMenu(menu as Variant)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Implement this method to customize the menu of a pop-up-style path.

Notes: This event is called before the pop-up menu is shown. At this time, you can further customize the menu as required, adding and removing items. This method is called only when the style is set to NSPathStylePopUp. Implementation of this method is optional.

Menu is a NSMenuMBS object.

9.23.23 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.24 class `NSPathControlMBS`

9.24.1 class `NSPathControlMBS`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: `NSPathControl` is a subclass of `NSControl` that represents a file system path or virtual path.

Notes: The `NSPathControl` class uses `NSPathCell` to implement its user interface. `NSPathControl` provides cover methods for most `NSPathCell` methods—the cover method simply invokes the corresponding cell method. See also `NSPathComponentCell`, which represents individual components of the path, and two associated protocols: `NSPathCellDelegate` and `NSPathControlDelegate`.

`NSPathControl` has three styles represented by the `NSPathStyle` enumeration constants `NSPathStyleStandard`, `NSPathStyleNavigationBar`, and `NSPathStylePopUp`. The represented path can be a file system path or any other type of path leading through a sequence of nodes or components, as defined by the programmer.

`NSPathControl` automatically supports drag and drop, which can be further customized via delegate methods. To accept drag and drop, `NSPathControl` calls `registerForDraggedTypes:` with `NSFileNamesPboardType` and `NSURLPboardType`. When the URL value in the `NSPathControl` object changes because of an automatic drag and drop operation or the user selecting a new path via the open panel, the action is sent. On Mac OS X v10.5 the value returned by `clickedPathComponentCell` is `nil`, on Mac OS X v10.6 and later, `clickedPathComponentCell` returns the clicked cell.

Subclass of the `NSControlMBS` class.

Blog Entries

- [News from the MBS Xojo Plugins Version 24.0](#)
- [NSSegmentedControl and NSPathControl](#)
- [MBS Xojo Plugins, version 24.0pr7](#)
- [MBS Releases the MBS Real Studio plug-ins in version 12.0](#)
- [MBS Real Studio Plugins, version 12.0pr5](#)

Xojo Developer Magazine

- [10.3, page 9: News](#)

9.24.2 Methods

9.24.3 `clickedPathComponentCell` as `NSPathComponentCellMBS`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Returns component cell that was clicked.

Notes: The value returned is generally valid only when the action or double action is being sent.

Note: In Mac OS X 10.5 and earlier the returned value was nil if no cell had been clicked. In Mac OS X 10.6, the folder of the cell that the user selected is returned instead.

9.24.4 Constructor

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new path control with size 100/100 and position 0/0

Example:

```
dim t as new NSPathControlMBS
```

Notes: On success the handle property is not zero.

See also:

- 9.24.5 Constructor(Handle as Integer) 595
- 9.24.6 Constructor(left as Double, top as Double, width as Double, height as Double) 595

9.24.5 Constructor(Handle as Integer)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSPathControl handle.

Example:

```
dim t as new NSPathControlMBS(0, 0, 100, 100)
dim v as new NSPathControlMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSPathControl and the plugin retains this handle.

See also:

- 9.24.4 Constructor 595
- 9.24.6 Constructor(left as Double, top as Double, width as Double, height as Double) 595

9.24.6 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new path control with the given size and position.

Example:

```
dim x as new NSPathControlMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 9.24.4 Constructor 595
- 9.24.5 Constructor(Handle as Integer) 595

9.24.7 pathComponentCells as NSPathComponentCellMBS()

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Returns an array of the NSPathComponentCell objects currently being displayed.

Notes: Available in Mac OS X v10.5 and later.

9.24.8 setDraggingSourceOperationMask(mask as Integer, local as boolean)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Configures the default value returned from draggingSourceOperationMaskForLocal.

Notes: mask: The types of drag operations allowed.

isLocal: If true, mask applies when the drag destination object is in the same application as the receiver; if false, mask applies when the destination object is outside the receiver's application.

By default, draggingSourceOperationMaskForLocal returns NSDragOperationEvery when isLocal is true and NSDragOperationNone when isLocal is false.

Available in Mac OS X v10.5 and later.

9.24.9 setPathComponentCells(cells() as NSPathComponentCellMBS)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Sets the array of NSPathComponentCell objects currently being displayed.

Notes: cells: An array of NSPathComponentCell objects.

Each item in the array must be an instance of `NSPathComponentCell` or a subclass thereof. You cannot set this value to `nil`, but you can set it to an empty array.

Available in Mac OS X v10.5 and later.

9.24.10 Properties

9.24.11 `backgroundColor` as `NSColorMBS`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The background color.

Notes: By default, the background is set to a light blue color for `NSPathStyleStandard` and `nil` for the other styles. You can use `NSColorMBS.clearColor` to make the background transparent.

Available in Mac OS X v10.5 and later.

(Read and Write computed property)

9.24.12 `file` as `folderitem`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The path property.

Notes: When setting, an array of `NSPathComponentCell` objects is automatically set based on the path in `url`. If `url` is a file URL (returns true from `isFileURL`), the images are automatically filled with file icons, if the path exists. The URL value itself is stored in the `objectValue` property of the cell.

See also URL property.

(Read and Write computed property)

9.24.13 `menu` as `NSMenuMBS`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The menu used for the path control's cells.

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

(Read and Write computed property)

9.24.14 `pathStyle` as `Integer`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The path style.

Notes: Either `NSPathStyleStandard` or `NSPathStylePopUp`.
(Read and Write computed property)

9.24.15 URL as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: The path property.

Notes: When setting, an array of `NSPathComponentCell` objects is automatically set based on the path in url. If url is a file URL (returns true from `isFileURL`), the images are automatically filled with file icons, if the path exists. The URL value itself is stored in the `objectValue` property of the cell.

See also File property.

(Read and Write computed property)

9.24.16 Events

9.24.17 DoubleClick

Plugin Version: 12.0, Platform: macOS, Targets: .

Function: Called on a double click.

9.24.18 willDisplayOpenPanel(openPanel as Variant)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Implement this method to customize the Open panel shown by a pop-up-style path.

Notes: This method is called before the Open panel is shown but after its allowed file types are set to the cell's allowed types. At this time, you can further customize the Open panel as required. This method is called only when the style is set to `NSPathStylePopUp`. Implementation of this method is optional.

openPanel is a `NSOpenPanelMBS` object.

9.24.19 willPopUpMenu(menu as Variant)

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Implement this method to customize the menu of a pop-up-style path.

Notes: This event is called before the pop-up menu is shown. At this time, you can further customize the menu as required, adding and removing items. This method is called only when the style is set to `NSPathStylePopUp`. Implementation of this method is optional.

Menu is a `NSMenuMBS` object.

9.24.20 Constants

Path Style Constants

Constant	Value	Description
<code>NSPathStyleNavigationBar</code>	1	The navigation bar display style and behavior. Similar to the <code>NSPathStyleStandard</code> with the navigation bar drawing style. Also known as the breadcrumb style. Available in Mac OS X v10.5 and later.
<code>NSPathStylePopUp</code>	2	The pop-up display style and behavior. Only the last path component is displayed with an icon image and component name. The full path is shown when the user clicks on the cell. If the cell is editable, a Choose item is included to enable selecting a different path. Available in Mac OS X v10.5 and later.
<code>NSPathStyleStandard</code>	0	The standard display style and behavior. All path component cells are displayed with an icon image and component name. If the path can not fully be displayed, the middle parts are truncated as required. Available in Mac OS X v10.5 and later.

9.25 class NSPopUpButtonCellMBS

9.25.1 class NSPopUpButtonCellMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The NSPopUpButtonCell class defines the visual appearance of pop-up buttons that display pop-up or pull-down menus.

Notes: Pop-up menus present the user with a set of choices, much the way radio buttons do, but using much less space. Pull-down menus also provide a set of choices but present the information in a slightly different way, usually to provide a set of commands from which the user can choose.

The NSPopUpButtonCellMBS class implements the user interface for the NSPopUpButtonMBS class. Changes made to a menu (such as adding, removing, or changing the items) are not apparent while the menu is being displayed or interacted with.

Subclass of the NSMenuItemCellMBS class.

Blog Entries

- [MBS Xojo Plugins, version 17.1pr5](#)

9.25.2 Methods

9.25.3 addItemWithTitle(itemTitles() as string)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Adds multiple items to the end of the menu.

Notes: itemTitles: An array of strings containing the titles of the items you want to add. Each string in the array should be unique. If an item with the same title already exists in the menu, the existing item is removed and the new one is added.

The new menu items use the pop-up button,Äôs default action and target, but you can change these using the setAction: and setTarget: methods of the corresponding NSMenuItem object.

If you want to move an item, it,Äôs better to invoke removeItemWithTitle: explicitly and then call this method. After adding the items, this method uses the synchronizeTitleAndSelectedItem method to make sure the item being displayed matches the currently selected item.

Because this method searches for duplicate items, it should not be used if you are adding items to an already populated menu with more than a few hundred items. In a situation like this, add items directly to the receiver's menu instead.

9.25.4 addItemWithTitle(title as string)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Adds an item with the specified title to the end of the menu.

Notes: title: The title of the new menu item. If an item with the same title already exists in the menu, the existing item is removed and the new one is added.

The menu item uses the pop-up button's default action and target, but you can change these using the `setAction:` and `setTarget:` methods of the corresponding `NSMenuItemMBS` object.

Because this method searches for duplicate items, it should not be used if you are adding an item to an already populated menu with more than a few hundred items. In a situation like this, add items directly to the button's menu instead.

9.25.5 Constructor(image as NSImageMBS)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

See also:

- 9.25.6 Constructor(text as string, pullsDown as boolean)

601

9.25.6 Constructor(text as string, pullsDown as boolean)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Returns an `NSPopUpButtonCell` object initialized with the specified title.

See also:

- 9.25.5 Constructor(image as NSImageMBS)

601

9.25.7 dismissPopUp

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Dismisses the pop-up button's menu by ordering its window out.

Notes: If the pop-up button was not displaying its menu, this method does nothing.

You normally do not call this method explicitly. It is called by the Application Kit automatically to dismiss the menu for the pop-up button.

9.25.8 `indexOfItem(item as NSMenuItemMBS) as Integer`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the specified menu item.

Notes: item: The menu item whose index you want.

Returns the index of the item or -1 if no such item was found.

9.25.9 `indexOfItemWithTag(tag as Integer) as Integer`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the menu item with the specified tag.

Notes: tag: The tag of the menu item you want.

Returns the index of the item or -1 if no item with the specified tag was found.

Tags are values your application assigns to an object to identify it. You can assign tags to menu items using the tag property of NSMenuItemMBS.

9.25.10 `indexOfItemWithTitle(title as String) as Integer`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the item with the specified title.

Notes: title: The title of the item you want. You must not pass nil for this parameter.

Returns the index of the item or -1 if no item with the specified title was found.

9.25.11 `insertItemWithTitle(title as string, atIndex as Integer)`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Inserts an item at the specified position in the menu.

Notes: title: The title of the new item. If an item with the same title already exists in the menu, the existing item is removed and the new one is added

index: The zero-based index at which to insert the item. Specifying 0 inserts the item at the top of the menu.

The value in index must represent a valid position in the array. The menu item at index and all those that

follow it are shifted down one slot to make room for the new menu item.

This method assigns the pop-up button's default action and target to the new menu item. This triggers the action event of the NSActionCellMBS.

Because this method searches for duplicate items, it should not be used if you are adding an item to an already populated menu with more than a few hundred items. In a situation like this, add items directly to the button's menu instead.

9.25.12 itemArray as NSMenuItemMBS()

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: An array of NSMenuItemMBS objects that represent the items in the menu.

9.25.13 itemAtIndex(Index as Integer) as NSMenuItemMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Returns the menu item at the specified index.

Notes: index: The index of the item you want. The specified index must refer to an existing menu item.

Returns the menu item, or nil if no item exists at the specified index.

9.25.14 itemTitleAtIndex(Index as Integer) as String

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Returns the title of the item at the specified index.

Notes: index: The index of the item you want.

Returns the title of the item, or an empty string if no item exists at the specified index.

9.25.15 itemTitles as String()

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: An array of strings containing the titles of every item in the menu.

Notes: The titles appear in the order in which the items appear in the menu. If the menu contains separator items, the array contains an empty string ("") for each separator item.

9.25.16 `itemWithTitle(title as String)` as `NSMenuItemMBS`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Returns the menu item with the specified title.

Notes: title: The title of the menu item you want.

Returns the menu item, or nil if no item with the specified title exists in the menu.

9.25.17 `removeAllItems`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Removes all items in the receiver,Ãs item menu.

9.25.18 `removeItemAtIndex(Index as Integer)`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Removes the item at the specified index.

Notes: index: The zero-based index indicating which item to remove. Specifying 0 removes the item at the top of the menu. The index must be valid and non-negative.

9.25.19 `removeItemWithTitle(title as string)`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Removes the item with the specified title from the menu.

Notes: title: The title of the item you want to remove. If no menu item exists with the specified title, this method triggers an assertion.

9.25.20 `selectItem(item as NSMenuItemMBS)`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Selects the specified menu item.

Notes: item: The menu item to select, or nil if you want to deselect all menu items.

By default, selecting or deselecting a menu item from a pop-up menu changes its state. Selecting a menu item from a pull-down menu does not automatically alter the state of the item. To disassociate the current

selection from the state of menu items, set the `altersStateOfSelectedItem` property to NO.

9.25.21 `selectItemAtIndex(index as Integer)`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Selects the item in the menu at the specified index.

Notes: `index`: The index of the item you want to select, or -1 you want to deselect all menu items.

By default, selecting or deselecting a menu item from a pop-up menu changes its state. Selecting a menu item from a pull-down menu does not automatically alter the state of the item. To disassociate the current selection from the state of menu items, set the `altersStateOfSelectedItem` property to false. Subclassers can override this method to catch all select calls.

9.25.22 `selectItemWithTag(tag as Integer) as boolean`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Selects the menu item with the specified tag.

Notes: `tag`: The tag of the item you want to select.

Returns true if the item was successfully selected; otherwise, false.

If no item with the specified tag is found, this method returns false and leaves the menu state unchanged. You typically assign tags to menu items from Interface Builder, but you can also assign them programmatically using the `tag` property of `NSMenuItemMBS`.

9.25.23 `selectItemWithTitle(title as string)`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Selects the item with the specified title.

Notes: `title`: The title of the item to select. If you specify nil, an empty string, or a string that does not match the title of a menu item, this method deselects the currently selected item.

By default, selecting or deselecting a menu item changes its state. To disassociate the current selection from the state of menu items, set the `altersStateOfSelectedItem` property to false.

9.25.24 setTitle(title as string)

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Sets the string displayed in the receiver when the user isn't pressing the mouse button.

Notes: title: The string to display.

For pull-down menus that get their titles from a menu item, this method simply sets the pop-up button cell's menu item to the first item in the menu. For pop-up menus, if a menu item whose title matches aString exists, this method makes that menu item the current selection; otherwise, it creates a new menu item with the title aString, adds it to the pop-up menu, and selects it.

9.25.25 synchronizeTitleAndSelectedItem

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Synchronizes the the pop-up button's displayed item with the currently selected menu item.

Notes: If no item is currently selected, this method synchronizes the pop-up buttons displayed item with the first menu item. If the pop-up button cell does not get its displayed item from a menu item, this method does nothing.

For pull-down menus, this method sets the displayed item to the title first menu item.

If the pop-up button's menu does not contain any menu items, this method sets the pop-up button's displayed item to nil, resulting in nothing being displayed in the control.

9.25.26 Properties

9.25.27 altersStateOfSelectedItem as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates if the pop-up button links the state of the selected menu item to the current selection.

Notes: When the value of this property is true (which is the default value), the state of the selected item is set to NSOnState. When the value of this property is false, the items in the menu are left alone. When you change the value of this property, the state of the currently selected item is updated appropriately.

Note that this property affects only pop-up buttons (it is ignored for pull-down menus).

(Read and Write property)

9.25.28 arrowPosition as Integer

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The position of the arrow displayed on the button.

Notes: When the value of this property is `NSPopUpNoArrow`, the control displays no arrow. `NSPopUpArrowAtCenter` displays the arrow centered horizontally within the cell and `NSPopUpArrowAtBottom` displays the arrow at the edge of the cell. This property is used with `preferredEdge` to determine the exact location and orientation of the arrow.

This property applies to only bezel style and borderless pop-up buttons.
(Read and Write property)

9.25.29 autoenablesItems as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates if the button automatically enables and disables its items every time a user event occurs.

Notes: When the value of this property is true, the button automatically enables and disables items. The default value is true. For more information about enabling and disabling menu items, see `NSMenuValidation`.
(Read and Write property)

9.25.30 indexOfSelectedItem as Integer

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The index of the item last selected by the user.

Notes: The value of this property is the index of the selected item, or -1 if no item is selected.
(Read only property)

9.25.31 lastItem as NSMenuItemMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The last item in the menu.

Notes: (Read only property)

9.25.32 menu as NSMenuMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The pop-up button's associated menu.

Notes: (Read and Write property)

9.25.33 `numberOfItems` as Integer

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The number of items in the menu.

Notes: (Read only property)

9.25.34 `preferredEdge` as Integer

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The edge of the cell from which the menu should pop out when screen conditions are restrictive.

Notes: At display time, if attaching the menu to the preferred edge would cause part of the menu to be obscured, the pop-up button may use a different edge. If no preferred edge is set, the pop-up button uses the bottom edge by default, which is `NSMaxYEdge` for flipped views or `NSMinYEdge` for unflipped views. Additional values for this property include `NSMinXEdge` and `NSMaxXEdge`.

The exact location of the arrow is determined by examining the value of this property and `arrowPosition`.

If the arrow position is `NSPopUpArrowAtCenter`, the arrow stays in the center of the button and the value of this property determines which edge the arrow points to: `NSMinXEdge` points to the left, `NSMaxYEdge` points to the top, `NSMaxXEdge` points to the right, and `NSMinYEdge` points to the bottom.

If the arrow position is `NSPopUpArrowAtBottom`, the value of this property determines which edge at which the arrow is placed: `NSMinXEdge` places the arrow at the center of the left side, pointing to the left, `NSMinYEdge` places the arrow at bottom right corner, pointing up, `NSMaxXEdge` places the arrow at the center of the right side, pointing to the right, and `NSMaxYEdge` places the arrow at the bottom right corner, pointing down.

(Read and Write property)

9.25.35 `pullsDown` as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates the behavior of the button's menu.

Notes: When the value of this property is true, the menu behaves like a pull-down menu; when the value is false, it behaves like a pop-up menu. If you use this property to change the menu type from a pop-up menu to a pull-down menu, and the cell alters the state of its selected items, the state of the currently selected item is set to `NSOffState` before the menu type is changed.

(Read and Write property)

9.25.36 selectedItem as NSMenuItemMBS

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The menu item last selected by the user.

Notes: The value of this property is the menu item that is currently selected, or nil if no item is selected. The last selected menu item is the one that was highlighted when the user released the mouse button. It is possible for a pull-down menu,Ãs selected item to be its first item.
(Read only property)

9.25.37 titleOfSelectedItem as String

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: The title of the item last selected by the user.

Notes: The value of this property is the title of the selected menu item, or an empty string if no item is selected.
(Read only property)

9.25.38 usesItemFromMenu as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates if the control uses an item from the menu for its own title.

Notes: When the value of this property is true, a pull-down menu uses the title of the first menu item, and a pop-up menu uses the title of the currently selected menu (if no menu item is selected, the pop-up button displays no item and is drawn empty). When the value is false, the menu item set with menuItem (NSMenuItemMBS) is always displayed. The default value is true.
(Read and Write property)

9.25.39 Constants

Constants

Constant	Value	Description
NSPopUpArrowAtBottom	2	One of the arrowPosition constants. Arrow is drawn at the edge of the button, pointing toward the preferredEdge.
NSPopUpArrowAtCenter	1	One of the arrowPosition constants. Arrow is centered vertically, pointing toward the preferredEdge.
NSPopUpNoArrow	0	One of the arrowPosition constants. Does not display any arrow in the control.

9.26 class NSSearchFieldCellMBS

9.26.1 class NSSearchFieldCellMBS

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The programmatic interface for text fields that are used for text-based searches.

Notes: The NSSearchFieldCell class defines the programmatic interface for text fields that are optimized for text-based searches. An NSSearchFieldCell object is „wrapped,“ by an NSSearchFieldMBS control object, which directly inherits from the NSTextFieldMBS class. The search field implemented by these classes presents a standard user interface for searches, including a search button, a cancel button, and a pop-up icon menu for listing recent search strings and custom search categories.

When the user types and then pauses, the cell’s action message is sent to its target. You can query the cell’s string value for the current text to search for. Do not rely on the sender of the action to be an NSMenuMBS object because the menu may change. If you need to change the menu, modify the search menu template and update the value in the searchMenuTemplate property.

Subclass of the NSTextFieldCellMBS class.

Blog Entries

- [News from the MBS Xojo Plugins Version 22.3](#)
- [MBS Xojo Plugins in version 22.3](#)
- [MBS Xojo Plugins, version 22.3pr2](#)

9.26.2 Methods

9.26.3 Constructor(image as NSImageMBS)

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor not to use.

See also:

- 9.26.4 Constructor(text as string) 610

9.26.4 Constructor(text as string)

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The constructor.

See also:

- 9.26.3 Constructor(image as NSImageMBS) 610

9.26.5 recentSearches as String()

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: An array of the recent search strings to display in the pop-up icon menu of the search field.

Notes: This property contains an array of string objects, each of which contains a search string either displayed in the search menu or from a recent autosave archive. If there have been no recent searches and no prior searches saved under an autosave name, this array may be empty. When loading your interface, you might set the value of this property to a set of saved search strings.

9.26.6 resetCancelButtonCell

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: Resets the cancel button cell to its default attributes.

Notes: This method resets the target, action, regular image, and pressed image for the cancel button cell. By default, when users click the cancel button, the delete: action message is sent up the responder chain to the first NSText object that can handle it. This method gives you a way to customize the cancel button for specific situations and then reset the button defaults without having to undo changes individually.

9.26.7 resetSearchButtonCell

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: Resets the search button cell to its default attributes.

Notes: This method resets the target, action, regular image, and pressed image for the search button cell. By default, when users click the search button or press the Return key, the action defined for the receiver is sent to its designated target. This method gives you a way to customize the search button for specific situations and then reset the button defaults without having to undo changes individually.

9.26.8 setRecentSearches(RecentSearches() as String = nil)

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: An array of the recent search strings to display in the pop-up icon menu of the search field.

Notes: This property contains an array of string objects, each of which contains a search string either displayed in the search menu or from a recent autosave archive. If there have been no recent searches and no prior searches saved under an autosave name, this array may be empty. When loading your interface, you might set the value of this property to a set of saved search strings.

9.26.9 Properties

9.26.10 `cancelButtonCell` as `NSButtonCellMBS`

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The button cell used to display the cancel-button image.

Notes: (Read and Write property)

9.26.11 `maximumRecents` as `Integer`

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The maximum number of search strings that can appear in the search menu.

Notes: The value of this property must be between 0 and 254. Specifying a negative value for the property sets it to the default value, which is 10. Specifying a value greater than 254 sets the property to 254.

When the maximum number of search strings is exceeded, the oldest search string on the menu is dropped. (Read and Write property)

9.26.12 `recentsAutosaveName` as `String`

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The autosave name under which the search field automatically saves the list of recent search strings.

Notes: The autosave name is used as a key in the standard user defaults to save the recent searches. If you specify nil or an empty string for this parameter, no autosave name is set and searches are not automatically saved.

(Read and Write property)

9.26.13 `searchButtonCell` as `NSButtonCellMBS`

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The button cell used to display the search-button image.

Notes: (Read and Write property)

9.26.14 `searchMenuTemplate` as `NSMenuMBS`

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The menu object used to dynamically construct the search field,Äôs pop-up icon menu.

Notes: The cell looks for the tag constants described in Menu tags to determine how to populate the menu with items related to recent searches. For an example of how you might set up the search menu template, see *Configuring a Search Menu*:

https://developer.apple.com/library/archive/documentation/Cocoa/Conceptual/SearchFields/Articles/MenuTemplate.html#//apple_ref/doc/uid/20002245

(Read and Write property)

9.26.15 sendsSearchStringImmediately as Boolean

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the cell calls its action method immediately when an appropriate action occurs.

Notes: When the value of this property is true, the cell calls its action method immediately upon notification of any changes to the search field. When the value is NO, the cell pauses briefly after receiving a notification and then calls its action method. Pausing gives the user an opportunity to type more text into the search field and minimize the number of searches that are performed.

(Read and Write property)

9.26.16 sendsWholeSearchString as Boolean

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the cell calls its search action method when the user clicks the search button (or presses Return) or after each keystroke.

Notes: When the value of this property is true, the cell calls its action method when the user clicks the search button or presses Return. When the value is false, the cell calls the action method after each keystroke. The default value of this property is false.

(Read and Write property)

9.26.17 Constants

Menu Tags

Constant	Value	Description
ClearRecentsMenuItemTag	1002	The menu item for clearing the current set of recent string searches in the menu.
NoRecentsMenuItemTag	1003	The menu item that describes a lack of recent search strings.
RecentsMenuItemTag	1001	The location of recent search strings in the ,Äúrecents,Äù menu group.
RecentsTitleMenuItemTag	1000	The menu item that provides the title of the menu group for recent search strings.

9.27 control NSSearchFieldControlMBS

9.27.1 control NSSearchFieldControlMBS

Plugin Version: 13.4, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a NSSearchField.

Notes: This control embeds a special NSSearchField subclass.

Designed for Xojo 2013r1 and newer. May work on Xojo 2012, but not perfectly.

Please use view property to access the underlying object and set properties.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 24.0](#)
- [MBS Xojo Plugins, version 23.6pr1](#)
- [MBS Xojo Plugins, version 23.2pr1](#)
- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo Plugins, version 21.5pr1](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr8](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr2](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr10](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr7](#)

9.27.2 Properties

9.27.3 View as NSSearchFieldMBS

Plugin Version: 13.4, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.

Notes: Use this object to set more options on the control.

(Read only property)

9.27.4 Events

9.27.5 Action

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: The action event run, when e.g. return key is pressed.

9.27.6 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.27.7 Close

Plugin Version: 13.4, Platform: macOS, Targets: .

Function:

The control is about to close.

In Xojo version 2021r3 and newer this event is named Closing.

9.27.8 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

9.27.9 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Called when a menuitem is choosen.

Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

9.27.10 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.27.11 EnableMenuItems

Plugin Version: 17.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In Xojo version 2021r3 and newer this event is named MenuBarSelected.

9.27.12 FrameChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.27.13 GotFocus

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

Function:

The control itself got focus.

In Xojo version 2021r3 and newer this event is named FocusReceived.

Notes:

This only fires if the control itself got focus and not a sub control.

9.27.14 LostFocus

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

Function:

The control lost focus.

In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.27.15MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Ãs region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle theMouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles theMouseDown so the above event handlers do not get called.

9.27.16 MouseDrag(x as Integer, y as Integer)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of times per second), it is your responsibility to determine if the mouse has really moved.

9.27.17 MouseUp(x as Integer, y as Integer)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.27.18 Open

Plugin Version: 13.4, Platform: macOS, Targets: .

Function:

The control is about to be created and you can initialize it.

In Xojo version 2021r3 and newer this event is named Opening.

9.27.19 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.27.20 `TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)`

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

Function: Sent when a control with editable text begins an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidBeginEditingNotification`.

This event is invoked when the user begins editing text in a control such as a text field or a form field. The control posts a `NSControlTextDidBeginEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also delivered for inspection.

9.27.21 `TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)`

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

Function: Sent when the text in the receiving control changes.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidChangeNotification`.

This event is invoked when text in a control such as a text field or form changes. The control posts a `NSControlTextDidChangeNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is provided as parameter for inspection.

9.27.22 `TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)`

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

Function: Sent when a control with editable text ends an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidEndEditingNotification`.

This event is invoked when the user stops editing text in a control such as a text field or form. The control posts a `NSControlTextDidEndEditingNotification` notification, and if the control's subclass implements this

event, it is automatically registered to receive the notification. The field editor is also provided for inspection.

9.27.23 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean

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

Function: The event called to decide whether text editing should be allowed.

Notes: Return true to allow text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.27.24 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean

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

Function: The event called to decide whether ending text editing should be allowed.

Notes: Return true to allow end of text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.27.25 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.28 class NSSearchFieldMBS

9.28.1 class NSSearchFieldMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: An NSSearchField object implements a text field control that is optimized for performing text-based searches.

Example:

```
// create searchfield
dim n as new NSSearchFieldMBS(0,0,100,20)

// set placeholder
dim x as NSTextFieldCellMBS = n.cell
x.placeholderString = "Test"
```

Notes: The control provides a customized text field for entering search data, a search button, a cancel button, and a pop-up icon menu for listing recent search strings and custom search categories.

An NSSearchField object wraps an NSSearchFieldCell object. Access to most search field attributes occurs through the cell, which provides a more comprehensive programmatic interface for manipulating the search field. You can use an NSSearchField object though to manipulate some aspects of the search field. For additional information about search fields and how to manipulate them, see the NSSearchFieldCell class.

You can embed this view in a CustomNSViewMBS to get more events for mouse and keyboard. Subclass of the NSTextFieldMBS class.

Blog Entries

- [MBS Xojo Plugins, version 23.6pr1](#)
- [Cocoa SearchField with menu](#)
- [MBS Real Studio Plugins, version 11.2pr8](#)
- [MBS Real Studio Plugins, version 11.2pr3](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS REALbasic Plugins, version 11.1pr3](#)

9.28.2 Methods

9.28.3 Constructor

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new search field with size 100/100 and position 0/0

Example:

```
dim t as new NSSearchFieldMBS
```

Notes: On success the handle property is not zero.

See also:

- 9.28.4 Constructor(Handle as Integer) 621
- 9.28.5 Constructor(left as Double, top as Double, width as Double, height as Double) 621

9.28.4 Constructor(Handle as Integer)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSSearchField handle.

Example:

```
dim t as new NSSearchFieldMBS(0, 0, 100, 100)
```

```
dim v as new NSSearchFieldMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSSearchField and the plugin retains this handle.

See also:

- 9.28.3 Constructor 620
- 9.28.5 Constructor(left as Double, top as Double, width as Double, height as Double) 621

9.28.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new search field with the given size and position.

Example:

```
dim x as new NSSearchFieldMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 9.28.3 Constructor 620
- 9.28.4 Constructor(Handle as Integer) 621

9.28.6 recentSearches as string()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the list of recent search strings for the control.

Notes: An array of strings, each of which contains a search string either displayed in the search menu or from a recent autosave archive. If there have been no recent searches and no prior searches saved under an autosave name, this array may be empty.

9.28.7 setRecentSearches(values() as string)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Sets the list of recent search strings to list in the pop-up icon menu of the receiver.

Notes: You might use this method to set the recent list of searches from an archived copy.

9.28.8 Properties

9.28.9 Scrollable as Boolean

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver scrolls excess text past the cell's bounds.

Notes: (Read and Write property)

9.28.10 maximumRecents as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The maximum number of search strings that can appear in the search menu.

Notes: The maximum number of search strings that can appear in the menu. This value can be between 0 and 254. Specifying a value less than 0 sets the value to the default, which is 10. Specifying a value greater than 254 sets the maximum to 254.

When the limit is exceeded, the oldest search string on the menu is dropped.
(Read and Write computed property)

9.28.11 recentsAutosaveName as string

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The key under which the prior list of recent search strings has been archived.

Notes: The autosave name, which is used as a key in the standard user defaults to save the recent searches. The default value is "", which causes searches not to be autosaved.
(Read and Write computed property)

9.28.12 searchMenuTemplate as NSMenuMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The menu template object used to dynamically construct the search pop-up icon menu.

Notes: The receiver looks for the tag constants described in Menu tags to determine how to populate the menu with items related to recent searches. (See constants)

To modify the actual menu shown, please use NSMenuItemMBS.validateMenuItem event. There you can for example set the state of the menu item shown. The SearchField makes a copy of the NSMenuItem, so the menuItem where the event is called, is not the one shown. It's the one passes as parameter.
(Read and Write computed property)

9.28.13 sendsSearchStringImmediately as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Whether the cell sends its action message to the target immediately upon notification of any changes to the search field text or after a brief pause.

Notes: True to send the cell's action immediately upon notification of any changes to the search field; otherwise, false if you want the cell to pause briefly before sending its action message. Pausing gives the user the opportunity to type more text into the search field before initiating the search.
(Read and Write computed property)

9.28.14 sendsWholeSearchString as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver sends the search action message when the user clicks the search button (or presses return) or after each keystroke.

Notes: True to send the action message all at once when the user clicks the search button or presses return; otherwise, False to send the search string after each keystroke.
(Read and Write computed property)

9.28.15 Constants

Search menu template constants

Constant	Value	Description
<code>NSSearchFieldClearRecentsMenuItemTag</code>	1002	Identifies the menu item for clearing the current set of recent strings in the menu. This item is hidden if there are no recent strings.
<code>NSSearchFieldNoRecentsMenuItemTag</code>	1003	Identifies the menu item that describes a lack of recent search strings (for example, "No recent searches"). This item is hidden if there have been recent searches.
<code>NSSearchFieldRecentsMenuItemTag</code>	1001	Identifies where recent search strings should appear in the "recent" group.
<code>NSSearchFieldRecentsTitleMenuItemTag</code>	1000	Identifies the menu item that is the title of the menu group for recent strings. This item is hidden if there are no recent strings. You may use this tagged item for separator characters that also do not display if there are no recent strings to display.

9.29 control NSSecureTextFieldControlMBS

9.29.1 control NSSecureTextFieldControlMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a NSSecureTextField.

Notes: This control embeds a special NSSecureTextField subclass.

Designed for Xojo 2013r1 and newer. May work on Xojo 2012, but not perfectly.

Please use view property to access the underlying object and set properties.

Blog Entries

- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr8](#)

9.29.2 Properties

9.29.3 echosBullets as Boolean

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver echoes a bullet character rather than each character typed.

Example:

```
dim t as NSSecureTextFieldControlMBS // your textfield
t.echosBullets = true
```

Notes: If true, bullets are echoed. If false, the cursor is moved for each character typed, but nothing is displayed.

(Read and Write property)

9.29.4 View as NSSecureTextFieldMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.

Notes: Use this object to set more options on the control.

(Read only property)

9.29.5 Events

9.29.6 Action

Plugin Version: 15.0, Platform: macOS, Targets: .

Function: The control's action was triggered.

Notes: The text changed.

9.29.7 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.29.8 Close

Plugin Version: 15.0, Platform: macOS, Targets: .

Function:

The control is about to close.

In Xojo version 2021r3 and newer this event is named Closing.

9.29.9 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

9.29.10 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Called when a menuitem is chosen.

Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

9.29.11 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.29.12 EnableMenuItems

Plugin Version: 17.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In Xojo version 2021r3 and newer this event is named MenuBarSelected.

9.29.13 FrameChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.29.14 GotFocus

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

Function:

The control itself got focus.

In Xojo version 2021r3 and newer this event is named FocusReceived.

Notes:

This only fires if the control itself got focus and not a sub control.

9.29.15 LostFocus

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

Function:

The control lost focus.

In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.29.16 `MouseDown(x as Integer, y as Integer, Modifiers as Integer)` As Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control's region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.29.17 `MouseDrag(x as Integer, y as Integer)`

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of times per second), it is your responsibility to determine if the mouse has really moved.

9.29.18 `MouseUp(x as Integer, y as Integer)`

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.29.19 `Open`

Plugin Version: 15.0, Platform: macOS, Targets: .

Function:

The control is about to be created and you can initialize it.
In Xojo version 2021r3 and newer this event is named Opening.

9.29.20 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.29.21 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 15.0, Platform: macOS, Targets: .

Function: Sent when a control with editable text begins an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidBeginEditingNotification`.

This event is invoked when the user begins editing text in a control such as a text field or a form field. The control posts a `NSControlTextDidBeginEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also delivered for inspection.

9.29.22 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

Plugin Version: 15.0, Platform: macOS, Targets: .

Function: Sent when the text in the receiving control changes.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidChangeNotification`.

This event is invoked when text in a control such as a text field or form changes. The control posts a `NSControlTextDidChangeNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is provided as parameter for inspection.

9.29.23 `TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)`

Plugin Version: 15.0, Platform: macOS, Targets: .

Function: Sent when a control with editable text ends an editing session.

Notes: Notification: The notification object. The name of the notification is always `NSControlTextDidEndEditingNotification`.

This event is invoked when the user stops editing text in a control such as a text field or form. The control posts a `NSControlTextDidEndEditingNotification` notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also provided for inspection.

9.29.24 `textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean`

Plugin Version: 15.0, Platform: macOS, Targets: .

Function: The event called to decide whether text editing should be allowed.

Notes: Return true to allow text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.29.25 `textShouldEndEditing(fieldEditor as NSTextMBS) as boolean`

Plugin Version: 15.0, Platform: macOS, Targets: .

Function: The event called to decide whether ending text editing should be allowed.

Notes: Return true to allow end of text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.29.26 `willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)`

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.30 class NSSecureTextFieldMBS

9.30.1 class NSSecureTextFieldMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: NSSecureTextField is a subclass of NSTextField that hides its text from display or other access via the user interface.

Notes: It's suitable for use as a password-entry object or for any item in which a secure value must be kept.

NSSecureTextField uses NSSecureTextFieldCell to implement its user interface.

Subclass of the NSTextFieldMBS class.

Blog Entries

- [MBS Plugins 11.1 Release notes](#)
- [MBS REALbasic Plugins, version 11.1pr3](#)

9.30.2 Methods

9.30.3 Constructor

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new secure text field with size 100/100 and position 0/0

Example:

```
dim t as new NSSecureTextFieldMBS
```

Notes: On success the handle property is not zero.

See also:

- 9.30.4 Constructor(Handle as Integer) 631
- 9.30.5 Constructor(left as Double, top as Double, width as Double, height as Double) 632

9.30.4 Constructor(Handle as Integer)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSSecureTextField handle.

Example:

```
dim t as new NSSecureTextFieldMBS(0, 0, 100, 100)
dim v as new NSSecureTextFieldMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSSecureTextField and the plugin retains this handle.

See also:

- 9.30.3 Constructor 631
- 9.30.5 Constructor(left as Double, top as Double, width as Double, height as Double) 632

9.30.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new secure text field with the given size and position.

Example:

```
dim x as new NSSecureTextFieldMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 9.30.3 Constructor 631
- 9.30.4 Constructor(Handle as Integer) 631

9.30.6 Properties

9.30.7 echosBullets as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver echoes a bullet character rather than each character typed.

Example:

```
dim t as NSSecureTextFieldMBS // your textfield
t.echosBullets = true
```

Notes: If true, bullets are echoed. If false, the cursor is moved for each character typed, but nothing is displayed.

(Read and Write computed property)

9.31 control NSSegmentedControlControlMBS

9.31.1 control NSSegmentedControlControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The Xojo control to host a NSSegmentedControl.

Notes: We know the fact, that the control wrapping the segmented control has control twice in the name. Suggestions to rename are welcome, but unlikely to happen as it would break code.

Blog Entries

- [News from the MBS Xojo Plugins Version 24.0](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 24.0](#)
- [NSSegmentedControl and NSPathControl](#)
- [MBS Xojo Plugins, version 24.0pr7](#)

9.31.2 Properties

9.31.3 View as NSSegmentedControlMBS

Plugin Version: 24.0, Platform: macOS, Targets: Desktop only.

Function: The reference to the underlying NSSegmentedControlMBS.

Notes: (Read only property)

9.31.4 Events

9.31.5 Action

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event is called when user clicks on a date/time and changes something.

9.31.6 BoundsChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.31.7 Close

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control is about to close.

In Xojo version 2021r3 and newer this event is named Closing.

9.31.8 ConstructContextMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

9.31.9 ContextualMenuItem(hitItem as MenuItem) as Boolean

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: Called when a menuitem is chosen.

Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

9.31.10 didCloseContextMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextMenu event.

9.31.11 EnableMenuItems

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In Xojo version 2021r3 and newer this event is named MenuBarSelected.

9.31.12 FrameChanged

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.31.13 GotFocus

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control itself got focus.

In Xojo version 2021r3 and newer this event is named FocusReceived.

Notes:

This only fires if the control itself got focus and not a sub control.

9.31.14 LostFocus

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control lost focus.

In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.31.15MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control's region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return `False`, the system handles the `MouseDown` so the above event handlers do not get called.

9.31.16 `MouseDown(x as Integer, y as Integer)`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to `x`, `y`.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.31.17 `MouseUp(x As Integer, y As Integer)`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the `x` and `y` parameters to determine if the mouse button was released within the control's boundaries.

9.31.18 `Open`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function:

The control is about to be created and you can initialize it.

In Xojo version 2021r3 and newer this event is named `Opening`.

9.31.19 `ScaleFactorChanged(NewFactor as double)`

Plugin Version: 24.0, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.31.20 `willShowContextMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)`

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.32 class NSSegmentedControlMBS

9.32.1 class NSSegmentedControlMBS

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The plugin class for a Cocoa NSSegmentedControl.

Notes: An NSSegmentedControl object implements a horizontal button made of multiple segments.

The NSSegmentedControl class uses an NSSegmentedCell class to implement much of the control's functionality. Most methods in NSSegmentedControl are simply "cover methods" that call the corresponding method in NSSegmentedCell. The methods of NSSegmentedCell that do not have covers relate to accessing and setting values for tags and tool tips; programatically setting the key segment; and establishing the mode of the control.

The features of a segmented control include:

- Each segment can have an image, text (label), menu, tooltip, and tag
- Either the whole control or individual segments can be enabled or disabled
- There are three tracking modes for segments: select one mode (also known as radio button mode and illustrated by Finder's view mode selection control), momentary mode (as illustrated by Safari's toolbar buttons), or select any mode (where any combination of buttons may be on or off)
- Each segment can be either a fixed width or autosized to fit the contents
- If a segment has text and is marked as autosizing, then the text may be truncated so that the control completely fits
- If an image is too large to fit in a segment, it is clipped
- Full keyboard control of the user interface

Subclass of the NSControlMBS class.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr4](#)
- [NSSegmentedControl and NSPathControl](#)
- [MBS Xojo / Real Studio Plugins, version 15.2pr4](#)
- [MBS Xojo / Real Studio Plugins, version 15.2pr3](#)
- [MonkeyBread Software releases MBS Real Studio plug-ins in version 12.1](#)
- [MBS Real Studio Plugins, version 12.1pr10](#)

Xojo Developer Magazine

- [10.4, page 9: News](#)

9.32.2 Methods

9.32.3 Constructor

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new path control with size 100/100 and position 0/0

Example:

```
dim t as new NSSegmentedControlMBS
```

Notes: On success the handle property is not zero.

See also:

- [9.32.4 Constructor\(Handle as Integer\)](#) 639
- [9.32.5 Constructor\(left as Double, top as Double, width as Double, height as Double\)](#) 640

9.32.4 Constructor(Handle as Integer)

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSPathControl handle.

Example:

```
dim t as new NSSegmentedControlMBS(0, 0, 100, 100)
dim v as new NSSegmentedControlMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSSegmentedControl and the plugin retains this handle.

See also:

- [9.32.3 Constructor](#) 639
- [9.32.5 Constructor\(left as Double, top as Double, width as Double, height as Double\)](#) 640

9.32.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new path control with the given size and position.

Example:

```
dim x as new NSSegmentedControlMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 9.32.3 Constructor 639
- 9.32.4 Constructor(Handle as Integer) 639

9.32.6 makeNextSegmentKey

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Selects the next segment.

Notes: The next segment is the one to the right of the currently selected segment. For the last segment, the selection wraps back to the beginning of the control.

9.32.7 makePreviousSegmentKey

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Selects the previous segment.

Notes: The previous segment is the one to the left of the currently selected segment. For the first segment, the selection wraps around to the last segment of the control.

9.32.8 selectSegmentWithTag(Tag as Integer) as Boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Selects the segment with the specified tag.

Notes: tag: The tag associated with the desired segment.

True if the segment was selected successfully; otherwise, false.

Typically, you use Interface Builder to specify the tag for each segment. You may also set this value programmatically using the `setTag:forSegment:` method of `NSSegmentedCell`.

9.32.9 Properties

9.32.10 cellTrackingMode as Integer

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: The tracking mode used for the segments of the receiver.

Notes: Possible values for trackingMode are described in NSSegmentSwitchTracking. The default value is NSSegmentSwitchTrackingSelectOne.

This property was named only trackingMode in 15.1 and older plugins.
(Read and Write property)

9.32.11 doubleValueForSelectedSegment as Double

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: Query the double value for selected segment.

Notes: This message is valid only for trackingMode = NSSegmentSwitchTrackingMomentaryAccelerator and provides the double value for the selected segment.

Available on Mac OS X 10.10.3.

(Read only property)

9.32.12 segmentCount as Integer

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Returns the number of segments in the receiver.

Notes: (Read and Write property)

9.32.13 segmentStyle as Integer

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The visual style used to display the receiver.

Notes: See style constants.

Available in Mac OS X v10.5 and later.

(Read and Write property)

9.32.14 `selectedSegment` as Integer

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The index of the selected segment of the receiver.

Notes: The index of the currently selected segment or -1 if no segment is selected. If the receiver allows multiple selections, this method returns the most recently selected segment.

(Read and Write property)

9.32.15 `springLoaded` as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: Sends action on deep-press or extended hover while dragging.

Notes: Defaults to false.

(Read and Write property)

9.32.16 `trackingMode` as Integer

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: The type of tracking behavior the control exhibits.

Notes: An `NSSegmentSwitchTracking` value specifies how the control responds when the user presses a keyboard key or clicks, force clicks (applies pressure in a pressure-sensitive system), releases pressure, and so on.

see `NSSegmentSwitchTracking*` constants.

Available on Mac OS X 10.10.3 and newer.

(Read and Write property)

9.32.17 `imageForSegment(segment as Integer)` as `NSImageMBS`

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The image for the specified segment.

Notes: `image`: The image to apply to the segment or nil if you want to clear the existing image. Images are not scaled to fit inside a segment. If the image is larger than the available space, it is clipped.

`segment`: The index of the segment whose image you want to set. This method raises an `NSRangeException` if the index is out of bounds.

(Read and Write computed property)

9.32.18 imageScalingForSegment(segment as Integer) as Integer

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The scaling mode used to display the specified segment's image.

Notes: scaling: One of the image scaling constants. For a list of possible values, see constants.

segment: The index of the segment whose enabled state you want to get. This method raises an NSRangeException if the index is out of bounds.

Available in Mac OS X v10.5 and later.

(Read and Write computed property)

9.32.19 isEnabledForSegment(segment as Integer) as Boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The enabled state of the specified segment.

Notes: True to enable the segment; otherwise, false to disable it.

segment: The index of the segment you want to enable or disable. This method raises an NSRangeException if the index is out of bounds.

(Read and Write computed property)

9.32.20 isSelectedForSegment(segment as Integer) as Boolean

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The selection state of the specified segment.

Notes: True if you want to select the segment; otherwise, false.

segment: The index of the segment whose selection state you want to set. This method raises an NSRangeException if the index is out of bounds.

If the receiver allows only a single selection, this method deselects any other selected segments.

(Read and Write computed property)

9.32.21 labelForSegment(segment as Integer) as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The label for the specified segment.

Notes: label: The label you want to display in the segment. If the width of the string is greater than the width of the segment, the string's text is truncated during drawing.

segment: The index of the segment whose label you want to set. This method raises an NSRangeException

if the index is out of bounds.
(Read and Write computed property)

9.32.22 menuForSegment(segment as Integer) as NSMenuMBS

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The menu for the specified segment.

Notes: menu: The menu you want to add to the segment or nil to clear the current menu. This menu is displayed when the user clicks and holds the mouse button while the mouse is over the segment.

segment: The index of the segment whose menu you want to set. This method raises an NSRangeException if the index is out of bounds.

Adding a menu to a segment allows that segment to be used as a pop-up button.
(Read and Write computed property)

9.32.23 tagForSegment(segment as Integer) as Integer

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The tag value for the segment.

Notes: segment: The index of the segment whose width you want to get. This method raises an NSRangeException if the index is out of bounds.

The tag is an integer you define to identify your items.
(Read and Write computed property)

9.32.24 ToolTipForSegment(segment as Integer) as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The tool tip for the specified segment.

Notes: segment: The index of the segment whose tool tip you want to set. This method raises an NSRangeException if the index is out of bounds.

Tool tips are currently not displayed. Apple may change that in the future.
(Read and Write computed property)

9.32.25 widthForSegment(segment as Integer) as Double

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The width of the specified segment.

Notes: width: The width of the segment, measured in points. Specify the value 0 if you want the segment to be sized to fit the available space automatically.

segment: The index of the segment whose width you want to set. This method raises an NSRangeException if the index is out of bounds.

(Read and Write computed property)

9.32.26 Constants

Scaling Modes

Constant	Value	Description
NSImageScaleAxesIndependently	1	Scale each dimension to exactly fit destination. This setting does not preserve the aspect ratio of the image. Available in Mac OS X v10.5 and later.
NSImageScaleNone	2	Do not scale the image. Available in Mac OS X v10.5 and later.
NSImageScaleProportionallyDown	0	If it is too large for the destination, scale the image down while preserving aspect ratio. Available in Mac OS X v10.5 and later.
NSImageScaleProportionallyUpOrDown	3	Scale the image to its maximum possible dimensions while both staying within the destination area and preserving its aspect ratio. Available in Mac OS X v10.5 and later.

Style Constants

Constant	Value	Description
NSSegmentStyleAutomatic	0	The appearance of the segmented control is automatically determined based on the type of window in which the control is displayed and the position within the window. Available in Mac OS X v10.5 and later.
NSSegmentStyleRounded	1	The control is displayed using the rounded style. Available in Mac OS X v10.5 and later.
NSSegmentStyleRoundRect	2	The control is displayed using the round rect style. Available in Mac OS X v10.5 and later.
NSSegmentStyleSmallSquare	6	The control is displayed using the small square style. Available in Mac OS X v10.5 and later.
NSSegmentStyleTexturedSquare	4	The control is displayed using the textured square style. Available in Mac OS X v10.5 and later.

Constants for the Switch Tracking

Constant	Value	Description
<code>NSSegmentSwitchTrackingMomentary</code>	2	A segment is selected only when tracking.
<code>NSSegmentSwitchTrackingMomentaryAccelerator</code>	3	accelerator behavior, only selected while tracking.
<code>NSSegmentSwitchTrackingSelectAny</code>	1	Any segment can be selected.
<code>NSSegmentSwitchTrackingSelectOne</code>	0	Only one segment may be selected.

9.33 control NSSplitViewControlMBS

9.33.1 control NSSplitViewControlMBS

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Our control to wrap a NSSplitViewMBS object.

Blog Entries

- [MBS Xojo Plugins, version 23.5pr7](#)
- [News from the MBS Xojo Plugins Version 23.4](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.4](#)
- [NSSplitView Control for Xojo](#)
- [MBS Xojo Plugins, version 23.4pr5](#)

9.33.2 Properties

9.33.3 View as NSSplitViewMBS

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: The reference to the underlying NSSplitView.

Notes: (Read only property)

9.33.4 Events

9.33.5 BoundsChanged

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.33.6 canCollapseSubview(subview as NSViewMBS) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to determine whether the user can collapse and expand the specified subview.

Notes: subview: The subview to collapse.

Returns true if subview collapses when the user drags a divider beyond the halfway mark between its minimum size and its edge; otherwise, false.

The subview expands when the user drags the divider beyond the halfway mark between its minimum size and its edge.

To specify the minimum size, define the `constrainMaxCoordinate` and `constrainMinCoordinate` events. A subview can collapse only if you also define `constrainMinCoordinate`.

A collapsed subview isn't visible, but the split view object retains it with the same size as before the collapse.

If the events are implemented, the subviews can collapse.

9.33.7 Close

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The control is about to close.

In Xojo version 2021r3 and newer this event is named `Closing`.

9.33.8 `constrainMaxCoordinate(proposedMaximumPosition as double, dividerIndex as Integer) as Double`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to constrain the maximum coordinate limit of a divider when the user drags it.

Notes: `proposedMax`: The proposed maximum coordinate limit of the subview in the split view, flipped coordinate system.

`dividerIndex`: Specifies the divider the user is moving, with the first divider being 0 and increasing from top to bottom (or left to right).

Returns the maximum coordinate limit of the divider.

The delegate receives this message before the split view begins tracking the mouse to position a divider. You can further constrain the limits, but you can't extend the divider limits.

If the split bars are horizontal (views are one on top of the other), `proposedMax` is the bottom limit. If the split bars are vertical (views are side by side), `proposedMax` is the right limit. The initial value of `proposedMax` is the bottom (or right side) of the subview after the divider.

9.33.9 constrainMinCoordinate(proposedMinimumPosition as double, dividerIndex as Integer) as Double

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to constrain the minimum coordinate limit of a divider when the user drags it.

Notes: proposedMin: The proposed minimum coordinate limit of the subview in the split view,Ãs flipped coordinate system.

dividerIndex: Specifies the divider the user is moving, with the first divider being 0 and increasing from top to bottom (or left to right).

Returns the minimum coordinate limit of the divider.

The delegate receives this message before the split view begins tracking the cursor to position a divider. You can further constrain the limits, but you can,Ãt extend the divider limits.

If the split bars are horizontal (views are one on top of the other), proposedMin is the top limit. If the split bars are vertical (views are side by side), proposedMin is the left limit. The initial value of proposedMin is the top (or left side) of the subview before the divider.

9.33.10 constrainSplitPosition(proposedPosition as double, dividerIndex as Integer) as Double

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to constrain the divider to certain positions.

Notes: proposedPosition: The cursor,Ãs current position, and the proposed position of the divider.

dividerIndex: The index of the divider the user is moving, with the first divider being 0 and increasing from top to bottom (or left to right).

Returns the position for constraining the divider.

If the delegate implements this method, the split view calls it repeatedly as the user moves the divider.

If a subview,Ãs height must be a multiple of a certain number, use this method to return the multiple nearest to proposedPosition.

9.33.11 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

9.33.12 ContextualMenuItem(hitItem as MenuItem) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Called when a menuitem is chosen.

Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

9.33.13 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.33.14 drawDivider(graphics as NSGraphicsMBS, Rect as NSRectMBS) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Draws a divider between two of the split view's subviews.

Example:

Function drawDivider(graphics as NSGraphicsMBS, Rect as NSRectMBS) **Handles** drawDivider as Boolean

```
// draw a red box with blue border
graphics.setFillColor NSColorMBS.redColor
graphics.setStrokeColor NSColorMBS.blueColor
```

```
Dim r As New NSRectMBS(rect.X, rect.y + rect.Height/2 - 5, 10, 10)
graphics.fillRect r
graphics.strokeRect r
```

```
Return True
End Function
```

Notes: Rect: The entire divider rectangle in the split view's flipped coordinate system.

If you override this method to use a custom style for the divider, you may need to change the size of the divider.

9.33.15 EnableMenuItems

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In Xojo version 2021r3 and newer this event is named MenuBarSelected.

9.33.16 FrameChanged

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.33.17 GotFocus

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The control itself got focus.

In Xojo version 2021r3 and newer this event is named FocusReceived.

Notes:

This only fires if the control itself got focus and not a sub control.

9.33.18 LostFocus

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The control lost focus.

In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.33.19 `MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control's region at the location passed in to `x`, `y`.

Notes: The coordinates `x` and `y` are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return `True` if you are going to handle the `MouseDown`. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the `MouseDown` and `MouseUp` events.

If you return `False`, the system handles the `MouseDown` so the above event handlers do not get called.

9.33.20 `MouseDown(x as Integer, y as Integer)`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to `x`, `y`.

As this event is fired continuously (hundreds of times per second), it is your responsibility to determine if the mouse has really moved.

9.33.21 `MouseUp(x As Integer, y As Integer)`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the `x` and `y` parameters to determine if the mouse button was released within the control's boundaries.

9.33.22 `Open`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function:

The control is about to be created and you can initialize it.

In Xojo version 2021r3 and newer this event is named `Opening`.

9.33.23 resizeSubviewsWithOldSize(oldSize as NSSizeMBS)

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to specify custom sizing behavior for the subviews of the split view.

Notes: oldSize: The size of the split view before the user resizes it.

If the delegate implements this method, it receives this message after the split view resizes.

Resize the subviews so that the sum of the sizes of the subviews plus the sum of the thickness of the dividers equals the size of the new frame of the NSSplitView. You can get the thickness of a divider through the dividerThickness method.

If you implement this delegate method to resize subviews on your own, the NSSplitViewMBS doesn't perform any error checking for you. However, you can invoke adjustSubviews() to perform the default sizing behavior.

9.33.24 ScaleFactorChanged(NewFactor as double)

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.33.25 shouldAdjustSizeOfSubview(view as NSViewMBS) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to specify whether to resize the subview.

Notes: subview: The subview to resize.

If adjustSubviews() can change the size of the subview, return true; otherwise, false. By returning false, you lock the size of the split view subview while the split view resizes.

Regardless of the value that this method returns, adjustSubviews() may change the origin of the subview. Nonresizable subviews may resize to prevent an invalid subview layout.

If a split view has no delegate, or if its delegate doesn't respond to this message, the split view behaves as if this method returns true.

9.33.26 `shouldHideDivider(dividerIndex as Integer) as Boolean`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Allows the delegate to determine whether the user can drag a divider or adjust it off the edge of the split view.

Notes: `dividerIndex`: The zero-based index of the divider.

Return true if the user can drag the divider off the edge of the split view, resulting in it not being visible.

If a split view has no delegate, or if its delegate doesn't respond to this message, the split view behaves as if it has a delegate that returns false when it receives this message.

9.33.27 `splitViewDidResizeSubviews`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Notifies the delegate when the split view resizes its subviews.

Notes: The control invokes this event after the split view resizes two of its subviews in response to the repositioning of a divider.

9.33.28 `splitViewWillResizeSubviews`

Plugin Version: 23.4, Platform: macOS, Targets: .

Function: Notifies the delegate when the split view is about to resize its subviews.

Notes: The control invokes this method before the split view resizes two of its subviews in response to the repositioning of a divider.

9.33.29 `willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)`

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.34 class NSSplitViewMBS

9.34.1 class NSSplitViewMBS

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: A view that arranges two or more views in a linear stack running horizontally or vertically.

Example:

```
Sub Opening()
// add sub views
Dim view As NSSplitViewMBS = Me.View
```

```
view.Vertical = True
view.addArrangedSubview TextArea1.NSViewMBS
view.addArrangedSubview TextArea2.NSViewMBS
```

End Sub

Notes: By default, dividers have a horizontal orientation so that the split view arranges its panes vertically from top to bottom.

Divider indices are zero-based. If the Vertical property is false, which is the default value, the top divider has an index of 0. If isVertical is true, the leading divider has an index of 0.

Subclass of the NSViewMBS class.

Blog Entries

- [News from the MBS Xojo Plugins Version 23.4](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.4](#)
- [NSSplitView Control for Xojo](#)
- [MBS Xojo Plugins, version 23.4pr5](#)

Xojo Developer Magazine

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

9.34.2 Methods

9.34.3 addArrangedSubview(view as NSViewMBS)

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Adds a view as an arranged split pane.

Notes: If the view isn't a subview of the split view, calling this method adds it to the split view.

subviews array.

9.34.4 adjustSubviews

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Adjusts the sizes of the split view's subviews so they (plus the dividers) fill the split view.

Notes: When you call this method, the split view's subviews resize proportionally; the relative sizes of the subviews don't change.

The default implementation of this method resizes subviews proportionally so that the ratio of heights (when using horizontal dividers) or widths (when using vertical dividers) doesn't change, even though the absolute sizes change.

Call this method on split views where you've added or removed subviews to reestablish the consistency of subview placement.

This method invalidates the cursor when it is over a divider, ensuring the cursor is always of the correct type during and after resizing animations.

9.34.5 Constructor

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Creates a new split view with size 100/100 and position 0/0

Example:

```
dim t as new NSDatePickerMBS
```

Notes: On success the handle property is not zero.

See also:

- 9.34.6 Constructor(Handle as Integer) 656
- 9.34.7 Constructor(left as double, top as double, width as double, height as double) 657

9.34.6 Constructor(Handle as Integer)

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSSplitView handle.

Example:

```
dim t as new NSSplitViewMBS(0, 0, 100, 100)
```

```
dim v as new NSSplitViewMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSSplitView and the plugin retains this handle.
See also:

- 9.34.5 Constructor 656
- 9.34.7 Constructor(left as double, top as double, width as double, height as double) 657

9.34.7 Constructor(left as double, top as double, width as double, height as double)

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Creates a new split view with the given size and position.

Example:

```
dim x as new NSSplitViewMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.
See also:

- 9.34.5 Constructor 656
- 9.34.6 Constructor(Handle as Integer) 656

9.34.8 insertArrangedSubview(view as NSViewMBS, index as Integer)

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Adds a view as an arranged split pane at the specified index.

Notes: If the view is already an arranged view, calling this method moves the view to the specified index in the arrangedSubviews array. This change doesn't affect the view's index in the split view's subviews array.

If the view isn't a subview of the split view, calling this method adds it to the split view's subviews array.

9.34.9 isSubviewCollapsed(view as NSViewMBS) as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Returns whether the specified view is in a collapsed state.

Notes: subview: The subview in the split view.

Returns true if subview is in a collapsed state; otherwise, false.

9.34.10 `maxPossiblePositionOfDividerAtIndex(dividerIndex as Integer) as Double`

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Returns the maximum possible position of the divider at the specified index.

Notes: dividerIndex: The index of the divider.

Returns a number that specifies the maximum possible position of the divider.

The position is possible because the bounds of the split view and the current position of other dividers dictate it. Allowable positions result from letting the delegate apply constraints to the possible positions.

You can invoke this method to determine the range of values that you can pass to `setPosition`. You can also invoke it from delegate methods like `constrainSplitPosition` to implement relatively complex behaviors that depend on the current state of the split view.

The result of invoking this method when you haven't invoked `adjustSubviews`, and the subview frames are invalid, is undefined.

9.34.11 `minPossiblePositionOfDividerAtIndex(dividerIndex as Integer) as Double`

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Returns the minimum possible position of the divider at the specified index.

Notes: dividerIndex: The index of the divider.

Return a number that specifies the minimum possible position of the divider.

The position is possible because the bounds of the split view and the current position of other dividers dictate it. Allowable positions result from letting the delegate apply constraints to the possible positions.

You can invoke this method to determine the range of values that you can pass to `setPosition()`. You can

also invoke it from delegate methods like `constrainSplitPosition` to implement relatively complex behaviors that depend on the current state of the split view.

The result of invoking this method when you haven't invoked `adjustSubviews`, and the subview frames are invalid, is undefined.

9.34.12 `removeArrangedSubview(view as NSViewMBS)`

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Removes a view as an arranged split pane.

Notes: If the value of `arrangesAllSubviews` is false, calling this method doesn't remove the view as a subview; it remains in the split view's subviews array.

If you remove a view as a subview (either by calling `removeFromSuperview()` or removing it from the split view's subviews array), the system automatically removes the view as an arranged subview.

9.34.13 `setPosition(position as double, dividerIndex as Integer)`

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: Updates the location of a divider you specify by index.

Notes: `dividerIndex`: The index of the divider.

One of the views adjacent to the divider may collapse because the method's default implementation assumes a person is dragging the divider to the new location. The Auto Layout system collapses the view if it can't satisfy the view's constraints—typically imposed by its delegate—with the divider's new location.

`NSSplitView` doesn't invoke this method.

9.34.14 Properties

9.34.15 `arrangesAllSubviews as Boolean`

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that determines whether the split view arranges all of its subviews as split panes.

Notes: If the value of this property is true, the split view arranges all of its subviews automatically. The `arrangedSubviews` array is identical to the split view's subviews array, so any change to subviews reflects in the `arrangedSubviews` array. The default value of this property is true.

If the value of this property is false, you must explicitly add a view as an arranged subview to arrange it as a split pane. You add an arranged subview using `addArrangedSubview`.

When you change the value of this property from true to false, all existing subviews stay as arranged subviews in `arrangedSubviews`. When you change the value of this property from false to true, all existing subviews become arranged subviews, and the value of the `subviews` array becomes the `arrangedSubviews` array.
(Read and Write property)

9.34.16 `autosaveName` as String

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: The name to use when the system automatically saves the split view,Äôs divider configuration.

Notes: If this property,Äôs value is empty, autosaving doesn,Äôt occur.

(Read and Write property)

9.34.17 `dividerColor` as NSColorMBS

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: The color of the dividers that the split view draws between subviews.

Notes: The default implementation of this method returns clear when the split view,Äôs `dividerStyle` is thick, or when `dividerStyle` is `paneSplitter` and the split view is in a textured window. The system draws all other thin dividers with a color that provides appropriate contrast between two white panes.

Can be assigned to when you use the `NSSplitView` from our `NSSplitViewControlMBS`.

(Read only property)

9.34.18 `dividerStyle` as Integer

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: The style of divider between views.

Notes: See `NSSplitViewDividerStyle*` constants.

(Read and Write property)

9.34.19 `dividerThickness` as Double

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: The thickness of the dividers for the split view.

Notes: Can be assigned to when you use the NSSplitView from our NSSplitViewControlMBS.
(Read and Write property)

9.34.20 Vertical as Boolean

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that determines the geometric orientation of the split view's dividers.

Notes: The default value of this property is false, which indicates horizontal dividers and views that stack one above the other (top-to-bottom) in the containing split view controller,Äôs view.

To specify vertical dividers and a horizontal (side-by-side) arrangement of views within a split view controller, implement this property to return true.

(Read and Write property)

9.34.21 holdingPriorityForSubview(subviewIndex as Integer) as Integer

Plugin Version: 23.4, Platform: macOS, Targets: Desktop only.

Function: The priority of the subview,Äôs width or height when resizing.

Notes: The priority is the manner that the split view subviews use to maintain their width (for a vertical split view) or height (for a horizontal split view). During a split view resize, subviews with higher priorities maintain their sizes before subviews with lower priorities. The subview with the lowest priority is the first to gain additional thickness if the split view grows or shrinks.

(Read and Write computed property)

9.34.22 Constants

Layout Priorities

Constant	Value	Description
NSLayoutPriorityDefaultHigh	750	Priority level with which a button resists compressing its contents.
NSLayoutPriorityDefaultLow	250	Priority level at which a button hugs its contents horizontally.
NSLayoutPriorityDragThatCannotResizeWindow	490	Priority level at which a split view divider, say, is dragged.
NSLayoutPriorityDragThatCanResizeWindow	510	Appropriate priority level for a drag that may end up resizing the window.
NSLayoutPriorityFittingSizeCompression	50	When you send a fittingSize message to a view, the smallest size that is large enough for the view's contents is computed.
NSLayoutPriorityRequired	1000	A required constraint.
NSLayoutPriorityWindowSizeStayPut	500	Priority level for the window,Äôs current size.

Divider Styles

Constant	Value	Description
<code>NSSplitViewDividerStylePaneSplitter</code>	3	A thick style divider with a 3D appearance displays between subviews.
<code>NSSplitViewDividerStyleThick</code>	1	A thick style divider displays between subviews.
<code>NSSplitViewDividerStyleThin</code>	2	A thin style divider displays between subviews.

9.35 class NSTabViewItemMBS

9.35.1 class NSTabViewItemMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The Cocoa class for items on the tabview.

Notes: An NSTabViewItem is a convenient way for presenting information in multiple pages. A tab view is usually distinguished by a row of tabs that give the visual appearance of folder tabs. When the user clicks a tab, the tab view displays a view page provided by your application. A tab view keeps a zero-based array of NSTabViewItems, one for each tab in the view.

Blog Entries

- [Disable tab panel item in Xojo](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr4](#)

9.35.2 Methods

9.35.3 Constructor(identifier as Variant)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new tabview item with the identifier.

9.35.4 Properties

9.35.5 color as NSColorMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The color of the tab view item.

Notes: May not be used by the control.

(Read and Write property)

9.35.6 Enabled as Boolean

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The enabled state of the tab view item.

Example:

```
dim n as NSTabViewMBS = TabPanel1.NSTabViewMBS
dim t as NSTabViewItemMBS = n.tabViewItemAtIndex(0)
```

t.Enabled = `false`

Notes: (Read and Write property)

9.35.7 Handle as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The internal reference to the NSTabViewItem object.

Notes: (Read and Write property)

9.35.8 identifier as Variant

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The identifier for this item.

Notes: (Read and Write property)

9.35.9 image as NSImageMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The image for the tab panel item.

Notes: Available in Mac OS X 10.10.

(Read and Write property)

9.35.10 initialFirstResponder as NSViewMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The initial first responder for the view associated with the receiver.

Notes: Sets the initial first responder for the view associated with the receiver (the view that is displayed when a user clicks on the tab) to view.

(Read and Write property)

9.35.11 label as string

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The label text for the receiver.

Notes: (Read and Write property)

9.35.12 tabState as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the current display state of the tab associated with the receiver.

Notes: The possible values are NSSelectedTab, NSBackgroundTab, or NSPressedTab. Your application does not directly set the tab state.

(Read only property)

9.35.13 tabView as NSTabViewMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the parent tab view for the receiver.

Notes: Note that this is the tab view itself, not the view displayed when a user clicks the tab.

A tab view item normally learns about its parent tab view when it is inserted into the view's array of items. The NSTabView methods addTabViewItem and insertTabViewItem set the tab view for the added or inserted item.

(Read only property)

9.35.14 toolTip as string

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The tooltip displayed for the tab view item.

Notes: (Read and Write property)

9.35.15 view as NSViewMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The view associated with the receiver to view.

Notes: This is the view displayed when a user clicks the tab. When you set a new view, the old view is released.

(Read and Write property)

9.35.16 Constants

Constants

Constant	Value	Description
<code>NSBackgroundTab</code>	1	One of the constants describing the current display state of a tab. A tab that's not being displayed.
<code>NSPressedTab</code>	2	One of the constants describing the current display state of a tab. A tab that the user is in the process of clicking. That is, the user has pressed the mouse button while the cursor is over the tab but has not released the mouse button.
<code>NSSelectedTab</code>	0	One of the constants describing the current display state of a tab. The tab that's being displayed.

9.36 class NSTabViewMBS

9.36.1 class NSTabViewMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The Cocoa tabpanel control.

Notes: An NSTabView object provides a convenient way to present information in multiple pages. The view contains a row of tabs that give the appearance of folder tabs, as shown in the following figure. The user selects the desired page by clicking the appropriate tab or using the arrow keys to move between pages. Each page displays a view hierarchy provided by your application.

You can embed this view in a CustomNSViewMBS to get more events for mouse and keyboard.
Subclass of the NSViewMBS class.

Blog Entries

- [Disable tab panel item in Xojo](#)

9.36.2 Methods

9.36.3 addTabViewItem(tabViewItem as NSTabViewItemMBS)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Adds the tab item specified by tabViewItem.

Notes: tabViewItem: The tab view item to be added.

The item is added at the end of the array of tab items, so the new tab appears on the right side of the view.

9.36.4 Constructor

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new tab view with size 100/100 and position 0/0

Example:

```
dim t as new NSTabViewMBS
```

Notes: On success the handle property is not zero.

See also:

- 9.36.5 Constructor(Handle as Integer)

- 9.36.6 Constructor(left as Double, top as Double, width as Double, height as Double) 668

9.36.5 Constructor(Handle as Integer)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSTabView handle.

Example:

```
dim t as new NSTabViewMBS(0, 0, 100, 100)
dim v as new NSTabViewMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSTabView and the plugin retains this handle.

See also:

- 9.36.4 Constructor 667
- 9.36.6 Constructor(left as Double, top as Double, width as Double, height as Double) 668

9.36.6 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new tab view with the given size and position.

Example:

```
dim x as new NSTabViewMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 9.36.4 Constructor 667
- 9.36.5 Constructor(Handle as Integer) 668

9.36.7 contentRect as NSRectMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the rectangle describing the content area of the receiver.

Notes: This area does not include the space required for the receiver's tabs or borders (if any).

9.36.8 indexOfTabViewItem(tabViewItem as NSTabViewItemMBS) as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the specified item in the tab view.

Notes: The zero-based index of tabViewItem, or [NSNotFound] if the item is not found.

9.36.9 indexOfTabViewItemWithIdentifier(identifier as Variant) as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the item that matches the specified identifier. identifier, or NSNotFound (-1) if the item is not found.

Notes: Returns nil on any error.

9.36.10 insertTabViewItem(tabViewItem as NSTabViewItemMBS, atIndex as Integer)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Inserts tabViewItem into the receiver's array of tab view items at index.

Notes: tabViewItem: The tab view item to be added.

index: The index at which to insert the tab view item. The index parameter is zero-based.

9.36.11 minimumSize as NSSizeMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the minimum size necessary for the receiver to display tabs in a useful way.

Notes: You can use the value returned by this method to limit how much a user can resize a tab view.

9.36.12 numberOfTabViewItems as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the number of items in the receiver's array of tab view items.

9.36.13 removeTabViewItem(tabViewItem as NSTabViewItemMBS)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Removes the item specified by `tabViewItem` from the receiver's array of tab view items.

9.36.14 `selectedTabViewItem` as `NSTabViewItemMBS`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Selects the specified tab view item.

9.36.15 `selectFirstTabViewItem`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: This action method selects the first tab view item.

9.36.16 `selectLastTabViewItem`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: This action method selects the last tab view item.

9.36.17 `selectNextTabViewItem`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: This action method selects the next tab view item in the sequence.

Notes: If the currently visible item is the last item in the sequence, this method does nothing, and the last page remains displayed.

9.36.18 `selectPreviousTabViewItem`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: This action method selects the previous tab view item in the sequence.

Notes: If the currently visible item is the first item in the sequence, this method does nothing, and the first page remains displayed.

9.36.19 selectTabViewItem(tabViewItem as NSTabViewItemMBS)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the tab view item for the currently selected tab.

Notes: Returns the currently selected tab view item, or nil if no item is selected.

9.36.20 selectTabViewItemAtIndex(index as Integer)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Selects the tab view item specified by index.

Notes: The index parameter is base 0.

9.36.21 selectTabViewItemWithIdentifier(identifier as Variant)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Selects the tab view item specified by identifier.

9.36.22 tabViewItemAtIndex(index as Integer) as NSTabViewItemMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the tab view item at index in the tab view's array of items.

Notes: index: The index at which to insert the tab view item. The index parameter is zero-based.

Returns the tab view item at the specified index.

9.36.23 tabViewItemAtPoint(x as Double, y as Double) as NSTabViewItemMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the tab view item at the specified point.

Notes: Returns the tab view item under the hit point, or nil if no tab view item is under that location. You can use this method to find a tab view item based on a user's mouse click.

9.36.24 `tabViewItems` as `NSTabViewItemMBS()`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's array of tab view items.

Notes: A tab view keeps an array containing one tab view item for each tab in the view.

9.36.25 Properties

9.36.26 `allowsTruncatedLabels` as `boolean`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Whether if the receiver allows truncating for labels that don't fit on a tab.

Notes: Value is true if the receiver allows truncating for labels that don't fit on a tab, otherwise false.

The default is true.

When truncating is allowed, the tab view inserts an ellipsis, if necessary, to fit a label in the tab.

(Read and Write computed property)

9.36.27 `controlSize` as `Integer`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The size of the receiver.

Notes: Use `NSRegularControlSize`, `NSSmallControlSize` or `NSMiniControlSize`.

(Read and Write computed property)

9.36.28 `controlTint` as `Integer`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The tab view's control tint.

Notes: Use `NSDefaultControlTint`, `NSBlueControlTint`, `NSGraphiteControlTint` or `NSClearControlTint`.

(Read and Write computed property)

9.36.29 `drawsBackground` as `boolean`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Whether if the receiver draws a background color when the tab view type is NSNoTabsNoBorder.
Notes: True if the receiver draws a background color when the tab view type is NSNoTabsNoBorder, otherwise false.

If the receiver uses bezeled edges or a line border, the appropriate background color for that border is used.
(Read and Write computed property)

9.36.30 font as NSFontMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The font for tab label text.

Notes: Tab height is adjusted automatically to accommodate a new font size. If the view allows truncating, tab labels are truncated as needed.
(Read and Write computed property)

9.36.31 tabViewType as Integer

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The tab type for the receiver.

Notes: Use constants: NSTopTabsBezelBorder, NSLeftTabsBezelBorder, NSBottomTabsBezelBorder, NSRightTabsBezelBorder, NSNoTabsBezelBorder, NSNoTabsLineBorder or NSNoTabsNoBorder.
(Read and Write computed property)

9.36.32 Constants

Constants

Constant	Value	Description
NSBlueControlTint	1	One of the constants to specify a the control tint. Aqua control tint
NSBottomTabsBezelBorder	2	One of the constants to specify the tab view's type as used by <code>tabViewType</code> . Tabs are on the bottom of the view with a beveled border.
NSClearControlTint	7	One of the constants to specify a the control tint. Clear control tint
NSDefaultControlTint	0	One of the constants to specify a the control tint. The current default tint setting
NSGraphiteControlTint	6	One of the constants to specify a the control tint. Graphite control tint
NSLeftTabsBezelBorder	1	One of the constants to specify the tab view's type as used by <code>tabViewType</code> . Tabs are on the left of the view with a beveled border.
NSMiniControlSize	2	One of the values for the <code>ControlSize</code> property. The control has a smaller size than <code>NSSmallControlSize</code> .
NSNoTabsBezelBorder	4	One of the constants to specify the tab view's type as used by <code>tabViewType</code> . The view does not include tabs and has a beveled border.
NSNoTabsLineBorder	5	One of the constants to specify the tab view's type as used by <code>tabViewType</code> . The view does not include tabs and has a lined border.
NSNoTabsNoBorder	6	One of the constants to specify the tab view's type as used by <code>tabViewType</code> . The view does not include tabs and has no border.
NSRegularControlSize	0	One of the values for the <code>ControlSize</code> property. The control is sized as regular.
NSRightTabsBezelBorder	3	One of the constants to specify the tab view's type as used by <code>tabViewType</code> . Tabs are on the right of the view with a beveled border.
NSSmallControlSize	1	One of the values for the <code>ControlSize</code> property. This constant is for controls that cannot be resized in one direction, such as push buttons, radio buttons, checkboxes, sliders, scroll bars, pop-up buttons, tabs, and progress indicators. You should use a small system font with a small control.
NSTopTabsBezelBorder	0	One of the constants to specify the tab view's type as used by <code>tabViewType</code> . The view includes tabs on the top of the view and has a beveled border (the default).

9.37 class NSTextFieldCellMBS

9.37.1 class NSTextFieldCellMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The text field class for using as NSCell.

Example:

```
// ask for the textfield behind the label
dim n as NSTextFieldMBS = NSTextFieldMBS(label1.NSViewMBS)

// query cell
dim c as NSTextFieldCellMBS = n.cell

// and set background style
c.backgroundColor = NSTextFieldCellMBS.NSBackgroundColorRaised
```

Notes: The NSTextFieldCell class adds to the text display capabilities of the NSCell class by allowing you to set the color of both the text and its background. You can also specify whether the cell draws its background at all.

All of the methods declared by this class are also declared by the NSTextField class, which uses NSTextFieldCell objects to draw and edit text. These NSTextField cover methods call the corresponding NSTextFieldCell methods.

Placeholder strings, set using PlaceholderString or PlaceholderAttributedString, now appear in the text field cell if the actual string is "". They are drawn in grey on the cell.

Subclass of the NSActionCellMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 11.3pr1](#)
- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)

Xojo Developer Magazine

- [7.6, page 8: News](#)
- [14.4, page 29: NSTabula Rasa, What to do when your new sports car arrives in parts by Ulrich Bogun](#)

9.37.2 Methods

9.37.3 `allowedInputSourceLocales` as `string()`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns an array of locale identifiers representing input sources that are allowed to be enabled when the receiver has the keyboard focus.

Notes: Available in Mac OS X v10.5 and later.

9.37.4 `Constructor(text as string)`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new Cell object with a text.

Example:

```
dim c as new NSTextFieldCellMBS("Hello")
MsgBox c.StringValue
```

9.37.5 `setAllowedInputSourceLocales(identifiers() as string)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets an array of locale identifiers representing input sources that are allowed to be enabled when the receiver has the keyboard focus.

Notes: You can use the meta-locale identifier, `NSAllRomanInputSourcesLocaleIdentifier`, to specify input sources that are limited for Roman script editing.

Available in Mac OS X v10.5 and later.

9.37.6 `setUpFieldEditorAttributes(textobj as NSTextMBS) as NSTextMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets up the field editor. You never invoke this method directly; by overriding it, however, you can customize the field editor.

9.37.7 setWantsNotificationForMarkedText(value as boolean)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Directs the cell's associated field editor to post text change notifications.

Notes: If true, the field editor posts text change notifications (NSTextDidChangeNotification) while editing marked text; if false, notifications are delayed until the marked text confirmation.

Available in Mac OS X v10.5 and later.

9.37.8 Properties

9.37.9 backgroundColor as NSColorMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The color of the background the receiver draws behind the text.

Notes: (Read and Write property)

9.37.10 bezelStyle as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's bezel style.

Notes: To set the bezel style, you must have already set Bezeled to true.
(Read and Write property)

9.37.11 drawsBackground as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver draws its background color.

Notes: In order to prevent inconsistent rendering, background color rendering is disabled for rounded-bezel text fields.
(Read and Write property)

9.37.12 placeholderAttributedString as NSAttributedStringMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The placeholder of the cell as an attributed string.

Notes: Note that invoking this successfully will clear out any plain text string set by PlaceholderString.

Available in Mac OS X v10.3 and later.
(Read and Write property)

9.37.13 placeholderString as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The placeholder of the cell as a plain text string.

Notes: Note that invoking this successfully will clear out any attributed string set by setPlaceholderAttributedString.

(Read and Write property)

9.37.14 textColor as NSColorMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The color used to draw the receiver's text.

Notes: (Read and Write property)

9.37.15 Constants

Constants

Constant	Value	Description
NSTextFieldRoundedBezel	1	One of the constants to specify the bezel style of the text field cell. Corners are rounded. Available in Mac OS X v10.2 and later.
NSTextFieldSquareBezel	0	One of the constants to specify the bezel style of the text field cell. Corners are square. Available in Mac OS X v10.2 and later.

9.38 control NSTextFieldControlMBS

9.38.1 control NSTextFieldControlMBS

Plugin Version: 13.4, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a NSTextField.

Notes: This control embeds a special NSTextField subclass.

Designed for Xojo 2013r1 and newer. May work on Xojo 2012, but not perfectly.

Please use view property to access the underlying object and set properties.

Blog Entries

- [MBS Xojo Plugins, version 23.2pr1](#)
- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr2](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr7](#)

9.38.2 Properties

9.38.3 View as NSTextFieldMBS

Plugin Version: 13.4, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.

Notes: Use this object to set more options on the control.

(Read only property)

9.38.4 Events

9.38.5 Action

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: The control's action was triggered.

Notes: The text changed.

9.38.6 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.38.7 Close

Plugin Version: 13.4, Platform: macOS, Targets: .

Function:

The control is about to close.

In Xojo version 2021r3 and newer this event is named Closing.

9.38.8 ConstructContextMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

9.38.9 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Called when a menuitem is chosen.

Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

9.38.10 didCloseContextMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextMenu event.

9.38.11 EnableMenuItems

Plugin Version: 17.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In Xojo version 2021r3 and newer this event is named MenuBarSelected.

9.38.12 FrameChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.38.13 GotFocus

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

Function:

The control itself got focus.

In Xojo version 2021r3 and newer this event is named FocusReceived.

Notes:

This only fires if the control itself got focus and not a sub control.

9.38.14 LostFocus

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

Function:

The control lost focus.

In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.38.15MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control's region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return `False`, the system handles the `MouseDown` so the above event handlers do not get called.

9.38.16 MouseDrag(x as Integer, y as Integer)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

9.38.17 MouseUp(x as Integer, y as Integer)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.38.18 Open

Plugin Version: 13.4, Platform: macOS, Targets: .

Function:

The control is about to be created and you can initialize it.

In Xojo version 2021r3 and newer this event is named `Opening`.

9.38.19 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.38.20 TextDidBeginEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

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

Function: Sent when a control with editable text begins an editing session.

Notes: Notification: The notification object. The name of the notification is always NSControlTextDidBeginEditingNotification.

This event is invoked when the user begins editing text in a control such as a text field or a form field. The control posts a NSControlTextDidBeginEditingNotification notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also delivered for inspection.

9.38.21 TextDidChange(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

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

Function: Sent when the text in the receiving control changes.

Notes: Notification: The notification object. The name of the notification is always NSControlTextDidChangeNotification.

This event is invoked when text in a control such as a text field or form changes. The control posts a NSControlTextDidChangeNotification notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is provided as parameter for inspection.

9.38.22 TextDidEndEditing(fieldEditor as NSTextMBS, notification as NSNotificationMBS)

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

Function: Sent when a control with editable text ends an editing session.

Notes: Notification: The notification object. The name of the notification is always NSControlTextDidEndEditingNotification.

This event is invoked when the user stops editing text in a control such as a text field or form. The control posts a NSControlTextDidEndEditingNotification notification, and if the control's subclass implements this event, it is automatically registered to receive the notification. The field editor is also provided for inspection.

9.38.23 textShouldBeginEditing(fieldEditor as NSTextMBS) as boolean

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

Function: The event called to decide whether text editing should be allowed.

Notes: Return true to allow text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.38.24 textShouldEndEditing(fieldEditor as NSTextMBS) as boolean

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

Function: The event called to decide whether ending text editing should be allowed.

Notes: Return true to allow end of text editing or false to deny.

Be aware that an event in Xojo without return will cause false to be returned.

9.38.25 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.39 class NSTextFieldMBS

9.39.1 class NSTextFieldMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: An NSTextField object is a kind of NSControl that displays text that the user can select or edit and that sends its action message to its target when the user presses the Return key while editing.

Notes: Subclass of the NSControlMBS class.

Blog Entries

- [MBS Xojo Plugins, version 17.1pr4](#)
- [MBS Xojo / Real Studio Plugins, version 14.4pr1](#)
- [MBS Xojo / Real Studio Plugins, version 13.3pr1](#)
- [MBS Real Studio Plugins, version 12.5pr6](#)
- [MBS Real Studio Plugins, version 12.3pr2](#)
- [MBS REALbasic Plugins Version 10.4 release notes](#)
- [MBS REALbasic Plugins, version 10.4pr7](#)

9.39.2 Methods

9.39.3 Constructor

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new text field with size 100/100 and position 0/0

Example:

```
dim t as new NSTextFieldMBS
```

Notes: On success the handle property is not zero.

See also:

- [9.39.4 Constructor\(Handle as Integer\)](#) 685
- [9.39.5 Constructor\(left as Double, top as Double, width as Double, height as Double\)](#) 686

9.39.4 Constructor(Handle as Integer)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSTextField handle.

Example:

```
dim t as new NSTextFieldMBS(0, 0, 100, 100)
dim v as new NSTextFieldMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSTextField and the plugin retains this handle.

See also:

- 9.39.3 Constructor 685
- 9.39.5 Constructor(left as Double, top as Double, width as Double, height as Double) 686

9.39.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new text field with the given size and position.

Example:

```
dim x as new NSTextFieldMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 9.39.3 Constructor 685
- 9.39.4 Constructor(Handle as Integer) 685

9.39.6 selectText

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Ends editing and selects the entire contents of the receiver if it's selectable.

Notes: If the receiver isn't in some window's view hierarchy, this method has no effect.

9.39.7 Properties

9.39.8 AllowsCharacterPickerTouchBarItem as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Whether to allow character picker in touch bar.

Notes: Available in macOS 10.12.2.

(Read and Write property)

9.39.9 allowsEditingTextAttributes as boolean

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver allows the user to change font attributes of the receiver's text.

Notes: If true, the user is permitted to change font attributes of the receiver's text; if flag is false, the user isn't so permitted. You can change text attributes programmatically regardless of this setting.

(Read and Write property)

9.39.10 AutomaticTextCompletionEnabled as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Whether automatic text completion is enabled.

Notes: Available in macOS 10.12.2.

(Read and Write property)

9.39.11 backgroundColor as NSColorMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The color of the background that the receiver's cell draws behind the text.

Notes: (Read and Write property)

9.39.12 Bezeled as boolean

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the receiver draws a bezeled frame.

Notes: (Read and Write property)

9.39.13 `bezelStyle` as Integer

Plugin Version: 10.4, Platform: macOS, Targets: Desktop only.

Function: The receiver's bezel style.

Notes: You must have already sent the receiver `Bezeled` with `true` to make this property take affect.
(Read and Write property)

9.39.14 `Bordered` as boolean

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the receiver draws a black border around its contents.

Notes: True if the receiver draws a solid black border around its contents; otherwise false.
(Read and Write property)

9.39.15 `drawsBackground` as boolean

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Controls whether the receiver's cell draws its background color behind its text.

Notes: In order to prevent inconsistent rendering, background color rendering is disabled for rounded-bezel text fields.

To really make the background go away, also set `bordered=false`.
(Read and Write property)

9.39.16 `Editable` as boolean

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Whether the user can edit the receiver's text.

Notes: If true, then the user is allowed to both select and edit text. If flag is false, then the user isn't permitted to edit text, and the receiver's selectability is restored to its previous value.

For example, if an `NSTextField` object is selectable but not editable, then made editable for a time, then made not editable, it remains selectable. To guarantee that text is neither editable nor selectable, simply use `setSelectable` to turn off selectability.

(Read and Write property)

9.39.17 importsGraphics as boolean

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Controls whether the receiver allows the user to drag image files into it.

Notes: If true, the receiver accepts dragged images; if false, it doesn't. You can add images programmatically regardless of this setting.

(Read and Write property)

9.39.18 placeholderAttributedString as NSAttributedStringMBS

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: The attributed placeholder string.

Notes: (Read and Write property)

9.39.19 placeholderString as String

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: The plain text placeholder string.

Notes: (Read and Write property)

9.39.20 Selectable as boolean

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver is selectable (but not editable).

Notes: If true, the receiver is made selectable but not editable (use Editable to make text both selectable and editable). If false, the text is neither editable nor selectable.

(Read and Write property)

9.39.21 textColor as NSColorMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The color used to draw the receiver's text.

Notes: (Read and Write property)

9.39.22 Constants

Constants

Constant	Value	Description
<code>NSTextFieldRoundedBezel</code>	1	One of the constants for the <code>bezelStyle</code> property. Corners are rounded.
<code>NSTextFieldSquareBezel</code>	0	One of the constants for the <code>bezelStyle</code> property. Corners are square.

9.40 class NSTextFinderMBS

9.40.1 class NSTextFinderMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: An optional search-and-replace find interface inside a view, usually a scroll view.

Notes: The class serves as a controller for the standard Cocoa find bar. The NSTextFinder class interacts heavily with a client object which supports the NSTextFinderClient protocol. The client object provides access to the content being searched and provides visual feedback for a search operation.

All menu items related to finding (Find, ⌘F, Find Next, Find Previous, Use Selection for Find, etc.) should have the same action, performTextFinderAction:, which gets sent down the responder chain in the standard method.

see

<https://developer.apple.com/documentation/appkit/nstextfinder>

Blog Entries

- [XDC Anywhere - MBS Xojo Plugins](#)
- [MBS Xojo Plugin, June 2021 News](#)
- [Use Text Finder for TextArea in Xojo](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.5](#)
- [MBS Xojo Plugins, version 18.5pr3](#)

Videos

- [XDC Anywhere - MBS Xojo Plugins](#)
- [MBS Xojo Videos - MBS Xojo Plugin, June 2021 News](#)

9.40.2 Methods

9.40.3 cancelFindIndicator

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Cancels the find indicator immediately.

Notes: There may be some circumstances where the find indicator should be immediately cancelled or hidden, such as when the view's content or selection is changed without the knowledge of the text finder. This method will immediately cancel the current find indicator.

The NSTextFinderMBS and NSViewMBS classes will handle the find indicator correctly when a content view is resized, moved, or removed from the view hierarchy. If your content view's scrolling is done by an NSScrollViewMBS, the find indicator will also be handled for you during scrolling.

9.40.4 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Creates a new object.

9.40.5 performAction(operation as Integer)

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Performs the specified text finding action.

Notes: operation: The text finding action. See NSTextFinderAction for the possible values.

Objects that respond to performTextFinderAction typically call validateAction: to ensure that the action is valid and then invoke performAction if validation is successful.

When invoking the validateAction and performAction the item or sender,Ãs tag should be passed as the parameter. By convention, the sender parameter for this method will have an kAction* constants set as its tag. The responder that receives this method should pass the tag as the action for this method:

9.40.6 validateAction(operation as Integer) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Allows validation of the find action before performing.

Notes: Returns true if the operation is valid; otherwise false.

9.40.7 Properties

9.40.8 client as NSViewMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: The object that provides the target search string, find bar location, and feedback methods.

Notes: The NSTextFinderMBS instance class must be associated with a client object that implements the NSTextFinderClient protocol in order to function. The client is responsible for providing the string to be searched, the location for the find bar, and the methods which control feedback to the user about the search results.

(Read and Write property)

9.40.9 findBarController as NSViewMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Specifies the find bar container.

Notes: This property must be set to support the find bar.

When the find bar is to be shown, NSTextFinderMBS invokes showFindBarView on the findBarController object, passing the view for the find bar, which it should display somewhere that is associated appropriately with the content being searched.

The NSScrollViewMBS class implements NSTextFinderBarController protocol and is the correct place to display the find bar in most circumstances. The container may freely modify the find bar view's width and origin, but not its height.

If this property is not set, then the find bar cannot be shown.

(Read and Write property)

9.40.10 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: The internal reference number.

Notes: (Read and Write property)

9.40.11 incrementalSearchingEnabled as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Determines if incremental searching is enabled.

Notes: If true, then the find bar will do incremental searches. If it returns false, then the find bar will behave regularly.

The default value is false.

(Read and Write property)

9.40.12 incrementalSearchingShouldDimContentView as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: Determines the type of incremental search feedback to be presented.

Notes: If true, then when an incremental search begins, the findBarController instance's parent contentView will be dimmed, except for the locations of the incremental matches. If false, then the incremental matches will not be highlighted automatically, but you can use incrementalMatchRanges to highlight the matches yourself.

The default value is true.

(Read and Write property)

9.40.13 Constants

Actions

Constant	Value	Description
kActionHideFindInterface	11	Hides the find bar interface.
kActionHideReplaceInterface	13	Displays the find bar interface including the replace functionality.
kActionNextMatch	2	The next match, if any, is displayed.
kActionPreviousMatch	3	The previous match, if any, is displayed.
kActionReplace	5	Replaces a single instance of the string.
kActionReplaceAll	4	All occurrences of the string are replaced.
kActionReplaceAllInSelection	8	Selects all matching search strings within the current selection.
kActionReplaceAndFind	6	Replaces a single instance of the string and searches for the next match.
kActionSelectAll	9	Selects all matching search strings.
kActionSelectAllInSelection	10	Replaces all occurrences of the string within the current selection.
kActionSetSearchString	7	Sets the search string.
kActionShowFindInterface	1	The find bar interface is displayed.
kActionShowReplaceInterface	12	Displays the find bar interface including the replace functionality.

9.41 control NSTextViewControlMBS

9.41.1 control NSTextViewControlMBS

Plugin Version: 13.4, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a NSTextView.

Notes: This control embeds a special NSTextView subclass.

Designed for Xojo 2013r1 and newer. May work on Xojo 2012, but not perfectly.

Please use view property to access the underlying object and set properties.

Blog Entries

- [News from the MBS Xojo Plugins in version 21.5](#)
- [New desktop controls](#)
- [MBS Xojo / Real Studio Plugins, version 16.5pr6](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr8](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr1](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr7](#)

9.41.2 Properties

9.41.3 AcceptTabs as Boolean

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Whether the control should accept tab keys.

Notes: If true, the plugin will not forward the tab keydown/keyup events to Xojo, because Xojo would do switch to next control.

(Read and Write property)

9.41.4 Scrollview as Variant

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The scrollview for this textview.

Notes: (Read only property)

9.41.5 View as NSTextViewMBS

Plugin Version: 13.4, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.

Notes: Use this object to set more options on the control.
(Read only property)

9.41.6 Events

9.41.7 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

9.41.8 Close

Plugin Version: 13.4, Platform: macOS, Targets: .

Function:

The control is about to close.

In Xojo version 2021r3 and newer this event is named Closing.

9.41.9 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

9.41.10 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Called when a menuitem is chosen.

Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

9.41.11 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

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

Function: Did close contextual menu.

Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

9.41.12 EnableMenuItems

Plugin Version: 17.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.

In Xojo version 2021r3 and newer this event is named MenuBarSelected.

9.41.13 FrameChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the frame changed.

Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

9.41.14 GotFocus

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

Function:

The control itself got focus.

In Xojo version 2021r3 and newer this event is named FocusReceived.

Notes:

This only fires if the control itself got focus and not a sub control.

9.41.15 LostFocus

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

Function:

The control lost focus.

In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

9.41.16 `MouseDown(x as Integer, y as Integer, Modifiers as Integer)` As Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control's region at the location passed in to x, y.

Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner of the Control.

Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

9.41.17 `MouseDrag(x as Integer, y as Integer)`

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.

Notes: Mouse location is local to the control passed in to x, y.

As this event is fired continuously (hundreds of times per second), it is your responsibility to determine if the mouse has really moved.

9.41.18 `MouseUp(x as Integer, y as Integer)`

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was released.

Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

9.41.19 `Open`

Plugin Version: 13.4, Platform: macOS, Targets: .

Function:

The control is about to be created and you can initialize it.
In Xojo version 2021r3 and newer this event is named Opening.

9.41.20 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.

Notes: Please invalidate any cached bitmaps or other relevant state.

9.41.21 shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string) as boolean

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: Sent when a text view needs to determine if text in a specified range should be changed.

Notes: affectedCharRange: The range of characters to be replaced.

replacementString: The characters that will replace the characters in affectedCharRange; nil if only text attributes are being changed.

Return true to allow the replacement, or false to reject the change.

9.41.22 textDidBeginEditing

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: Informs you that the text object has begun editing (that the user has begun changing it).

9.41.23 textDidChange

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: Informs you that the text object has changed its characters or formatting attributes.

9.41.24 textDidEndEditing

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: Informs you that the text object has finished editing (that it has resigned first responder status).

9.41.25 `textShouldBeginEditing` as boolean

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: Invoked when a text object begins to change its text, this method requests permission to begin editing.

Notes: If the delegate returns false, the text object proceeds to make changes. If the delegate returns true, the text object abandons the editing operation. This method is also invoked when the user drags and drops a file onto the text object.

9.41.26 `textShouldEndEditing` as boolean

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: Invoked from a text object's implementation of `resignFirstResponder`, this method requests permission to end editing.

Notes: If the delegate returns false, the text object proceeds to finish editing and resign first responder status. If the delegate returns true, the text object selects all of its text and remains the first responder.

9.41.27 `textViewDidChangeSelection`

Plugin Version: 13.4, Platform: macOS, Targets: .

Function: Sent when the selection changes in the text view.

9.41.28 `willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)`

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

Function: Will show contextual menu.

Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

9.42 class NSTextViewMBS

9.42.1 class NSTextViewMBS

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The cocoa text view class.

Notes: Like the editfield in Xojo.

Should be placed in a scrollview.

You can embed this view in a CustomNSViewMBS to get more events for mouse and keyboard.

Subclass of the NSTextMBS class.

Blog Entries

- [RTF functions in MBS Plugins](#)
- [Adding a Scintilla based control to Xojo](#)
- [Use Text Finder for TextArea in Xojo](#)
- [Windows Font Dialog](#)
- [Line Wrap for Textarea in Xojo Mac applications](#)
- [SuperScript and SubScript for Xojo Textarea](#)
- [Tip of the day: Adding links to Textarea on OS X](#)
- [Tip of the day: Autohide scrollbar](#)
- [Show invisible characters in NSTextView/TextArea](#)
- [Using NSTextViewMBS](#)

Videos

- [Presentation from Xojo Developer Conference 2019 in Miami.](#)

Xojo Developer Magazine

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

9.42.2 Methods

9.42.3 alignJustified

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Applies full justification to selected paragraphs (or all text, if the receiver is a plain text object).

9.42.4 breakUndoCoalescing

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Informs the receiver that it should begin coalescing successive typing operations in a new undo grouping.

Notes: This method should be invoked when saving the receiver's contents to preserve proper tracking of unsaved changes and the document's dirty state.

9.42.5 changeAttributes

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Changes the attributes of the current selection.

Notes: This method changes the attributes by invoking `convertAttributes:` on sender and applying the returned attributes to the appropriate text. See the `NSFontManager` class reference for more information on attribute conversion.

Available in Mac OS X v10.3 and later.

9.42.6 changeColor

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets the color of the selected text.

9.42.7 changeDocumentBackgroundColor

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: An action method used to set the background color.

Notes: This method gets the new color by sending a color message to sender.

This will only set the background color if `allowsDocumentBackgroundColorChanger` returns true.

9.42.8 checkTextInDocument

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Check text in document.

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

9.42.9 checkTextInSelection

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Check text in selection.

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

9.42.10 complete

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Invokes completion in a text view.

Notes: By default invoked using the Escape key, this method provides users with a choice of completions for the word currently being typed. May be invoked programmatically if autocompletion is desired by a client of the text system. You can change the key invoking this method using the text system's key bindings mechanism; see "Text System Defaults and Key Bindings" for an explanation of the procedure (on Apple website).

9.42.11 Constructor

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new text view with size 100/100 and position 0/0

Example:

```
dim t as new NSTextViewMBS
```

Notes: On success the handle property is not zero.

See also:

- 9.42.12 Constructor(Handle as Integer) 703
- 9.42.13 Constructor(left as Double, top as Double, width as Double, height as Double) 704

9.42.12 Constructor(Handle as Integer)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSTextView 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 `NSTextView` and the plugin retains this handle.

See also:

- 9.42.11 Constructor 703
- 9.42.13 Constructor(left as Double, top as Double, width as Double, height as Double) 704

9.42.13 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Creates a new text view with the given size and position.

Example:

```
dim x as new NSTextViewMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 9.42.11 Constructor 703
- 9.42.12 Constructor(Handle as Integer) 703

9.42.14 didChangeText

Plugin Version: 18.1, Platform: macOS, Targets: Desktop only.

Function: Sends out necessary notifications when a text change completes.

Notes: Invoked automatically at the end of a series of changes, this method posts an `NSTextDidChangeNotification` to the default notification center, which also results in the delegate receiving an `NSText` delegate `textDidChange` message.

Subclasses implementing methods that change their text should invoke this method at the end of those methods. See `Subclassing NSTextView` for more information.

9.42.15 insertText(attributedString as NSAttributedStringMBS)

Plugin Version: 12.2, Platform: macOS, Targets: Desktop only.

Function: Inserts text into the receiver's text at the insertion point if there is one, otherwise replacing the selection.

Example:

```

dim textView as NSTextViewMBS // your view
dim a as new NSMutableAttributedStringMBS

if a.initWithString( "Hello World. This is just a little test." ) then

Dim NSFont as NSFontMBS = NSFontMBS.fontWithName("Arial", 24.0)
Dim NSColor as NSColorMBS = NSColorMBS.blueColor
Dim NSRange as NSRangeMBS = NSMakeRangeMBS( 0, 20)
Dim NSAttributes as New Dictionary

NSAttributes.value(NSAttributedStringMBS.NSFontAttributeName) = NSFont
NSAttributes.value(NSAttributedStringMBS.NSForegroundColorAttributeName) = NSColor

a.addAttributes( NSAttributes, NSRange)

textView.insertText a

// replace text with new one:
'textView.textStorage.setAttributedString a

end if

```

Notes: text: The string to insert. Can be either a string or an NSAttributedStringMBS object.

The inserted text is assigned the current typing attributes.

This method is the means by which text typed by the user enters an NSTextView. See the NSInputManager class and NSTextInput protocol specifications for more information.

This method is the entry point for inserting text typed by the user and is generally not suitable for other purposes. Programmatic modification of the text is best done by operating on the text storage directly. Because this method pertains to the actions of the user, the text view must be editable for the insertion to work.

See also:

- 9.42.16 insertText(text as string)

705

9.42.16 insertText(text as string)

Plugin Version: 12.2, Platform: macOS, Targets: Desktop only.

Function: Inserts text into the receiver’s text at the insertion point if there is one, otherwise replacing the selection.

Notes: text: The string to insert. Can be either an string or an NSAttributedStringMBS object.

The inserted text is assigned the current typing attributes.

This method is the means by which text typed by the user enters an NSTextView. See the NSInputManager class and NSTextInput protocol specifications for more information.

This method is the entry point for inserting text typed by the user and is generally not suitable for other purposes. Programmatic modification of the text is best done by operating on the text storage directly. Because this method pertains to the actions of the user, the text view must be editable for the insertion to work.

See also:

- 9.42.15 insertText(attributedString as NSAttributedStringMBS) 704

9.42.17 invalidateTextContainerOrigin

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Invalidates the calculated origin of the text container.

Notes: This method is invoked automatically; you should never need to invoke it directly. Usually called because the text view has been resized or the contents of the text container have changed.

9.42.18 loosenKerning

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Increases the space between glyphs in the receiver’s selection, or in all text if the receiver is a plain text view.

Notes: Kerning values are determined by the point size of the fonts in the selection.

9.42.19 lowerBaseline

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Lowers the baseline offset of selected text by 1 point, or of all text if the receiver is a plain text view.

Notes: As such, this method defines a more primitive operation than subscripting.

9.42.20 orderFrontLinkPanel

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Brings forward a panel allowing the user to manipulate links in the text view.

Notes: Available in Mac OS X v10.4 and later.

9.42.21 orderFrontListPanel

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Brings forward a panel allowing the user to manipulate text lists in the text view.

Notes: Available in Mac OS X v10.4 and later.

9.42.22 orderFrontSpacingPanel

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Brings forward a panel allowing the user to manipulate text line heights, interline spacing, and paragraph spacing, in the text view.

Notes: Available in Mac OS X v10.4 and later.

9.42.23 orderFrontSubstitutionsPanel

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Orders the substitution panel to the front.

Notes: Available on Mac OS X 10.6 or newer.

9.42.24 orderFrontTablePanel

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Brings forward a panel allowing the user to manipulate text tables in the text view.

9.42.25 outline

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Adds the outline attribute to the selected text attributes if absent; removes the attribute if present.

Notes: If there is a selection and the first character of the selected range has a non-zero stroke width, or if there is no selection and the typing attributes have a non-zero stroke width, then the stroke width is removed; otherwise the value of `NSStrikeWidthAttributeName` is set to the default value for outline (3.0).

Operates on the selected range if the receiver contains rich text. For plain text the range is the entire contents of the receiver.

9.42.26 `pasteAsPlainText`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Inserts the contents of the pasteboard into the receiver's text as plain text.

Notes: This method behaves analogously to `insertText`.

9.42.27 `pasteAsRichText`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: This action method inserts the contents of the pasteboard into the receiver's text as rich text, maintaining its attributes.

Notes: The text is inserted at the insertion point if there is one, otherwise replacing the selection.

9.42.28 `performFindPanelAction(FindAction as Integer)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Performs a find panel action specified by the sender's tag.

Example:

```
const NSFindPanelActionShowFindPanel = 1
```

```
dim n as NSTextViewMBS = TextArea1.NSTextViewMBS
n.usesFindPanel = true
n.performFindPanelAction(NSFindPanelActionShowFindPanel)
```

Notes: This is the generic action method for the find menu and find panel, and can be overridden to implement a custom find panel.

Possible values:

NSFindPanelActionShowFindPanel = 1
 NSFindPanelActionNext = 2
 NSFindPanelActionPrevious = 3
 NSFindPanelActionReplaceAll = 4
 NSFindPanelActionReplace = 5
 NSFindPanelActionReplaceAndFind = 6
 NSFindPanelActionSetFindString = 7
 NSFindPanelActionReplaceAllInSelection = 8
 NSFindPanelActionSelectAll = 9
 NSFindPanelActionSelectAllInSelection = 10

See also:

- 9.42.29 performFindPanelAction(sender as object) 709

9.42.29 performFindPanelAction(sender as object)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Performs a find panel action specified by the sender.

Notes: This is the generic action method for the find menu and find panel, and can be overridden to implement a custom find panel.

Sender could be a NSMenuItem or maybe also a NSView.

See also:

- 9.42.28 performFindPanelAction(FindAction as Integer) 708

9.42.30 raiseBaseline

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Raises the baseline offset of selected text by 1 point, or of all text if the receiver is a plain text view.

Notes: As such, this method defines a more primitive operation than superscripting.

9.42.31 replaceTextContainer(textContainer as NSTextContainerMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Replaces the text container for the group of text system objects containing the receiver, keeping the association between the receiver and its layout manager intact.

Notes: textContainer: The new text container. This method raises NSInvalidArgumentException if TextContainer is nil.

9.42.32 `shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string = "") as Boolean`

Plugin Version: 18.1, Platform: macOS, Targets: Desktop only.

Function: Initiates a series of delegate messages (and general notifications) to determine whether modifications can be made to the characters and attributes of the receiver, `text`.

Notes: `affectedCharRange`: The range of characters affected by the proposed change.

`replacementString`: The characters that will replace those in `affectedCharRange`. If only text attributes are being changed, `replacementString` is `""`.

Returns true to allow the change, false to prohibit it.

This method checks with the delegate as needed using `textShouldBeginEditing` and `shouldChangeTextInRange`.

This method must be invoked at the start of any sequence of user-initiated editing changes. If your subclass of `NSTextView` implements methods that modify the text, make sure to invoke this method to determine whether the change should be made. If the change is allowed, complete the change by invoking the `didChangeText` method.

If the receiver is not editable, this method automatically returns false. This result prevents instances in which a text view could be changed by user actions even though it had been set to be non-editable.

In macOS 10.4 and later, if there are multiple selections, this method acts on the first selected subrange.

See also:

- 9.42.97 `shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string) as boolean` 727

9.42.33 `showFindIndicatorForRange(charRange as NSRangeMBS)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Causes a temporary highlighting effect to appear around the visible portion (or portions) of the specified range.

Notes: `charRange`: The character range around which indicators appear.

This method supports lozenge-style indication of find results. The indicators automatically disappear after a certain period of time, or when the method is called again, or when any of a number of changes occur to the view (such as changes to text, view size, or view position).

This method does not itself scroll the specified range to be visible; any desired scrolling should be done before this method is called, first, because the method acts only on the visible portion of the specified range, and, second, because scrolling causes the indicators to disappear. Calling this method with a zero-length range always removes any existing indicators.

Available in OS X v10.5 and later.

9.42.34 startSpeaking

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Speaks the selected text, or all text if no selection.

9.42.35 stopSpeaking

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Stops the speaking of text.

9.42.36 tightenKerning

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Decreases the space between glyphs in the receiver's selection, or for all glyphs if the receiver is a plain text view.

Notes: Kerning values are determined by the point size of the fonts in the selection.

9.42.37 toggleAutomaticDashSubstitution

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Toggles automatic dash substitution.

Notes: Available on Mac OS X 10.6 or newer.

9.42.38 toggleAutomaticDataDetection

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Toggles automatic data detection.

Notes: Available on Mac OS X 10.6 or newer.

9.42.39 toggleAutomaticLinkDetection

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Changes the state of automatic link detection from enabled to disabled and vice versa.

Notes: Automatic link detection causes strings representing URLs typed in the view to be automatically made into links to those URLs.

Available in Mac OS X v10.5 and later.

9.42.40 toggleAutomaticQuoteSubstitution

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Changes the state of automatic quotation mark substitution from enabled to disabled and vice versa.

Notes: Automatic quote substitution causes ASCII quotation marks and apostrophes to be automatically replaced, on a context-dependent basis, with more typographically accurate symbols.

Available in Mac OS X v10.5 and later.

9.42.41 toggleAutomaticSpellingCorrection

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Toggles automatic spelling correction.

Notes: Available on Mac OS X 10.6 or newer.

9.42.42 toggleAutomaticTextReplacement

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Toggles automatic text replacements.

Notes: Available on Mac OS X 10.6 or newer.

9.42.43 toggleBold

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

Function: Toggles the use of a bold/non-bold font.

Example:

```
// some textview
dim textview as NSTextView = TextArea1.NSTextViewMBS

// switch between bold and non bold
textview.toggleBold
```

Notes: You can set this to continue typing with/without bold or change current selection. Can only provide bold if the font supports it.

9.42.44 toggleContinuousSpellChecking

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Toggles whether continuous spell checking is enabled for the receiver.

9.42.45 toggleGrammarChecking

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Changes the state of grammar checking from enabled to disabled and vice versa.

Notes: Available in Mac OS X v10.5 and later.

9.42.46 toggleItalic

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

Function: Toggles the use of an italic/non-italic font.

Example:

```
// some textview
dim textview as NSTextView = TextArea1.NSTextViewMBS

// switch between italic and non italic
textview.toggleItalic
```

Notes: You can set this to continue typing with/without bold or change current selection. Can only provide italic if the font supports it.

9.42.47 toggleSmartInsertDelete

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Changes the state of smart insert and delete from enabled to disabled and vice versa.

Notes: Controls whether the receiver inserts or deletes space around selected words so as to preserve proper spacing and punctuation.

Available in Mac OS X v10.5 and later.

9.42.48 toggleTraditionalCharacterShape

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Toggles the `NSCharacterShapeAttributeName` attribute at the current selection.

9.42.49 turnOffKerning

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets the receiver to use nominal glyph spacing for the glyphs in its selection, or for all glyphs if the receiver is a plain text view.

9.42.50 turnOffLigatures

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets the receiver to use only required ligatures when setting text, for the glyphs in the selection if the receiver is a rich text view, or for all glyphs if it's a plain text view.

9.42.51 updateDragTypeRegistration

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Updates the acceptable drag types of all text views associated with the receiver's layout manager.

Notes: If the receiver is editable and is a rich text view, causes all text views associated with the receiver's layout manager to register their acceptable drag types. If the text view isn't editable or isn't rich text, causes those text views to unregister their dragged types.

Subclasses can override this method to change the conditions for registering and unregistering drag types,

whether as a group or individually based on the current state of the text view. They should invoke this method when that state changes to perform the necessary update.

9.42.52 updateFontPanel

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Updates the Font panel to contain the font attributes of the selection.

Notes: Does nothing if the receiver doesn't use the Font panel. You should never need to invoke this method directly, but you can override it if needed to handle additional font attributes.

9.42.53 updateRuler

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Updates the ruler view in the receiver's enclosing scroll view to reflect the selection's paragraph and marker attributes.

Notes: Does nothing if the ruler isn't visible or if the receiver doesn't use the ruler. You should never need to invoke this method directly, but you can override this method if needed to handle additional ruler attributes.

9.42.54 useAllLigatures

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets the receiver to use all ligatures available for the fonts and languages used when setting text, for the glyphs in the selection if the receiver is a rich text view, or for all glyphs if it's a plain text view.

9.42.55 useStandardKerning

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Set the receiver to use pair kerning data for the glyphs in its selection, or for all glyphs if the receiver is a plain text view.

9.42.56 useStandardLigatures

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets the receiver to use the standard ligatures available for the fonts and languages used when setting text, for the glyphs in the selection if the receiver is a rich text view, or for all glyphs if it's a plain text view.

9.42.57 Properties

9.42.58 `acceptsGlyphInfo` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver accepts the glyph info attribute.

Notes: True if the receiver should accept the `NSGlyphInfoAttributeName` attribute from text input sources such as input methods and the pasteboard, false otherwise.

(Read and Write property)

9.42.59 `allowsDocumentBackgroundColorChange` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Sets whether the receiver allows its background color to change.

Notes: This corresponds to the background color of the entirety of the text view, not just to a selected range of text.

(Read and Write property)

9.42.60 `allowsImageEditing` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether image attachments should permit editing of their images.

Notes: True if image editing is allowed; otherwise, false.

For image editing to be allowed, the text view must be editable and the text attachment cell must support image editing.

Available in Mac OS X v10.5 and later.

(Read and Write property)

9.42.61 `allowsUndo` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether undo support is enabled.

Notes: (Read and Write property)

9.42.62 AutomaticDashSubstitutionEnabled as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Whether automatic dash substitution is enabled.

Example:

```
Sub CellGotFocus(row as Integer, column as Integer) Handles CellGotFocus
// disable dash substitution for current cell
Dim v As Variant = Me.ActiveCell
if v isa TextArea then
Dim t As TextArea = v
Dim n As NSTextViewMBS = t.NSTextViewMBS
nAutomaticDashSubstitutionEnabled = False
End If
End
```

Notes: Turning on automatic dash substitution enables automatic conversion of sequences of ASCII hyphen (-) characters to typographic dashes.

Available in Mac OS X v10.6 and later.

(Read and Write property)

9.42.63 AutomaticDataDetectionEnabled as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Whether automatic data detection is enabled.

Notes: Automatic data detection enables detection of dates, addresses, and phone numbers.

Available in Mac OS X v10.6 and later.

(Read and Write property)

9.42.64 AutomaticLinkDetectionEnabled as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Enables or disables automatic link detection.

Notes: If true, automatic link detection is enabled; if false, it is disabled.

Automatic link detection causes strings representing URLs typed in the view to be automatically made into links to those URLs.

Available in Mac OS X v10.5 and later.
(Read and Write property)

9.42.65 AutomaticQuoteSubstitutionEnabled as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether automatic quotation mark substitution is enabled.

Notes: True if automatic quotation mark substitution is enabled; otherwise, false.

Automatic quote substitution causes ASCII quotation marks and apostrophes to be automatically replaced, on a context-dependent basis, with more typographically accurate symbols.

Available in Mac OS X v10.5 and later.
(Read and Write property)

9.42.66 AutomaticSpellingCorrectionEnabled as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Whether automatic spelling correction is enabled.

Notes: Available in Mac OS X v10.6 and later.
(Read and Write property)

9.42.67 AutomaticTextReplacementEnabled as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Whether automatic text replacement is enabled.

Notes: Turning on automatic text replacement enables automatic substitution of a variety of static text items based on user preferences.

Available in Mac OS X v10.6 and later.
(Read and Write property)

9.42.68 backgroundColor as NSColorMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The background color.

Notes: (Read and Write property)

9.42.69 Bold as Boolean

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

Function: Whether the current typing uses a bold font.

Example:

```
// some textview
dim textview as NSTextView = TextArea1.NSTextViewMBS

// switch to bold font
textview.Bold = true
```

Notes: You can set this to continue typing with/without bold or change current selection.

Can only provide bold if the font supports it.

(Read and Write property)

9.42.70 ContinuousSpellCheckingEnabled as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver has continuous spell checking enabled.

Notes: True if the receiver has continuous spell checking enabled, otherwise, false.

See Textarea.WinSpellcheckingMBS property for Windows.

(Read and Write property)

9.42.71 defaultParagraphStyle as Variant

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Gets or sets the receiver's default paragraph style.

Example:

```
Dim n As NSTextViewMBS = TextArea1.NSTextViewMBS
```

```

Dim a As NSTextStorageMBS = n.textStorage

// get first paragraph style
Dim m As New NSMutableParagraphStyleMBS

// get old tab stops
Dim tabs() As NSTextTabMBS

For i As Integer = 0 To 11
Dim options As New Dictionary
Dim neu As New NSTextTabMBS(n.NSLeftTextAlignment, i * 50.0, options)
tabs.append neu
next

m.setTabStops tabs

// change for current text
a.addAttribute(a.NSParagraphStyleAttributeName, m, New NSRangeMBS(0, a.Length))

// and for future text
n.defaultParagraphStyle = m

```

Notes: Use with `NSParagraphStyleMBS` class.
(Read and Write property)

9.42.72 `displaysLinkToolTips` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether the text view automatically supplies the destination of a link as a tooltip for text that has a link attribute.

Notes: True if link tooltips are automatically displayed; otherwise, false.

The default value for this feature is true; clients who do not wish tooltips to be displayed automatically must explicitly disable it.

Available in Mac OS X v10.5 and later.
(Read and Write property)

9.42.73 `enabledTextCheckingTypes` as Int64

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: The default text checking types.

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

Can be `NSTextCheckingAllSystemTypes (&hfffffff)` or `NSTextCheckingAllCustomTypes (&hfffffff00000000)` or `NSTextCheckingAllTypes (&hfffffff)`.

(Read and Write property)

9.42.74 GrammarCheckingEnabled as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Whether or not grammar checking is enabled.

Notes: Available on Mac OS X 10.5 or newer.

If grammar checking is enabled, then it is performed alongside spell checking, whenever the text view checks spelling, whether continuously or manually.

(Read and Write property)

9.42.75 insertionPointColor as NSColorMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The color used to draw the insertion point.

Notes: (Read and Write property)

9.42.76 isCoalescingUndo as boolean

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns whether undo coalescing is in progress.

Notes: True if undo coalescing is in progress, otherwise false.

Available in OS X v10.6 and later.

(Read only property)

9.42.77 Italic as Boolean

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

Function: Whether the current typing uses a italic font.

Example:

```
// some textView
```

```
dim textview as NSTextView = TextArea1.NSTextViewMBS
```

```
// switch to italic font  
textview.Italic = true
```

Notes: You can set this to continue typing with/without italic or change current selection. Can only provide italic if the font supports it. (Read and Write property)

9.42.78 layoutManager as NSLayoutManagerMBS

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Returns the layout manager that lays out text for the receiver's text container.

Notes: The layout manager that lays out text for the receiver's text container, or nil if there's no such object, such as when a text view isn't linked into a group of text objects. (Read only property)

9.42.79 linkTextAttributes as dictionary

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Gets and sets the attributes used to draw the onscreen presentation of link text.

Notes: A dictionary of attributes corresponding to the onscreen presentation of link text.

Link text attributes are applied as temporary attributes to any text with a link attribute. Candidates include those attributes that do not affect layout.

In applications created prior to OS X v10.3, the default value is an empty dictionary. In applications created with OS X v10.3 or greater, the default attributes specify blue text with a single underline and the pointing hand cursor.

(Read and Write property)

9.42.80 markedTextAttributes as dictionary

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Gets or sets the attributes used to draw marked text.

Notes: A dictionary of attributes used to draw marked text. Text color, background color, and underline are the only supported attributes for marked text.

(Read and Write property)

9.42.81 RTFData as Memoryblock

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Get or set the textview content as RTF data.

Notes: Works only for Cocoa and uses RTF parser/generator from Apple.

See also RTFDataMBS in StyledText, RTFDataMBS and WinRTFDataMBS in TextArea and Desktop-TextArea controls.

(Read and Write property)

9.42.82 RulerVisible as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether the scroll view enclosing the text views sharing the receiver's layout manager shows its ruler.

Notes: True if the scroll view enclosing the text views sharing the receiver's layout manager shows its ruler, false otherwise. The default is false.

(Read and Write property)

9.42.83 selectedTextAttributes as dictionary

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Gets or sets the attributes used to indicate the selection.

Example:

```
// underline selected text in TextArea1

dim t as NSTextViewMBS = TextArea1.NSTextViewMBS
dim s as NSTextStorageMBS = t.textStorage

dim d as Dictionary = t.selectedTextAttributes
d.Value(NSAttributedStringMBS.NSStrikethroughStyleAttributeName) = s.NSUnderlineStyleSingle
t.selectedTextAttributes = d
```

Notes: A dictionary of attributes used to indicate the selection. Text color, background color, and underline are the only supported attributes for selected text.

(Read and Write property)

9.42.84 `smartInsertDeleteEnabled` as `boolean`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Whether the view inserts or deletes space around selected words so as to preserve proper spacing and punctuation.

Notes: (Read and Write property)

9.42.85 `spellCheckerDocumentTag` as `Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a tag identifying the text view's text as a document for the spell checker server.

Notes: The document tag is obtained by sending a `uniqueSpellDocumentTag` message to the spell server the first time this method is invoked for a particular group of text views. See the `NSSpellChecker` and `NSSpellServer` class specifications for more information on how this tag is used.

(Read only property)

9.42.86 `textContainer` as `NSTextContainerMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The text container.

Notes: The receiver uses the layout manager and text storage of a `TextContainer`.

Special Considerations

This method is invoked automatically when you create a text view; you should never invoke it directly, but might want to override it. To change the text view for an established group of text system objects, use `TextView` setter on the text container. To replace the text container for a text view and maintain the view's association with the existing layout manager and text storage, use `replaceTextContainer`.

(Read and Write property)

9.42.87 `textContainerInset` as `NSSizeMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: The empty space the receiver leaves around its text container.

Example:

```
dim n as NSTextViewMBS = TextArea1.NSTextViewMBS
```

```
n.textContainerInset = NSMakeSizeMBS(-3,0)
```

Notes: It is possible to set the text container and view sizes and resizing behavior so that the inset cannot be maintained exactly, although the text system tries to maintain the inset wherever possible. In any case, the textContainerOrigin and size of the text container are authoritative as to the location of the text container within the view.

The text itself can have an additional inset, inside the text container, specified by the setLineFragmentPadding method of NSTextContainer.

(Read and Write property)

9.42.88 textContainerOrigin as NSPointMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns the origin of the receiver's text container.

Notes: The origin of the receiver's text container, which is calculated from the receiver's bounds rectangle, container inset, and the container's used rect.

(Read only property)

9.42.89 textStorage as NSTextStorageMBS

Plugin Version: 12.2, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's text storage object.

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: (Read only property)

9.42.90 `typingAttributes` as dictionary

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Get or set typing attributes.

Notes: Typing attributes are reset automatically whenever the selection changes. However, if you add any user actions that change text attributes, the action should use this method to apply those attributes afterwards. User actions that change attributes should always set the typing attributes because there might not be a subsequent change in selection before the next typing.

(Read and Write property)

9.42.91 `usesFindBar` as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether to use the find bar for this text view.

Notes: The value of this property is true if the find bar is used for this text view; otherwise false.

See `NSTextFinderMBS` class for information about the find bar.

A text view can use either a find panel or a find bar. If `usesFindBar` is set to true, `usesFindPanel` is set to false and vice versa.

(Read and Write property)

9.42.92 `usesFindPanel` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver allows for a find panel.

Notes: (Read and Write property)

9.42.93 `usesFontPanel` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Controls whether the text views sharing the receiver's layout manager use the Font panel and Font menu.

Notes: (Read and Write property)

9.42.94 `usesInspectorBar` as Boolean

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: Whether this text view uses the inspector bar.

Example:

```
dim t as NSTextViewMBS = textarea1.NSTextViewMBS
t.usesInspectorBar = true
```

Notes: Available in OS X v10.7 and later.

(Read and Write property)

9.42.95 usesRuler as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Whether the text views sharing the receiver's layout manager use a ruler.

Notes: True to cause text views sharing the receiver's layout manager to respond to NSRulerView client messages and to paragraph-related menu actions, and update the ruler (when visible) as the selection changes with its paragraph and tab attributes, otherwise false.

Text views must use a ruler to respond to Format menu commands. If a set of text views don't use the ruler, the ruler is hidden, and the text views disallow paragraph attribute changes. By default, text view objects use the ruler.

(Read and Write property)

9.42.96 Events

9.42.97 shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string) as boolean

Plugin Version: 10.1, Platform: macOS, Targets: .

Function: Sent when a text view needs to determine if text in a specified range should be changed.

Notes: affectedCharRange: The range of characters to be replaced.

replacementString: The characters that will replace the characters in affectedCharRange; nil if only text attributes are being changed.

Return true to allow the replacement, or false to reject the change.

See also:

- 9.42.32 shouldChangeTextInRange(affectedCharRange as NSRangeMBS, replacementString as string = "") as Boolean 710

9.42.98 textViewDidChangeSelection

Plugin Version: 8.4, Platform: macOS, Targets: .

Function: Sent when the selection changes in the text view.

9.42.99 Constants

Constants

Constant	Value	Description
NSFindPanelSubstringMatchTypeContains	0	One of the constants to specify the type of substring matching Find panel. Finds a word containing the search string. Available in Mac OS X v10.5 and later.
NSFindPanelSubstringMatchTypeEndsWith	3	One of the constants to specify the type of substring matching Find panel. Finds a word ending with the search string. Available in Mac OS X v10.5 and later.
NSFindPanelSubstringMatchTypeFullWord	2	One of the constants to specify the type of substring matching Find panel. Finds a word exactly matching the search string. Available in Mac OS X v10.5 and later.
NSFindPanelSubstringMatchTypeStartsWith	1	One of the constants to specify the type of substring matching Find panel. Finds a word starting with the search string. Available in Mac OS X v10.5 and later.
NSSelectByCharacter	0	One of the constants to specify how much the text view extends when the user drags the mouse. Extends the selection character by character.
NSSelectByParagraph	2	One of the constants to specify how much the text view extends when the user drags the mouse. Extends the selection paragraph by paragraph.
NSSelectByWord	1	One of the constants to specify how much the text view extends when the user drags the mouse. Extends the selection word by word.
NSSelectionAffinityDownstream	1	One of the constants to specify the preferred direction of selection. The selection is moving toward the bottom of the document.
NSSelectionAffinityUpstream	0	One of the constants to specify the preferred direction of selection. The selection is moving toward the top of the document.

9.43 class SearchField

9.43.1 class SearchField

Plugin Version: 20.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The built-in search field control in Xojo.

Notes: Only for Xojo 2020r2 or newer.

9.43.2 Methods

9.43.3 NSSearchFieldMBS as NSSearchFieldMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Queries search field object for the control.

Notes: Only for Xojo 2020r2 or newer.

9.44 class SegmentedControl

9.44.1 class SegmentedControl

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Build in Segmented Control class in Xojo.

9.44.2 Methods

9.44.3 NSSegmentedControlMBS as NSSegmentedControlMBS

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Creates a NSSegmentedControlMBS object for the given control.

Example:

```
SegmentedControl1.NSSegmentedControlMBS.selectedSegment = 0
```

Notes: This way you can manipulate Cocoa controls directly.

9.45 class Statictext

9.45.1 class Statictext

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: The built in Statictext class in Xojo.

9.45.2 Methods

9.45.3 NSTextFieldMBS as NSTextFieldMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a NSTextFieldMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

9.46 class TextArea

9.46.1 class TextArea

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

Function: The built in textarea class in Xojo.

Example:

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

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

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

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

if f<>Nil then

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

  b.Write s

end if
```

Notes: Requires RB 2009r4 or newer.

9.46.2 Methods

9.46.3 NSTextFieldMBS as NSTextFieldMBS

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Creates a NSTextFieldMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

Xojo uses a NSTextField for text areas without style and without multiline.

9.46.4 NSTextViewMBS as NSTextViewMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a NSTextViewMBS object for the given control.

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: This way you can manipulate Cocoa controls directly.

Xojo uses a NSTextField for text areas without style and without multiline.

9.46.5 Properties

9.46.6 RTFDataMBS as Memoryblock

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Get or set the textview content as RTF data.

Example:

```
dim rtf as MemoryBlock = TextArea1.RTFDataMBS
TextArea2.RTFDataMBS = rtf
```

Notes: Works only for Cocoa and uses RTF parser/generator from Apple.

See also RTFDataMBS in StyledText, WinRTFDataMBS in TextArea and DesktopTextArea controls and RTFData in NSTextViewMBS class.

(Read and Write computed property)

Chapter 10

Cocoa Drawing

10.1 class NSColorPanelMBS

10.1.1 class NSColorPanelMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: A class to control a Color Panel (floating window).

Notes: If you use SelectColor in your application, this color panel will be converted in a dialog which makes this class useless.

You should only have one instance of this class in your application.

If you compile for Cocoa, a TextField/TextArea automatically registers for color panel. So in order to avoid them updating text color with color panel selection, clear the focus by calling window.clearfocus method.

This class does only work on desktop computers, not in a webbrowser.

See WinColorPickerMBS class for Windows.

Subclass of the NSPanelMBS class.

Blog Entries

- [Xojo 2021r1.1 available](#)
- [Windows Font Dialog](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr4](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr1](#)
- [MBS Real Studio Plugins, version 11.2pr9](#)

10.1.2 Methods

10.1.3 `attachColorList(list as NSColorListMBS)`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Attaches the given color list to the panel.

Notes: An application should use this method to add an NSColorList saved with a document in its file package or in a directory other than NSColorList's standard search directories.

10.1.4 Constructor

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The constructor to create a new color panel.

10.1.5 `detachColorList(list as NSColorListMBS)`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Detaches the color list from the panel.

10.1.6 `GetColor(byref red as single, byref green as single, byref blue as single, byref alpha as single) as boolean`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current color in the RGB color model.

Notes: Values from 0.0 to 1.0.

Returns true if the values are valid.

See also:

- 10.1.21 `getColor as NSColorMBS`

740

10.1.7 `GetColorFromDrag as color`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns color values from the drag.

Notes: If you receive a drag flavor which can't be handled, it may be a NSColor.

In this case only this method will be successful.

See also:

- 10.1.8 GetColorFromDrag(byref red as single, byref green as single, byref blue as single, byref alpha as single) as boolean 737

10.1.8 GetColorFromDrag(byref red as single, byref green as single, byref blue as single, byref alpha as single) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns color values from the drag.

Notes: If you receive a drag flavor which can't be handled, it may be a NSColor.

In this case only this method will be successful.

See also:

- 10.1.7 GetColorFromDrag as color 736

10.1.9 orderFrontColorPanel

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Brings up the color panel, an instance of NSColorPanel.

Notes: If the NSColorPanel object does not exist yet, this method creates one. This method is typically invoked when the user chooses Colors from a menu.

10.1.10 SetColor(red as single, green as single, blue as single, alpha as single)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Sets the color.

Notes: Values from 0.0 to 1.0.

See also:

- 10.1.11 setColor(value as NSColorMBS) 737

10.1.11 setColor(value as NSColorMBS)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Sets the color.

See also:

- 10.1.10 SetColor(red as single, green as single, blue as single, alpha as single) 737

10.1.12 setContinuous(value as boolean)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Whether you want to receive Changed events while the user chooses the color.

Notes: Set value to true to have the receiver calls the Changed event continuously as the color of the NSColorPanel is set by the user; otherwise false.

10.1.13 setMode(value as Integer)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Sets the mode of the panel the mode is one of the modes allowed by the color mask.

Notes: Constants:

```
const NSGrayModeColorPanel      = 0
const NSRGBModeColorPanel      = 1
const NSCMYKModeColorPanel     = 2
const NSHSBModeColorPanel      = 3
const NSCustomPaletteModeColorPanel = 4
const NSColorListModeColorPanel = 5
const NSWheelModeColorPanel     = 6
const NSCrayonModeColorPanel   = 7
```

10.1.14 SetPickerMode(value as Integer)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Specifies the color panel's initial picker.

Notes: Constants:

```
NSColorPanelGrayModeMask      = &h00000001
NSColorPanelRGBModeMask      = &h00000002
NSColorPanelCMYKModeMask     = &h00000004
NSColorPanelHSBModeMask      = &h00000008
NSColorPanelCustomPaletteModeMask = &h00000010
NSColorPanelColorListModeMask = &h00000020
NSColorPanelWheelModeMask    = &h00000040
NSColorPanelCrayonModeMask   = &h00000080
NSColorPanelAllModesMask     = &h0000ffff
```

10.1.15 setShowsAlpha(value as boolean)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Tells the panel whether or not to show alpha values and an opacity slider.

Notes: Note that calling the NSColor method setIgnoresAlpha with a value of true overrides any value set with this method.

10.1.16 SharedColorPanelExists as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether the NSColorPanel has been created already.

Notes: True if the NSColorPanel has been created already; otherwise false.

10.1.17 Properties

10.1.18 accessoryView as NSViewMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The accessory view displayed in the panel.

Notes: The accessory view can be any custom view you want to display with NSColorPanel, such as a view offering color blends in a drawing program. The accessory view is displayed below the color picker and above the color swatches in the NSColorPanel. The NSColorPanel automatically resizes to accommodate the accessory view.

(Read and Write property)

10.1.19 alpha as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's current alpha value based on its opacity slider.

Notes: Value is in the range between 0.0 and 1.0.

This is 1.0 (opaque) if the panel has no opacity slider.

(Read only property)

10.1.20 ColorValue as Color

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current color as a Xojo color in the RGB model.

Notes: (Read and Write property)

10.1.21 getColor as NSColorMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current color.

Notes: The color returned can be RGB or something else.

(Read and Write property)

See also:

- 10.1.6 GetColor(byref red as single, byref green as single, byref blue as single, byref alpha as single) as boolean 736

10.1.22 getColorAsRGB as NSColorMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The current color.

Notes: Returned NSColorMBS object is in RGB color mode.

(Read only property)

10.1.23 isContinuous as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether the class continuously calls the Changed event.

Notes: Returns true if the receiver continuously calls the Changed event as the user manipulates the color picker; otherwise false.

(Read and Write property)

10.1.24 mode as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns the color picker mode.

Notes: Constants:

(Read and Write property)

```
const NSGrayModeColorPanel      = 0
const NSRGBModeColorPanel       = 1
const NSCMYKModeColorPanel      = 2
const NSHSBModeColorPanel       = 3
const NSCustomPaletteModeColorPanel = 4
const NSColorListModeColorPanel = 5
const NSWheelModeColorPanel     = 6
const NSCrayonModeColorPanel    = 7
```

10.1.25 showsAlpha as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether or not the panel shows alpha values and an opacity slider.

Notes: Note that calling the NSColor method setIgnoresAlpha with a value of true overrides any value set with setShowAlpha.

(Read and Write property)

10.1.26 Events

10.1.27 Changed

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: The color changed.

Notes: May not fire in the RB IDE.

10.1.28 DidMove

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The color panel did move.

10.1.29 GotFocus

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The color panel got focus.

10.1.30 Hidden

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The color panel is hidden.

10.1.31 LostFocus

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The color panel lost focus.

10.1.32 Shown

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The color panel shows.

10.1.33 WillClose

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The color panel will close.

10.1.34 Constants

Modes

Constant	Value	Description
NSCMYKModeColorPanel	2	Cyan-yellow-magenta-black
NSColorListModeColorPanel	5	Custom color list
NSCrayonModeColorPanel	7	Crayons.
NSCustomPaletteModeColorPanel	4	Custom palette
NSGrayModeColorPanel	0	Grayscale-alpha
NSHSBModeColorPanel	3	Hue-saturation-brightness
NSNoModeColorPanel	-1	Indicates no color panel mode. Available in OS X version 10.5 and later.
NSRGBModeColorPanel	1	Red-green-blue
NSWheelModeColorPanel	6	Color wheel

Masks

Constant	Value	Description
NSColorPanelAllModesMask	&h0000ffff	All of the above.
NSColorPanelCMYKModeMask	&h00000004	Cyan-yellow-magenta-black.
NSColorPanelColorListModeMask	&h00000020	Custom color list.
NSColorPanelCrayonModeMask	&h00000080	Crayons.
NSColorPanelCustomPaletteModeMask	&h00000010	Custom palette.
NSColorPanelGrayModeMask	&h00000001	Grayscale-alpha.
NSColorPanelHSBModeMask	&h00000008	Hue-saturation-brightness.
NSColorPanelRGBModeMask	&h00000002	Red-green-blue.
NSColorPanelWheelModeMask	&h00000040	Color wheel.

10.2 class NSGraphicsMBS

10.2.1 class NSGraphicsMBS

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: The Cocoa class for drawing.

Example:

```
// make new image
dim myImage as new NSImageMBS(500,500)
dim myGraphics as new NSGraphicsMBS(myImage)

// make logo image
dim myPicture as Picture = LogoMBS(500)
dim anotherImage as new NSImageMBS(myPicture)

// draw logo image to new image
myGraphics.drawRect(anotherImage, 0, 0, myImage.width, myImage.height, 0, 0, anotherImage.width,
anotherImage.height, myGraphics.NSCompositeSourceOver, 1.0)
myGraphics = nil // flush drawing

// save to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim b as BinaryStream = BinaryStream.Create(f, True)
if b<>nil then
b.Write myImage.JPEGRepresentation
b.Close
end if
```

Notes: The plugin often provides in events such objects for drawing. In that case please only use the object in the event and don't store it for later use. It is only valid with in a draw event.

Internally this is a NSGraphicsContext object.

If you create objects on your own, make sure you only use the methods while the object is valid.

Blog Entries

- [MBS Xojo Plugins, version 19.6pr3](#)
- [MBS Xojo / Real Studio Plugins, version 15.4pr4](#)
- [Apply fonts to font PopupMenu](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr12](#)
- [MBS Real Studio Plugins, version 12.5pr4](#)

- [MBS Real Studio Plugins, version 12.4pr7](#)
- [MBS Real Studio Plugins, version 12.3pr7](#)
- [MBS Real Studio Plugins, version 12.3pr6](#)
- [MonkeyBread Software Releases the MBS Real Studio plug-ins in version 12.2](#)
- [MBS Plugins 10.3 Release Notes](#)

Xojo Developer Magazine

- [10.5, page 10: News](#)

10.2.2 Methods

10.2.3 addClip(path as NSBezierPathMBS)

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Intersects the area enclosed by the receiver's path with the clipping path of the current graphics context and makes the resulting shape the current clipping path.

Notes: This method uses the current winding rule to determine the clipping shape of the receiver. This method does not affect the receiver's path.

10.2.4 boundingRectWithSize(text as NSAttributedStringMBS, size as NSSizeMBS, options as Integer = 0) as NSRectMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Calculates and returns bounding rectangle for the receiver drawn using the options specified, within the given rectangle in the current graphics context.

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))

// query size
dim g as new NSGraphicsMBS(Canvas1.NSViewMBS)
dim r as NSRectMBS = g.boundingRectWithSize(m, new NSSizeMBS(canvas1.Width, canvas1.Height),
g.NSStringDrawingUsesLineFragmentOrigin)

MsgBox r.String
```

Notes: size: The size of the rectangle to draw in.
options: The string drawing options.

Returns the bounding rectangle in the current graphics context.
The origin of the rectangle returned from this method is the first glyph origin.
See also:

- 10.2.5 `boundingRectWithSize(text as string, size as NSSizeMBS, options as Integer = 0, DicAttributes as dictionary = nil) as NSRectMBS` 746

10.2.5 `boundingRectWithSize(text as string, size as NSSizeMBS, options as Integer = 0, DicAttributes as dictionary = nil) as NSRectMBS`

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Calculates and returns the bounding rect for the text drawn using the given options and display characteristics, within the specified rectangle in the current graphics context.

Example:

```
Dim NSGraphics as New NSGraphicsMBS()
dim size as NSSizeMBS = new NSSizeMBS(100,100)
dim text as string = "Hello World. How are you? I'm fine. This is just a test string."
dim options as Integer = NSGraphics.NSStringDrawingUsesLineFragmentOrigin
Dim rect as NSRectMBS = NSGraphics.boundingRectWithSize(text, size, options)
```

```
MsgBox str(Rect.Width)+" "+str(Rect.Height)
```

Notes: text: the text to use for calculation.
size: The size of the rectangle to draw in.
options: String drawing options.
attributes: A dictionary of text attributes to be applied to the string. These are the same attributes that can be applied to an NSAttributedString object, but in the case of Strings, the attributes apply to the entire string, rather than ranges within the string.

Returns the bounding rect for the receiver drawn using the given options and display characteristics. The rect origin returned from this method is the first glyph origin.

Available in Mac OS X v10.4 and later.

See `NSStringDrawing*` constants. Use `NSStringDrawingUsesLineFragmentOrigin` to switch to multiline mode.

See also:

- 10.2.4 boundingRectWithSize(text as NSAttributedStringMBS, size as NSSizeMBS, options as Integer = 0) as NSRectMBS 745

10.2.6 clipRect(r as NSRectMBS)

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Intersects the specified rectangle with the clipping path of the current graphics context and makes the resulting shape the current clipping path

Notes: r: The rectangle to intersect with the current clipping path.

10.2.7 concat(transform as NSAffineTransformMBS)

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

Function: Concats this transform to the current transform of the graphics environment.

10.2.8 ConcatTransform(NSAffineTransform as Variant)

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Appends the receiver's matrix to the current transformation matrix stored in the current graphics context, replacing the current transformation matrix with the result.

Notes: Please use saveGraphicsState so you can restore the state before applying matrix for other drawings.

Concatenation is performed by matrix multiplication.

If this method is invoked from within an NSView drawRect method, then the current transformation matrix is an accumulation of the screen, window, and any superview's transformation matrices. Invoking this method defines a new user coordinate system whose coordinates are mapped into the former coordinate system according to the receiver's transformation matrix. To undo the concatenation, you must invert the receiver's matrix and invoke this method again.

NSAffineTransform must be a NSAffineTransformMBS object.

10.2.9 Constructor

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Initializes the current graphics context of the current thread.

See also:

- 10.2.10 Constructor(targetImage as NSBitmapImageRepMBS) 748
- 10.2.11 Constructor(targetImage as NSImageMBS) 749
- 10.2.12 Constructor(targetView as NSViewMBS) 749
- 10.2.13 Constructor(targetWindow as DesktopWindow) 750
- 10.2.14 Constructor(targetWindow as NSWindowMBS) 750
- 10.2.15 Constructor(targetWindow as window) 751

10.2.10 Constructor(targetImage as NSBitmapImageRepMBS)

Plugin Version: 12.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new graphics context for drawing into a bitmap image representation.

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: Please make sure the graphics object is destroyed (Set to nil) so the drawings flush to the image. Returns nil on any error.

See also:

- 10.2.9 Constructor 747
- 10.2.11 Constructor(targetImage as NSImageMBS) 749
- 10.2.12 Constructor(targetView as NSViewMBS) 749
- 10.2.13 Constructor(targetWindow as DesktopWindow) 750
- 10.2.14 Constructor(targetWindow as NSWindowMBS) 750
- 10.2.15 Constructor(targetWindow as window) 751

10.2.11 Constructor(targetImage as NSImageMBS)

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

Function: Creates a new graphics context for drawing into an image.

Example:

```
// make new image
dim myImage as new NSImageMBS(500,500)
dim myGraphics as new NSGraphicsMBS(myImage)

// make logo image
dim myPicture as Picture = LogoMBS(500)
dim anotherImage as new NSImageMBS(myPicture)

// draw logo image to new image
myGraphics.drawInRect(anotherImage, 0, 0, myImage.width, myImage.height, 0, 0, anotherImage.width,
anotherImage.height, myGraphics.NSCompositeSourceOver, 1.0)
myGraphics = nil // flush drawing

// save to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim b as BinaryStream = BinaryStream.Create(f, True)
if b<>nil then
b.Write myImage.JPEGRepresentation
b.Close
end if
```

Notes: Please make sure the graphics object is destroyed (Set to nil) so the drawings flush to the image.
See also:

- 10.2.9 Constructor 747
- 10.2.10 Constructor(targetImage as NSBitmapImageRepMBS) 748
- 10.2.12 Constructor(targetView as NSViewMBS) 749
- 10.2.13 Constructor(targetWindow as DesktopWindow) 750
- 10.2.14 Constructor(targetWindow as NSWindowMBS) 750
- 10.2.15 Constructor(targetWindow as window) 751

10.2.12 Constructor(targetView as NSViewMBS)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new graphics context for drawing into a Cocoa view.

Notes: Please make sure the graphics object is destroyed (Set to nil) so the drawings flush to the window.
See also:

- 10.2.9 Constructor 747
- 10.2.10 Constructor(targetImage as NSBitmapImageRepMBS) 748
- 10.2.11 Constructor(targetImage as NSImageMBS) 749
- 10.2.13 Constructor(targetWindow as DesktopWindow) 750
- 10.2.14 Constructor(targetWindow as NSWindowMBS) 750
- 10.2.15 Constructor(targetWindow as window) 751

10.2.13 Constructor(targetWindow as DesktopWindow)

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

Function: Creates a new graphics context for drawing into a window.

See also:

- 10.2.9 Constructor 747
- 10.2.10 Constructor(targetImage as NSBitmapImageRepMBS) 748
- 10.2.11 Constructor(targetImage as NSImageMBS) 749
- 10.2.12 Constructor(targetView as NSViewMBS) 749
- 10.2.14 Constructor(targetWindow as NSWindowMBS) 750
- 10.2.15 Constructor(targetWindow as window) 751

10.2.14 Constructor(targetWindow as NSWindowMBS)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new graphics context for drawing into a window.

See also:

- 10.2.9 Constructor 747
- 10.2.10 Constructor(targetImage as NSBitmapImageRepMBS) 748
- 10.2.11 Constructor(targetImage as NSImageMBS) 749
- 10.2.12 Constructor(targetView as NSViewMBS) 749
- 10.2.13 Constructor(targetWindow as DesktopWindow) 750
- 10.2.15 Constructor(targetWindow as window) 751

10.2.15 Constructor(targetWindow as window)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new graphics context for drawing into a window.

See also:

- 10.2.9 Constructor 747
- 10.2.10 Constructor(targetImage as NSBitmapImageRepMBS) 748
- 10.2.11 Constructor(targetImage as NSImageMBS) 749
- 10.2.12 Constructor(targetView as NSViewMBS) 749
- 10.2.13 Constructor(targetWindow as DesktopWindow) 750
- 10.2.14 Constructor(targetWindow as NSWindowMBS) 750

10.2.16 drawAtPoint(image as NSImageMBS, x as Double, y as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Draws all or part of the image at the specified point in the current coordinate system.

Notes: x/y: The location in the current coordinate system at which to draw the image.

sx/sy/sw/sh: The source rectangle specifying the portion of the image you want to draw. The coordinates of this rectangle are specified in the image's own coordinate system. If you pass in zeros, the entire image is drawn.

operation: The compositing operation to use when drawing the image. See the NSCompositingOperation constants.

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.

The image content is drawn at its current resolution and is not scaled unless the CTM of the current coordinate system itself contains a scaling factor. The image is otherwise positioned and oriented using the current coordinate system.

For Operation you use the Composite constants in this class.

In the Cocoa world the y axis is reversed. y=0 is on the bottom.

See also:

- 10.2.17 drawAtPoint(text as NSAttributedStringMBS, point as NSPointMBS) 752
- 10.2.18 drawAtPoint(text as string, point as NSPointMBS, DicAttributes as dictionary = nil) 752

10.2.17 drawAtPoint(text as NSAttributedStringMBS, point as NSPointMBS)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Draws the receiver with its font and other display attributes at the given point in the currently focused view.

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

// draw in Canvas
dim g as new NSGraphicsMBS(Canvas1.NSViewMBS)

g.drawAtPoint m, new NSPointMBS(20,20)
```

Notes: point: The point in the current view to draw the text.

The width (height for vertical layout) of the rendering area is unlimited, unlike drawInRect, which uses a bounding rectangle. As a result, this method renders the text in a single line.

Don't invoke this method when no NSView is focused.

See also:

- 10.2.16 drawAtPoint(image as NSImageMBS, x as Double, y as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double) 751
- 10.2.18 drawAtPoint(text as string, point as NSPointMBS, DicAttributes as dictionary = nil) 752

10.2.18 drawAtPoint(text as string, point as NSPointMBS, DicAttributes as dictionary = nil)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Draws the text with the font and other display characteristics of the given attributes, at the specified point in the currently focused view.

Notes: Point: The origin for the bounding box for drawing the string. If the focused view is flipped, the origin is the upper-left corner of the drawing bounding box; otherwise, the origin is the lower-left corner.

attributes: A dictionary of text attributes to be applied to the string. These are the same attributes that

can be applied to an NSAttributedString object, but in the case of strings, the attributes apply to the entire string, rather than ranges within the string.

The width (height for vertical layout) of the rendering area is unlimited, unlike drawInRect, which uses a bounding rectangle. As a result, this method renders the text in a single line.

You should only invoke this method when an NSView object has focus.

See also:

- 10.2.16 drawAtPoint(image as NSImageMBS, x as Double, y as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double) 751
- 10.2.17 drawAtPoint(text as NSAttributedStringMBS, point as NSPointMBS) 752

10.2.19 drawInRect(image as NSImageMBS, x as Double, y as Double, w as Double, h as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: draws the image into the given rectangle with the given source rectangle and the given mode.

Example:

```
// make new image
dim myImage as new NSImageMBS(500,500)
dim myGraphics as new NSGraphicsMBS(myImage)

// make logo image
dim myPicture as Picture = LogoMBS(500)
dim anotherImage as new NSImageMBS(myPicture)

// draw logo image to new image
myGraphics.drawInRect(anotherImage, 0, 0, myImage.width, myImage.height, 0, 0, anotherImage.width,
anotherImage.height, myGraphics.NSCompositeSourceOver, 1.0)
myGraphics = nil // flush

// save to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim b as BinaryStream = BinaryStream.Create(f, True)
if b<>nil then
b.Write myImage.JPEGRepresentation
b.Close
end if
```

Notes: For Operation you use the Composite constants in this class.

In the Cocoa world the y axis is reversed. y=0 is on the bottom.

See also:

- 10.2.20 drawInRect(text as NSAttributedStringMBS, rect as NSRectMBS) 754
- 10.2.21 drawInRect(text as string, rect as NSRectMBS, DicAttributes as dictionary = nil) 755

10.2.20 drawInRect(text as NSAttributedStringMBS, rect as NSRectMBS)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Draws the attributed string within the given rectangle in the currently view, clipping the text layout to this rectangle.

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

// draw in Canvas
dim g as new NSGraphicsMBS(Canvas1.NSViewMBS)

g.drawInRect m, new NSRectMBS(20,20, 100, 100)
```

Notes: rect: The rectangle in which to draw.

Text is drawn within rect according to its line sweep direction; for example, Arabic text will begin at the right edge and potentially be clipped on the left.

The rect parameter determines how many glyphs are typeset within the width of a line, but it's possible for a portion of a glyph to appear outside the area of rect if the image bounding box of the particular glyph exceeds its typographic bounding box.

If the focus view is flipped, the text origin is set at the upper-left corner of the drawing bounding box; otherwise the origin is set at the lower-left corner. For text rendering, whether the view coordinates are flipped or not doesn't affect the flow of line layout, which goes from top to bottom. However, it affects the interpretation of the text origin. So, for example, if the rect argument is { 0.0, 0.0, 100.0, 100.0 }, the text origin is { 0.0, 0.0 } when the view coordinates are flipped and { 0.0, 100.0 } when not.

Don't invoke this method when no `NSView` is focused.

See also:

- 10.2.19 `drawInRect(image as NSImageMBS, x as Double, y as Double, w as Double, h as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)` 753
- 10.2.21 `drawInRect(text as string, rect as NSRectMBS, DicAttributes as dictionary = nil)` 755

10.2.21 `drawInRect(text as string, rect as NSRectMBS, DicAttributes as dictionary = nil)`

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Draws the text with the font and other display characteristics of the given attributes, within the specified rectangle in the currently focused `NSView`.

Notes: `text`: The text to draw.

`Rect`: The rectangle in which to draw the string.

`attributes`: A dictionary of text attributes to be applied to the string. These are the same attributes that can be applied to an `NSAttributedString` object, but in the case of strings, the attributes apply to the entire string, rather than ranges within the string.

The rendering area is bounded by `rect`, unlike `drawAtPoint`, which has an unlimited width. As a result, this method renders the text in multiple lines.

You should only invoke this method when an `NSView` has focus.

See also:

- 10.2.19 `drawInRect(image as NSImageMBS, x as Double, y as Double, w as Double, h as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)` 753
- 10.2.20 `drawInRect(text as NSAttributedStringMBS, rect as NSRectMBS)` 754

10.2.22 `drawPicture(image as Picture, x as Double, y as Double, w as Double, h as Double, sx as Double, sy as Double, sw as Double, sh as Double, Operation as Integer, fraction as Double)`

Plugin Version: 17.1, Platform: macOS, Targets: Desktop only.

Function: Draws a picture.

Notes: Same as `drawInRect` with `NSImageMBS`, but using `picture`.

10.2.23 `drawRect(x as Double, y as Double, w as Double, h as Double)`

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Draws a rectangle with the current color.

Notes: In the Cocoa world the y axis is reversed. y=0 is on the bottom.

10.2.24 DrawWindowBackground(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Draws the window's default background pattern into the specified rectangle of the currently focused view.

Notes: Pass the rectangle (in the current coordinate system) in which to draw the window's background pattern.

10.2.25 drawWithRect(text as NSAttributedStringMBS, rect as NSRectMBS, options as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Draws the receiver with the specified options, within the given rectangle in the current graphics context.

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

// draw in Canvas
dim g as new NSGraphicsMBS(Canvas1.NSViewMBS)

g.drawWithRect m, new NSRectMBS(20,20, 100, 100), g.NSStringDrawingUsesLineFragmentOrigin
```

Notes: rect: The rectangle specifies the rendering origin in the current graphics context.

options: The string drawing options. See NSStringDrawingOptions for the available options..

The rect argument's origin field specifies the rendering origin. The point is interpreted as the baseline origin by default. With NSStringDrawingUsesLineFragmentOrigin, it is interpreted as the upper left corner of the line fragment rect. The size field specifies the text container size. The width part of the size field

specifies the maximum line fragment width if larger than 0.0. The height defines the maximum size that can be occupied with text if larger than 0.0 and `NSStringDrawingUsesLineFragmentOrigin` is specified. If `NSStringDrawingUsesLineFragmentOrigin` is not specified, height is ignored and considered to be single-line rendering (`NSLineBreakByWordWrapping` and `NSLineBreakByCharWrapping` are treated as `NSLineBreakByClipping`).

You should only invoke this method when there is a current graphics context.
Available in OS X v10.4 and later.

10.2.26 `eraseRect(x as Double, y as Double, w as Double, h as Double)`

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Erases a rectangle with the current color.

Notes: In the Cocoa world the y axis is reversed. `y=0` is on the bottom.

10.2.27 `fill(path as NSBezierPathMBS)`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Paints the region enclosed by the receiver's path.

Example:

```
dim n as new NSImageMBS(300, 300)
```

```
dim g as new NSGraphicsMBS(n)
```

```
g.setFillColor NSColorMBS.redColor
```

```
dim r as NSRectMBS = NSMakeRectMBS(50, 50, 100, 100)
```

```
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPathWithRect(r)
```

```
g.fill(b)
```

```
g = nil
```

```
window1.Backdrop = n.CopyPicture // black image with red color rect
```

Notes: This method fills the path using the current fill color and the receiver's current winding rule. If the path contains any open subpaths, this method implicitly closes them before painting the fill region.

The painted region includes the pixels right up to, but not including, the path line itself. For paths with large line widths, this can result in overlap between the fill region and the stroked path (which is itself centered on the path line).

10.2.28 fillRect(r as NSRectMBS)

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Fills the specified rectangular path with the current fill color.

Example:

```
dim n as new NSImageMBS(300, 300)
dim g as new NSGraphicsMBS(n)

g.setFillColor NSColorMBS.redColor

dim r as new NSRectMBS(10,10,200,200)
g.fillRect(r)

g = nil

window1.Backdrop = n.CopyPicture // black image with red color rectangle
```

Notes: r: A rectangle in the current coordinate system.

This method fills the specified region immediately. This method uses the compositing operation returned by the compositingOperation method of NSGraphicsContext.

See also:

- 10.2.29 fillRect(x as Double, y as Double, w as Double, h as Double) 758
- 10.2.30 fillRect(x as Double, y as Double, w as Double, h as Double, operation as Integer) 759

10.2.29 fillRect(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Fills a rectangle with the current color.

Notes: In the Cocoa world the y axis is reversed. y=0 is on the bottom.

See also:

- 10.2.28 fillRect(r as NSRectMBS) 758
- 10.2.30 fillRect(x as Double, y as Double, w as Double, h as Double, operation as Integer) 759

10.2.30 fillRect(x as Double, y as Double, w as Double, h as Double, operation as Integer)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Fills a rectangle using the current fill color and the specified compositing operation.

Notes: See NSComposite* constants.

See also:

- 10.2.28 fillRect(r as NSRectMBS) 758
- 10.2.29 fillRect(x as Double, y as Double, w as Double, h as Double) 758

10.2.31 flushGraphics

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Forces any buffered operations or data to be sent to the receiver's destination.

Notes: Graphics contexts use buffers to queue pending operations but for efficiency reasons may not always empty those buffers immediately. This method forces the buffers to be emptied.

10.2.32 graphicsContext as NSGraphicsMBS

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

Function: Creates a NSGraphicsMBS object with the current graphics context of the current thread.

Example:

```
Sub Paint(g As Graphics)
// Canvas Paint event in a Mac Cocoa application

dim ng as NSGraphicsMBS = NSGraphicsMBS.graphicsContext
ng.drawAtPoint "Hello World", new NSPointMBS(30, 30)

End Sub
```

Notes: Returns nil on any error.

10.2.33 graphicsContextWithCGContext(targetCGContext as Variant, initialFlipped-State as boolean = false) as NSGraphicsMBS

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

Function: Creates a new graphics context pointing to the given CGContextMBS object.

Example:

```
Sub Paint(g As Graphics)
// get current context
dim cg as CGContextMBS = GetCurrentCGContextMBS

// get graphics context
dim ng as NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithCGContext(cg)

// and draw inside
ng.drawAtPoint "Hello World", new NSPointMBS(30, 30)
End Sub
```

Notes: Returns nil on any error.

10.2.34 graphicsContextWithCGContextHandle(targetCGContextRef as Integer, initialFlippedState as boolean = false) as NSGraphicsMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Creates a new graphics context pointing to the given CGContext reference.

Example:

```
Sub Paint(g As Graphics)
// get current context
dim cg as CGContextMBS = GetCurrentCGContextMBS

// get graphics context
dim ng as NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithCGContextHandle(cg.Handle)

// and draw inside
ng.drawAtPoint "Hello World", new NSPointMBS(30, 30)
End Sub
```

Notes: Returns nil on any error.

See also:

- 10.2.35 graphicsContextWithCGContextHandle(targetCGContextRef as Ptr, initialFlippedState as boolean = false) as NSGraphicsMBS 761

10.2.35 graphicsContextWithCGContextHandle(targetCGContextRef as Ptr, initialFlippedState as boolean = false) as NSGraphicsMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a new graphics context pointing to the given CGContext reference.

Example:

```
Sub Paint(g As Graphics)
// get current context
dim cg as CGContextMBS = GetCurrentCGContextMBS

// get graphics context
dim ng as NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithCGContextHandle(cg.Handle)

// and draw inside
ng.drawAtPoint "Hello World", new NSPointMBS(30, 30)
End Sub
```

Notes: Returns nil on any error.

See also:

- 10.2.34 graphicsContextWithCGContextHandle(targetCGContextRef as Integer, initialFlippedState as boolean = false) as NSGraphicsMBS 760

10.2.36 graphicsContextWithNSBitmapImageRep(targetImage as NSBitmapImageRepMBS) as NSGraphicsMBS

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

Function: Creates a new graphics context for drawing into a bitmap image representation.

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 NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithNSBitmapImageRep(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: Please make sure the graphics object is destroyed (Set to nil) so the drawings flush to the image. Returns nil on any error.

10.2.37 graphicsContextWithNSImage(targetImage as NSImageMBS) as NSGraphicsMBS

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

Function: Creates a new graphics context for drawing into an image.

Example:

```
// make new image
dim myImage as new NSImageMBS(500,500)
dim myGraphics as NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithNSImage(myImage)

// make logo image
dim myPicture as Picture = LogoMBS(500)
dim anotherImage as new NSImageMBS(myPicture)

// draw logo image to new image
myGraphics.drawRect(anotherImage, 0, 0, myImage.width, myImage.height, 0, 0, anotherImage.width,
anotherImage.height, myGraphics.NSCompositeSourceOver, 1.0)
myGraphics = nil // flush drawing

// save to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim b as BinaryStream = BinaryStream.Create(f, True)
if b<>nil then
b.Write myImage.JPEGRepresentation
b.Close
end if
```

Notes: Please make sure the graphics object is destroyed (Set to nil) so the drawings flush to the image. Returns nil on any error.

10.2.38 graphicsContextWithNSView(targetView as NSViewMBS) as NSGraphicsMBS

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

Function: Creates a new graphics context for drawing into a Cocoa view.

Example:

```
// draws in a Cocoa view
dim gg as NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithNSView(canvas1.NSViewMBS)
gg.SetColorRGB 1.0,0,0,0.5
gg.fillRect 0, 0, 100, 100
gg = nil // flush
```

Notes: Please make sure the graphics object is destroyed (Set to nil) so the drawings flush to the window. Returns nil on any error.

10.2.39 graphicsContextWithNSWindow(targetNSWindow as NSWindowMBS) as NSGraphicsMBS

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

Function: Creates a new graphics context for drawing into a window.

Example:

```
// draw in a Cocoa window
dim g as NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithNSWindow(window1.NSWindowMBS)
g.SetColorRGB 1.0,0,0,0.5
g.fillRect 0, 0, 100, 100
g = nil // flush
```

Notes: Returns nil on any error.

10.2.40 graphicsContextWithWindow(targetWindow as DesktopWindow) as NSGraphicsMBS

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

Function: Creates a new graphics context for drawing into a window.

Notes: Returns nil on any error.

See also:

- 10.2.41 graphicsContextWithWindow(targetWindow as window) as NSGraphicsMBS

763

10.2.41 graphicsContextWithWindow(targetWindow as window) as NSGraphicsMBS

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

Function: Creates a new graphics context for drawing into a window.

Example:

```
// draws in a Cocoa window
dim g as NSGraphicsMBS = NSGraphicsMBS.graphicsContextWithWindow(window1)
g.SetColorRGB 1.0,0,0,0.5
g.fillRect 0, 0, 100, 100
g = nil // flush
```

Notes: Returns nil on any error.

See also:

- 10.2.40 graphicsContextWithWindow(targetWindow as DesktopWindow) as NSGraphicsMBS 763

10.2.42 graphicsPort as Variant

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns the low-level, platform-specific graphics context represented by the receiver.

Notes: In Mac OS X, this is the Core Graphics context, a CGContextMBS object.

10.2.43 highlightRect(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Highlights the rectangle.

Notes: In the Cocoa world the y axis is reversed. y=0 is on the bottom.

10.2.44 invalidate

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Invalidates the graphics object.

10.2.45 isDrawingToScreen as boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the drawing destination is the screen.

Notes: True if the drawing destination is the screen, otherwise false.

A return value of false may mean that the drawing destination is a printer, but the destination may also be a PDF or EPS file. If this method returns false, you can call attributes to see if additional information is available about the drawing destination.

10.2.46 isFlipped as boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates the receiver's flipped state.

Notes: True if the receiver is flipped, otherwise false.

The state is determined by sending isFlipped to the receiver's view that has focus. If no view has focus, returns false unless the receiver is instantiated using graphicsContextWithGraphicsPort:flipped: specifying true as the flipped parameter.

Available in Mac OS X v10.4 and later.

10.2.47 restoreGraphicsState

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Removes the receiver's graphics state from the top of the graphics state stack and makes the next graphics state the current graphics state.

Notes: This method must have been preceded with a saveGraphicsState message to add the graphics state to the stack. Invocations of saveGraphicsState and restoreGraphicsState methods may be nested.

Restoring the graphics state restores such attributes as the current drawing style, transformation matrix, color, and font of the original graphics state.

10.2.48 saveGraphicsState

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Saves the graphics state of the current graphics context.

Notes: This method pushes the context onto the per-thread stack.

10.2.49 ScaleCoordinates(x as Double, y as Double)

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Scales coordinate system so the next drawing commands will use different scaling.

Notes: Use `saveGraphicsState` and `restoreGraphicsState` so you can restore the old state.

10.2.50 `set(transform as NSAffineTransformMBS)`

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

Function: Sets the current transform of the graphics environment.

10.2.51 `setClip(path as NSBezierPathMBS)`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Replaces the clipping path of the current graphics context with the area inside the receiver's path.

Notes: You should avoid using this method as a way of adjusting the clipping path, as it may expand the clipping path beyond the bounds set by the enclosing view. If you do use this method, be sure to save the graphics state prior to modifying the clipping path and restore the graphics state when you are done.

This method uses the current winding rule to determine the clipping shape of the receiver. This method does not affect the receiver's path.

10.2.52 `setColor(c as NSColorMBS)`

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

Function: Sets the color of subsequent drawing to the color that the receiver represents.

10.2.53 `SetColorBW(white as Double, alpha as Double = 1.0)`

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Sets the color to a BW color.

Notes: Values range is from 0.0 to 1.0.

Alpha 0.0 is invisible and alpha 1.0 is visible.

10.2.54 SetColorCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Sets the color to a CMYK color.

Notes: Values range is from 0.0 to 1.0.

Alpha 0.0 is invisible and alpha 1.0 is visible.

10.2.55 SetColorHSV(hue as Double, saturation as Double, brightness as Double, alpha as Double = 1.0)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Sets the color to an HSV color.

Notes: Values range is from 0.0 to 1.0.

10.2.56 SetColorRGB(red as Double, green as Double, blue as Double, alpha as Double = 1.0)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Sets the color to a RGB color.

Notes: Values range is from 0.0 to 1.0.

Alpha 0.0 is invisible and alpha 1.0 is visible.

10.2.57 setCurrentContext

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Sets this context to be the current.

10.2.58 setFillColor(c as NSColorMBS)

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

Function: Sets the fill color of subsequent drawing to the receiver's color.

10.2.59 setStrokeColor(c as NSColorMBS)

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

Function: Sets the stroke color of subsequent drawing to the receiver's color.

10.2.60 SetTransform(NSAffineTransform as Variant)

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Sets the current transformation matrix to the receiver's transformation matrix.

Notes: Please use saveGraphicsState so you can restore the state before applying matrix for other drawings.

The current transformation is stored in the current graphics context and is applied to subsequent drawing operations. You should use this method sparingly because it removes the existing transformation matrix, which is an accumulation of transformation matrices for the screen, window, and any superviews. Instead use the concat method to add this transformation matrix to the current transformation matrix.

NSAffineTransform must be a NSAffineTransformMBS object.

10.2.61 sizeWithAttributes(text as string, DicAttributes as dictionary = nil) as NSSizeMBS

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns the bounding box size the text occupies when drawn with the given attributes.

Example:

```
Dim NSGraphics as New NSGraphicsMBS()
Dim NSStringWidth as Double = NSGraphics.sizeWithAttributes("Hello World").Width
```

```
MsgBox("StringWidth from NSGraphicsMBS: " + Str(NSStringWidth))
```

```
Dim REALGraphics as Graphics = window1.Graphics
Dim REALStringWidth as Double = REALGraphics.StringWidth("Hello World")
```

```
MsgBox("StringWidth from REAL Graphics: " + Str(REALStringWidth))
```

Notes: attributes: A dictionary of text attributes to be applied to the string. These are the same attributes that can be applied to an NSAttributedString object, but in the case of strings, the attributes apply to the entire string, rather than ranges within the string.

Returns the bounding box size the receiver occupies when drawn with attributes.

10.2.62 stroke(path as NSBezierPathMBS)

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Draws a line along the receiver's path using the current stroke color and drawing attributes.

Example:

```
dim n as new NSImageMBS(300, 300)
dim g as new NSGraphicsMBS(n)

g.setStrokeColor NSColorMBS.redColor

dim r as NSRectMBS = NSMakeRectMBS(50, 50, 100, 100)
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPathWithRect(r)
b.lineWidth = 5
g.stroke(b)

g = nil

window1.Backdrop = n.CopyPicture // black image with red color rect
```

Notes: The drawn line is centered on the path with its sides parallel to the path segment. This method uses the current drawing attributes associated with the receiver. If a particular attribute is not set for the receiver, this method uses the corresponding default attribute.

10.2.63 strokeLine(point1 as NSPointMBS, point2 as NSPointMBS)

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Strokes a line between two points using the current stroke color and the default drawing attributes.

Example:

```
dim n as new NSImageMBS(300, 300)
dim g as new NSGraphicsMBS(n)

g.setStrokeColor NSColorMBS.redColor

dim p1 as new NSPointMBS(10,10)
dim p2 as new NSPointMBS(50,50)
g.strokeLine(p1,p2)
```

```
g = nil
```

```
window1.Backdrop = n.CopyPicture // black image with red color line
```

Notes: point1: The starting point of the line.
point2: The ending point of the line.

This method strokes the specified path immediately.
See also:

- 10.2.64 `strokeLine(x1 as Double, y1 as Double, x2 as Double, y2 as Double)` 770

10.2.64 `strokeLine(x1 as Double, y1 as Double, x2 as Double, y2 as Double)`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Strokes a line.

See also:

- 10.2.63 `strokeLine(point1 as NSPointMBS, point2 as NSPointMBS)` 769

10.2.65 `strokeRect(r as NSRectMBS)`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Strokes the path of the specified rectangle using the current stroke color and the default drawing attributes.

Example:

```
dim n as new NSImageMBS(300, 300)
```

```
dim g as new NSGraphicsMBS(n)
```

```
g.setStrokeColor NSColorMBS.redColor
```

```
dim r as NSRectMBS = NSMakeRectMBS(50, 50, 100, 100)
```

```
g.strokeRect(r)
```

```
g = nil
```

```
window1.Backdrop = n.CopyPicture // black image with red color rect
```

Notes: r: A rectangle in the current coordinate system.

The path is drawn beginning at the rectangle's origin and proceeding in a counterclockwise direction. This method strokes the specified path immediately.

10.2.66 TranslateCoordinates(x as Double, y as Double)

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Moves coordinate system so the next drawing commands will use different starting point.

Notes: Use saveGraphicsState and restoreGraphicsState so you can restore the old state.

10.2.67 Properties

10.2.68 Handle as Integer

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The internal handle to the Graphics Context.

Notes: Reference to NSGraphicsContext object.

(Read and Write property)

10.2.69 Owner as Variant

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

Function: The owner object.

Notes: When you have a graphics object based on a window, view or image, this property points to the original object to keep it alive while drawing.

(Read and Write property)

10.2.70 Valid as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Whether this graphics object is still valid.

Notes: Set to true when the plugin creates an object and false when the object is no longer needed.

(Read and Write property)

10.2.71 `imageInterpolation` as Integer

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The interpolation behavior.

Notes: Note that this value is not part of the graphics state, so it cannot be reset using `restoreGraphicsState`.

(Read and Write computed property)

10.2.72 `shouldAntialias` as boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Whether the receiver should use antialiasing.

Notes: This value is part of the graphics state and is restored by `restoreGraphicsState`.

(Read and Write computed property)

10.2.73 Constants

Constants

Constant	Value	Description
NSCompositeClear	0	Transparent. ($R = 0$)
NSCompositeCopy	1	Source image. ($R = S$)
NSCompositeDestinationAtop	9	Destination image wherever both images are opaque, source image wherever source image is opaque but destination image is transparent, and transparent elsewhere. ($R = S*(1 - Da) + D*Sa$)
NSCompositeDestinationIn	7	Destination image wherever both images are opaque, and transparent elsewhere. ($R = D*Sa$)
NSCompositeDestinationOut	8	Destination image wherever destination image is opaque but source image is transparent, and transparent elsewhere. ($R = D*(1 - Sa)$)
NSCompositeDestinationOver	6	Destination image wherever destination image is opaque, and source image elsewhere. ($R = S*(1 - Da) + D$)
NSCompositeHighlight	12	Source image wherever source image is opaque, and destination image elsewhere. (Deprecated. Mapped to NSCompositeSourceOver.)
NSCompositePlusDarker	11	Sum of source and destination images, with color values approaching 0 as a limit. ($R = \text{MAX}(0, (1 - D) + (1 - S))$)
NSCompositePlusLighter	13	Sum of source and destination images, with color values approaching 1 as a limit. ($R = \text{MIN}(1, S + D)$)
NSCompositeSourceAtop	5	Source image wherever both images are opaque, destination image wherever destination image is opaque but source image is transparent, and transparent elsewhere. ($R = S*Da + D*(1 - Sa)$)
NSCompositeSourceIn	3	Source image wherever both images are opaque, and transparent elsewhere. ($R = S*Da$)
NSCompositeSourceOut	4	Source image wherever source image is opaque but destination image is transparent, and transparent elsewhere. ($R = S*(1 - Da)$)
NSCompositeSourceOver	2	Source image wherever source image is opaque, and destination image elsewhere. ($R = S + D*(1 - Sa)$)
NSCompositeXOR	10	Exclusive OR of source and destination images. ($R = S*(1 - Da) + D*(1 - Sa)$) Works only with black and white images and is not recommended for color contexts.
NSImageInterpolationDefault	0	One of the interpolation constants. Use the context's default interpolation.
NSImageInterpolationHigh	3	One of the interpolation constants. Slower, higher-quality interpolation.
NSImageInterpolationLow	2	One of the interpolation constants. Fast, low-quality interpolation.
NSImageInterpolationMedium	4	One of the interpolation constants. Medium quality, slower than NSImageInterpolationLow. Available in Mac OS X v10.6 and later.
NSImageInterpolationNone	1	One of the interpolation constants. No interpolation.

Drawing Option Constants

Constant	Value	Description
<code>NSStringDrawingDisableScreenFontSubstitution</code>	4	Disable screen font substitution (equivalent to <code>NSLayoutManagerFonts(false)</code>).
<code>NSStringDrawingOneShot</code>	16	Suppresses caching layout information.
<code>NSStringDrawingTruncatesLastVisibleLine</code>	32	Truncates and adds the ellipsis character to the last visible line if it doesn't fit into the bounds specified. This option is ignored if <code>NSStringDrawingUsesLineFragmentOrigin</code> is set. In addition, the line break mode must be either <code>NSLineBreakByWrapping</code> or <code>NSLineBreakByCharWrapping</code> for this option to have any effect. The line break mode can be specified in a paragraph style passed in the <code>paragraphStyle</code> dictionary argument of the drawing methods. Available in Mac OS X v10.5 and later.
<code>NSStringDrawingUsesDeviceMetrics</code>	8	Uses image glyph bounds instead of typographic bounds.
<code>NSStringDrawingUsesFontLeading</code>	2	Uses the font leading for calculating line heights.
<code>NSStringDrawingUsesLineFragmentOrigin</code>	1	The specified origin is the line fragment origin, not the baseline.

Chapter 11

Cocoa Networking

11.1 class `NSURLRequestCertificateFilterMBS`

11.1.1 class `NSURLRequestCertificateFilterMBS`

Plugin Version: 7.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class to filter certificate requests.

Notes: This is the only way to fix the problem with the webview that certificates which are not valid can still be used.

11.1.2 Events

11.1.3 `allowsAnyHTTPSCertificateForHost(host as string)` as boolean

Plugin Version: 7.5, Platform: macOS, Targets: .

Function: An event being called for each host which may have a https certificate.

Notes: This event is called very often, so make it very fast.

Also this event is often called with the same host value as it is called for each request.

Retrun true to allow this host to run without a valid https certificate.

Chapter 12

Cocoa Printing

12.1 class NSPageLayoutMBS

12.1.1 class NSPageLayoutMBS

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

Function: NSPageLayout is a panel that queries the user for information such as paper type and orientation.

Notes: It is normally displayed in response to the user selecting the Page Setup menu item. You obtain an instance with the `pageLayout` class method. The pane can then be run as a sheet using `beginSheetWithPrintInfo` or modally using `runModal` or `runModalWithPrintInfo`.

Blog Entries

- [MBS Real Studio Plugins, version 12.4pr1](#)

12.1.2 Methods

12.1.3 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow)`

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

Function: Presents a page setup sheet for the given `NSPrintInfo` object, document-modal relative to the given window.

Notes: `printInfo`: The `NSPrintInfo` object to use.

`win`: The window to which the sheet is attached.

This method calls the `printPanelDidEnd` event later passing `returnCode` which is either `NSCancelButton` (0) or `NSOKButton` (1).

See also:

- 12.1.4 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS)` 778
- 12.1.5 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window)` 778

12.1.4 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Presents a page setup sheet for the given `NSPrintInfo` object, document-modal relative to the given window.

Notes: `printInfo`: The `NSPrintInfo` object to use.

`win`: The window to which the sheet is attached.

This method calls the `printPanelDidEnd` event later passing `returnCode` which is either `NSCancelButton` (0) or `NSOKButton` (1).

See also:

- 12.1.3 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow)` 777
- 12.1.5 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window)` 778

12.1.5 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Presents a page setup sheet for the given `NSPrintInfo` object, document-modal relative to the given window.

Notes: `printInfo`: The `NSPrintInfo` object to use.

`win`: The window to which the sheet is attached.

This method calls the `printPanelDidEnd` event later passing `returnCode` which is either `NSCancelButton` (0) or `NSOKButton` (1).

See also:

- 12.1.3 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow)` 777
- 12.1.4 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS)` 778

12.1.6 Constructor

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

Function: Initializes a new page layout object.

12.1.7 `pageLayout` as `NSPageLayoutMBS`

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

Function: Returns a newly created `NSPageLayout` object.

12.1.8 `printInfo` as `NSPrintInfoMBS`

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

Function: Returns the `NSPrintInfo` object used when the receiver is run.

Notes: The `NSPrintInfo` object is set using the `beginSheetWithPrintInfo` or `runModalWithPrintInfo` method. The shared `NSPrintInfo` object is used if the receiver is run using `runModal`.

12.1.9 `runModal` as `Integer`

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

Function: Displays the receiver and begins the modal loop using the shared `NSPrintInfo` object.

Example:

```
dim p as new NSPageLayoutMBS
MsgBox str(p.runModal)
```

Notes: Returns `NSCancelButton` (0) if the user clicks the Cancel button; otherwise, `NSOKButton` (1). The receiver's values are recorded in the shared `NSPrintInfo` object.

12.1.10 `runModalWithPrintInfo(printInfo as NSPrintInfoMBS)` as `Integer`

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

Function: Displays the receiver and begins the modal loop using the given `NSPrintInfo` object.

Notes: `printInfo`: The `NSPrintInfo` object to use.

Returns `NSCancelButton` if the user clicks the Cancel button; otherwise, `NSOKButton`. The receiver's values are recorded in `printInfo`.

12.1.11 runPageLayout

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

Function: Displays the app's page layout panel, an instance of NSPageLayout.

Example:

```
NSPageLayoutMBS.runPageLayout
```

Notes: If the NSPageLayout instance does not exist, this method creates one. This method is typically invoked when the user chooses Page Setup from the application's File menu.

12.1.12 Properties

12.1.13 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

12.1.14 Events

12.1.15 printPanelDidEnd(returnCode as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: Called when the sheet is dismissed.

12.2 class NSPrinterMBS

12.2.1 class NSPrinterMBS

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

Function: An NSPrinter object describes a printer's capabilities as defined in its PPD file.

Example:

```
dim p as NSPrinterMBS = NSPrinterMBS.defaultPrinter
MsgBox p.name+EndOfLine+p.type
```

Notes: An NSPrinter object can be constructed by specifying either the printer name or the make and model of an available printer. You use a printer object to get information about printers, not to modify printer attributes or control a printing job.

Blog Entries

- [MBS Real Studio Plugins, version 12.4pr1](#)

12.2.2 Methods

12.2.3 booleanForKey(key as string, table as string) as boolean

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

Function: Returns the Boolean value associated with the specified key.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns the Boolean value associated with the key. Returns false if the key is not in the table or the receiver lacks a PPD file.

12.2.4 Constructor(name as string = "")

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

Function: Initializes a printer object.

Notes: On success the handle property is not zero.

Name can be empty to pick default printer. Else pass name of printer.

12.2.5 copy as NSPrinterMBS

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

Function: Creates a clone of the printer object.

12.2.6 defaultPrinter as NSPrinterMBS

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

Function: Returns the default printer.

Example:

```
dim p as NSPrinterMBS = NSPrinterMBS.defaultPrinter
MsgBox p.name
```

12.2.7 deviceDescription as Dictionary

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

Function: Returns a dictionary of keys and values describing the device.

Notes: A dictionary of the device properties. See NSGraphics.h for possible keys. The only key guaranteed to exist is NSDeviceIsPrinter.

12.2.8 floatForKey(key as string, table as string) as Double

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

Function: Returns the floating-point value associated with the specified key.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns the floating-point value. Returns 0.0 if the key is not in the table or the receiver lacks a PPD file.

12.2.9 intForKey(key as string, table as string) as Integer

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

Function: Returns the integer value associated with the specified key.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns the integer value. Returns 0 if the key is not in the table or the receiver lacks a PPD file.

12.2.10 isKey(key as string, table as string) as boolean

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

Function: Returns a Boolean value indicating whether the specified key is in the specified table.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns true if the key is in the table; otherwise, false.

12.2.11 languageLevel as Integer

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

Function: Returns the PostScript language level recognized by the printer.

Example:

```
dim p as NSPrinterMBS = NSPrinterMBS.defaultPrinter
MsgBox "languageLevel: "+str(p.languageLevel)
```

Notes: Returns the PostScript language level. The value is 0 if the receiver is not a PostScript printer.

12.2.12 name as string

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

Function: Returns the printer's name.

Example:

```
dim p as NSPrinterMBS = NSPrinterMBS.defaultPrinter
MsgBox p.name
```

12.2.13 pageSizeForPaper(paperName as string) as NSSizeMBS

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

Function: Returns the size of the page for the specified paper type.

Notes: `paperName`: Possible values are printer-dependent and are contained in the printer's PPD file. Typical values are "Letter" and "Legal".

Returns the size of the page, measured in points in the user coordinate space. The returned size is zero if the specified paper name is not recognized or its entry in the PPD file cannot be parsed.

12.2.14 `printerNames as string()`

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

Function: Returns the names of all available printers.

Example:

```
MsgBox Join(NSPrinterMBS.printerNames, EndOfLine)
```

Notes: An array of strings, each of which contains the name of an available printer. The user constructs the list of available printers using the Print Center application.

12.2.15 `printerTypes as string()`

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

Function: Returns descriptions of the makes and models of all available printers.

Notes: An array of strings, each of which contains the make and model information for a supported printer.

12.2.16 `printerWithName(name as string) as NSPrinterMBS`

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

Function: Creates and returns an `NSPrinter` object initialized with the specified printer name.

Notes: `name`: The name of the printer.

Returns an initialized `NSPrinter` object, or `nil` if the specified printer was not available.

12.2.17 `printerWithType(type as string) as NSPrinterMBS`

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

Function: Creates and returns an NSPrinter object initialized to the first available printer with the specified make and model information.

Notes: type: A string describing the make and model information. You can get this string using the printerTypes method.

Returns an initialized NSPrinter object, or nil if the specified printer was not available.

12.2.18 rectForKey(key as string, table as string) as NSRectMBS

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

Function: Returns the rectangle associated with the specified key.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns the rectangle value. Returns NSRectMBS.Zero if the key is not in the table or the receiver lacks a PPD file.

12.2.19 sizeForKey(key as string, table as string) as NSSizeMBS

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

Function: Returns the size data type associated with the specified key.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns the size value. Returns NSZeroSize if the key is not in the table or the receiver lacks a PPD file.

12.2.20 statusForTable(paperName as string) as Integer

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

Function: Returns the status of the specified table.

Notes: table: The name of a table from the printer's PPD file.

Returns one of the return values described in Constants.

12.2.21 stringForKey(key as string, table as string) as string

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

Function: Returns the first occurrence of a value associated with specified key.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns the value for the specified key, or nil if the key is not in the table. The returned string may also be empty.

If key is a main keyword only, and if that keyword has options in the PPD file, this method returns an empty string. Use `stringListForKey` to retrieve the values for all occurrences of a main keyword.

12.2.22 stringListForKey(key as string, table as string) as string()

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

Function: Returns an array of strings, one for each occurrence, associated with specified key.

Notes: key: The key whose value you want.

table: The name of a table from the printer's PPD file.

Returns an array of strings, each containing a value associated with the specified key. Returns nil if the key is not in the table.

12.2.23 type as string

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

Function: Returns a description of the printer's make and model.

Example:

```
dim p as NSPrinterMBS = NSPrinterMBS.defaultPrinter
MsgBox p.type
```

12.2.24 Properties

12.2.25 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

12.2.26 Constants

Printer Information Table States

Constant	Value	Description
NSPrinterTableError	2	Printer table is not valid.
NSPrinterTableNotFound	1	Printer table was not found.
NSPrinterTableOK	0	Printer table was found and is valid.

12.3 class NSPrintInfoMBS

12.3.1 class NSPrintInfoMBS

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

Function: An NSPrintInfo object stores information that's used to generate printed output.

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: A shared NSPrintInfo object is automatically created for an application and is used by default for all printing jobs for that application.

The printing information in an NSPrintInfo object is stored in a dictionary. To access the standard attributes in the dictionary directly, this class defines a set of keys and provides the dictionary method. You can also initialize an instance of this class using the Constructor method.

You can use this dictionary to store custom information associated with a print job. Any non-object values should be stored as NSNumber or NSValue objects in the dictionary. See NSNumber Class Reference for a list of types which should be stored as numbers. For other non-object values, use the NSValue class.

Beginning with OS X v10.5, to store custom information that belongs in printing presets you should use the dictionary returned by the printSettings method.

Blog Entries

- [Several ways for picture to PDF in MBS Plugins](#)
- [Tip fo the day: Print to PDF for WebView](#)

- [Tip of day: Save Xojo report to PDF](#)
- [Tip of the day: Print to PDF with OpenPrinter in Xojo](#)
- [\[ANN \] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.3](#)
- [PDF printing via PDFKit](#)
- [MBS Xojo / Real Studio Plugins, version 14.3pr10](#)
- [MBS Xojo / Real Studio Plugins, version 14.1pr3](#)
- [Print to PDF File on Mac OS X with Xojo](#)
- [MBS Real Studio Plugins, version 12.4pr1](#)

12.3.2 Methods

12.3.3 Constructor

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

Function: Initializes the print info with a new instance.

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
```

See also:

- [12.3.4 Constructor\(attributes as Dictionary\)](#) 790
- [12.3.5 Constructor\(Data as Memoryblock\)](#) 790

12.3.4 Constructor(attributes as Dictionary)

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

Function: Initialize the print info with the parameters in the specified dictionary.

See also:

- 12.3.3 Constructor 789
- 12.3.5 Constructor(Data as Memoryblock) 790

12.3.5 Constructor(Data as Memoryblock)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initialize the print info with serialized dictionary or NSPrintInfo.

Example:

```
dim p as new PrinterSetup
dim info as new NSPrintInfoMBS(p.SetupString)
MsgBox info.paperName
```

See also:

- 12.3.3 Constructor 789
- 12.3.4 Constructor(attributes as Dictionary) 790

12.3.6 copy as NSPrintInfoMBS

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

Function: Creates a copy of the object.

12.3.7 defaultPrinter as NSPrinterMBS

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

Function: Returns the default printer.

12.3.8 NSPrintAllPages as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A boolean.

12.3.9 NSPrintBottomMargin as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a floating-point value that specifies the bottom margin, in points.

12.3.10 NSPrintCancelJob as string

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

Function: One of the possible job disposition values.

Notes: Cancel print job.

12.3.11 NSPrintCopies as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: a number containing the number of copies of the print job to be printed

12.3.12 NSPrintDetailedErrorReporting as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A boolean.

12.3.13 NSPrintFaxNumber as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A string containing a fax number.

12.3.14 `NSPrintFirstPage` as string

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

Function: One of the keys for print job attributes that are recognized by `NSPrintInfo`.

Notes: A number containing the one-based index of the first job in the page to print.

12.3.15 `NSPrintHeaderAndFooter` as string

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

Function: One of the keys for print job attributes that are recognized by `NSPrintInfo`.

Notes: A boolean for whether the results of `NSView pageHeader` and `NSView pageFooter` should be drawn on pages

12.3.16 `NSPrintHorizontallyCentered` as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a Boolean value that is true if pages are centered horizontally.

12.3.17 `NSPrintHorizontalPagination` as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a `NSPrintingPaginationMode` value.

`NSAutoPagination`, `NSFitPagination`, or `NSClipPagination`. See `HorizontalPagination` for details.

12.3.18 `NSPrintJobDisposition` as string

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

Function: One of the keys for print job attributes that are recognized by `NSPrintInfo`.

Notes: A string equal to `NSPrintSpoolJob`, `NSPrintPreviewJob`, `NSPrintSaveJob`, or `NSPrintCancelJob`.

12.3.19 `NSPrintJobSavingFileNameExtensionHidden` as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A boolean for whether the job file's name extension should be hidden, for NSPrintSaveJob. Available on Mac OS X 10.6.

12.3.20 NSPrintJobSavingURL as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: An URL containing the location to which the job file will be saved, for NSPrintSaveJob. Available on Mac OS X 10.6.

12.3.21 NSPrintLastPage as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: An number containing the one-based index of the last job in the page to print.

12.3.22 NSPrintLeftMargin as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a floating-point value that specifies the left margin, in points.

12.3.23 NSPrintMustCollate as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A boolean value.

12.3.24 NSPrintOrientation as string

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

Function: One of the dictionary keys to access page format attributes.

Notes: A number containing an NSPrintingOrientation.

NSPortraitOrientation or NSLandscapeOrientation

12.3.25 NSPrintPagesAcross as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A number containing the number of logical pages to be placed across a physical sheet.

12.3.26 NSPrintPagesDown as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A number containing the number of logical pages to be placed down a physical sheet.

12.3.27 NSPrintPaperName as string

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

Function: One of the dictionary keys to access page format attributes.

Notes: A string containing the paper name.

12.3.28 NSPrintPaperSize as string

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

Function: One of the dictionary keys to access page format attributes.

Notes: A size value specifying the height and width of paper in points.

12.3.29 NSPrintPreviewJob as string

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

Function: One of the possible job disposition values.

Notes: Send to Preview application.

12.3.30 NSPrintPrinter as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: Value in dictionary is a NSPrinterMBS.

12.3.31 NSPrintPrinterName as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: A string containing the name of a printer.

12.3.32 NSPrintReversePageOrder as string

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

Function: One of the keys for print job attributes that are recognized by NSPrintInfo.

Notes: Value for this key is a boolean value.

12.3.33 NSPrintRightMargin as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a floating-point value that specifies the right margin, in points.

12.3.34 NSPrintSaveJob as string

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

Function: One of the possible job disposition values.

Notes: Save to a file.

12.3.35 NSPrintScalingFactor as string

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

Function: One of the dictionary keys to access page format attributes.

Notes: Scale factor percentage before pagination.

12.3.36 `NSPrintSelectionOnly` as string

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

Function: One of the keys for print job attributes that are recognized by `NSPrintInfo`.

Notes: A boolean value.

Available on Mac OS X 10.6 or newer.

12.3.37 `NSPrintSpoolJob` as string

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

Function: One of the possible job disposition values.

Notes: Normal print job.

12.3.38 `NSPrintTime` as string

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

Function: One of the keys for print job attributes that are recognized by `NSPrintInfo`.

Notes: An `NSDate` containing the time at which printing should begin.

12.3.39 `NSPrintTopMargin` as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a floating-point value that specifies the top margin, in points.

12.3.40 `NSPrintVerticallyCentered` as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a Boolean value that is true if pages are centered vertically.

12.3.41 NSPrintVerticalPagination as string

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

Function: One of the dictionary keys to access pagination attributes.

Notes: Number, containing a NSPrintingPaginationMode value.

NSAutoPagination, NSFitPagination, or NSClipPagination. See VerticalPagination for details.

12.3.42 SetSaveDestination(file as folderitem)

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

Function: Sets the print job to go to a PDF file.

Example:

```
// print to PDF in Xojo Cocoa app

// change print info to go to
dim s as NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
dim d as MemoryBlock = s.data // save old
s.SetSaveDestination SpecialFolder.Desktop.Child("test.pdf")

// now print something
dim g as Graphics = OpenPrinter
if g<>Nil then
g.DrawString "Hello World PDF", 20, 20
end if

s.data = d // restore original settings

// now print something to regular printer
g = OpenPrinter
if g<>Nil then
g.DrawString "Hello World Printer", 20, 20
end if
```

12.3.43 setSharedPrintInfo(printInfo as NSPrintInfoMBS)

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

Function: Sets the shared NSPrintInfo object to the specified object.

Notes: printInfo: The new shared printer information. This value must not be nil.

The shared `NSPrintInfo` object defines the settings for the `NSPageLayout` panel and print operations that will be used if no `NSPrintInfo` object is specified for those operations.

12.3.44 `setUpPrintOperationDefaultValues`

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

Function: Validates the attributes encapsulated by the receiver.

Notes: Invoked when the print operation is about to start. Subclasses may override this method to set default values for any attributes that are not set.

12.3.45 `sharedPrintInfo` as `NSPrintInfoMBS`

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

Function: Returns the shared `NSPrintInfo` object.

12.3.46 Properties

12.3.47 `bottomMargin` as `Double`

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

Function: The bottom margin, measured in points in the user coordinate space.

Notes: (Read and Write property)

12.3.48 `data` as `Memoryblock`

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

Function: Query or set the current settings as data.

Example:

```
// print to PDF in Xojo Cocoa app

// change print info to go to
dim s as NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
dim d as MemoryBlock = s.data // save old
s.SetSaveDestination SpecialFolder.Desktop.Child("test.pdf")

// now print something
```

```

dim g as Graphics = OpenPrinter
if g<>Nil then
g.DrawString "Hello World PDF", 20, 20
end if

```

```

s.data = d // restore original settings

// now print something to regular printer
g = OpenPrinter
if g<>Nil then
g.DrawString "Hello World Printer", 20, 20
end if

```

Notes: The plugin archives the current settings and you can later assign them back. (Read and Write property)

12.3.49 dictionary as dictionary

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

Function: Returns the receiver's dictionary that contains the printing attributes.

Notes: The key-value pairs contained in the dictionary are described in Constants. Modifying the returned dictionary changes the receiver's attributes.

This dictionary is key-value observing compliant. (Read and Write property)

12.3.50 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

12.3.51 HorizontallyCentered as boolean

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

Function: Whether the image is centered horizontally.

Notes: True if you want the image to be centered horizontally; otherwise, false.

(Read and Write property)

12.3.52 horizontalPagination as Integer

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

Function: The horizontal pagination to the specified mode.

Notes: One of the pagination modes described in constants.

(Read and Write property)

12.3.53 imageablePageBounds as NSRectMBS

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

Function: Returns the imageable area of a sheet of paper specified by the receiver.

Notes: Return the imageable area, measured in points in the user coordinate space.

This method takes into account the current printer, paper size, and orientation settings, but not scaling factors. Imageable area is the maximum area that can possibly be marked on by the printer hardware, not the area defined by the current margin settings.

The origin (0, 0) of the returned rectangle is in the lower-left corner of the oriented sheet. The imageable bounds may extend past the edges of the sheet when, for example, a printer driver specifies it so that borderless printing can be done reliably.

(Read only property)

12.3.54 jobDisposition as string

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

Function: The action specified for the job.

Notes: One of the following value:

- `NSPrintSpoolJob` is a normal print job.
- `NSPrintPreviewJob` sends the print job to the Preview application.
- `NSPrintSaveJob` saves the print job to a file.
- `NSPrintCancelJob` aborts the print job.

(Read and Write property)

12.3.55 leftMargin as Double

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

Function: The left margin to the specified size.

Notes: The size for the left margin, measured in points in the user coordinate space.

(Read and Write property)

12.3.56 localizedPaperName as string

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

Function: Returns the human-readable name of the currently selected paper size, suitable for presentation in user interfaces.

Notes: This is typically different from the name returned by `paperName`, which is almost never suitable for presentation to the user.

(Read only property)

12.3.57 orientation as Integer

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

Function: The page orientation to the specified value.

Notes: This printing orientation. See constants for possible values.

For consistency, this method may change either the paper name or the paper size.

(Read and Write property)

12.3.58 paperName as string

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

Function: The paper name to the specified value.

Notes: The name for the paper size. The string contains a value such as Letter or Legal. Paper names are implementation specific.

For consistency, this method may change either the paper size or the page orientation.

(Read and Write property)

12.3.59 `paperSize` as `NSSizeMBS`

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

Function: The width and height of the paper to the specified size.

Notes: The new size of the paper, measured in points in the user coordinate space.

For consistency, this method may change either the paper name or the page orientation.

(Read and Write property)

12.3.60 `printer` as `NSPrinterMBS`

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

Function: The printer object used for subsequent printing jobs.

Notes: This method iterates through the receiver's dictionary. If a feature in the dictionary is not supported by the new printer (as determined by a query to the PPD file), that feature is removed from the dictionary.

(Read and Write property)

12.3.61 `printerName` as `String`

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The printer name of printer used for subsequent printing jobs.

Notes: (Read only property)

12.3.62 `printSettings` as `dictionary`

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

Function: Returns a mutable dictionary containing the print settings from Core Printing.

Notes: A mutable dictionary containing the printing system's current settings.

You can use this method to get and set values from the system print settings. The keys in the returned dictionary represent the values returned by the Core Printing function `PMPrintSettingsGetValue`. They correspond to the settings currently in the print panel and include everything from custom values set by your accessory panels to values provided by the printer driver's print dialog extension.

(Read only property)

12.3.63 `rightMargin` as `Double`

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

Function: The right margin to the specified size.

Notes: The size for the right margin, measured in points in the user coordinate space.
(Read and Write property)

12.3.64 scalingFactor as Double

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

Function: The print info's scaling factor.

Notes: Default is 1.0.

Using smaller value increases paper size.

(Read and Write property)

12.3.65 SelectionOnly as boolean

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

Function: Whether only the current selection should be printed.

Notes: True if only the current selection should be printed, otherwise false.

(Read and Write property)

12.3.66 SetupString as Memoryblock

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Query or set the current settings as data.

Example:

```
// start with a printer setup
dim p as new PrinterSetup

// clone to NSPrintInfo
dim info as new NSPrintInfoMBS(p.SetupString)

// find out what name second printer has
dim printers() as string = NSPrinterMBS.printerNames
dim printer as NSPrinterMBS = NSPrinterMBS.printerWithName(printers(1))
System.DebugLog printers(1)

// now set a new paper size and this printer
info.paperSize = new NSSizeMBS(72*5, 72*6) // 5 by 6 inch
info.printer = printer
```

```
// and clone back
p.SetupString = info.SetupString

// now print to this printer with this paper
dim g as Graphics = OpenPrinter(p)

g.DrawString "Hello", 10, 10
```

Notes: While data property encodes the dictionary, this property encodes the `NSPrintInfo` which is same format as `PrinterSetup.SetupString` in Xojo.
(Read and Write property)

12.3.67 topMargin as Double

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

Function: The top margin, measured in points in the user coordinate space.

Notes: The size for the top margin, measured in points in the user coordinate space.
(Read and Write property)

12.3.68 VerticallyCentered as boolean

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

Function: Whether the image is centered vertically.

Notes: True if you want the image to be centered vertically; otherwise, false.
(Read and Write property)

12.3.69 verticalPagination as Integer

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

Function: The vertical pagination mode.

Notes: One of the pagination modes described in constants.
(Read and Write property)

12.3.70 Constants

Pagination Modes

Constant	Value	Description
NSAutoPagination	0	The image is divided into equal-sized rectangles and placed in one column of pages.
NSClipPagination	2	The image is clipped to produce one column or row of pages.
NSFitPagination	1	The image is scaled to produce one column or one row of pages.

Page Orientation Constants

Constant	Value	Description
NSLandscapeOrientation	1	Orientation is portrait (page is taller than it is wide).
NSPortraitOrientation	0	Orientation is landscape (page is wider than it is tall).

12.4 class NSPrintOperationMBS

12.4.1 class NSPrintOperationMBS

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

Function: An NSPrintOperation object controls operations that generate Encapsulated PostScript (EPS) code, Portable Document Format (PDF) code, or print jobs.

Notes: An NSPrintOperation object works in conjunction with two other objects: an NSPrintInfo object, which specifies how the code should be generated, and an NSView object, which generates the actual code.

It is important to note that the majority of methods in NSPrintOperation copy the instance of NSPrintInfo passed into them. Future changes to that print info are not reflected in the print info retained by the current NSPrintOperation object. All changes should be made to the print info before passing to the methods of this class.

Blog Entries

- [MBS Xojo Plugins, version 21.6pr3](#)
- [Several ways for picture to PDF in MBS Plugins](#)
- [MBS Xojo Plugins, version 21.2pr5](#)
- [MBS Xojo Plugins, version 20.6pr3](#)
- [Tip fo the day: Print to PDF for WebView](#)
- [MBS Xojo Plugins, version 17.6pr5](#)
- [\[ANN \] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.4](#)
- [MBS Xojo / Real Studio Plugins, version 14.4pr10](#)
- [PDF printing via PDFKit](#)
- [MBS Real Studio Plugins, version 12.4pr1](#)

12.4.2 Methods

12.4.3 Constructor

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

Function: Initialize the object the current print operation for this thread.

Notes: On success the handle property is not zero.

See also:

- 12.4.4 Constructor(other as NSPrintOperationMBS)

12.4. CLASS NSPRINTOPERATIONMBS	807
• 12.4.5 Constructor(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil)	807
• 12.4.6 Constructor(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil)	808
• 12.4.7 Constructor(view as NSViewMBS)	809
• 12.4.8 Constructor(view as NSViewMBS, printInfo as NSPrintInfoMBS)	809

12.4.4 Constructor(other as NSPrintOperationMBS)

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

Function: Special constructor to create new NSPrintOperationMBS for existing NSPrintOperationMBS object.

Notes: If you have a NSPrintOperationMBS and you want to use printOperationDidRun event, you can initialize a subclass of NSPrintOperationMBS with your existing object to get the event there.

See also:

• 12.4.3 Constructor	806
• 12.4.5 Constructor(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil)	807
• 12.4.6 Constructor(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil)	808
• 12.4.7 Constructor(view as NSViewMBS)	809
• 12.4.8 Constructor(view as NSViewMBS, printInfo as NSPrintInfoMBS)	809

12.4.5 Constructor(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil)

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

Function: Creates and returns an NSPrintOperation object ready to control the printing of the specified view using custom print settings.

Example:

```
// print a HTMLViewer
dim n as new NSPrintOperationMBS(HTMLViewer1)

n.showsPrintPanel = true
n.showsProgressPanel = true
n.runOperationModalForWindow(self)
```

Notes: View: The view whose contents you want to print.
PrintInfo: The print settings to use when printing the view.

Returns the new `NSPrintOperation` object. You must run the operation to print the view.

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.3 Constructor 806
- 12.4.4 Constructor(other as `NSPrintOperationMBS`) 807
- 12.4.6 Constructor(view as `HTMLViewer`, printInfo as `NSPrintInfoMBS = nil`) 808
- 12.4.7 Constructor(view as `NSViewMBS`) 809
- 12.4.8 Constructor(view as `NSViewMBS`, printInfo as `NSPrintInfoMBS`) 809

12.4.6 Constructor(view as `HTMLViewer`, printInfo as `NSPrintInfoMBS = nil`)

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an `NSPrintOperation` object ready to control the printing of the specified view using custom print settings.

Example:

```
// print a HTMLViewer
dim n as new NSPrintOperationMBS(HTMLViewer1)

n.showsPrintPanel = true
n.showsProgressPanel = true
n.runOperationModalForWindow(self)
```

Notes: View: The view whose contents you want to print.

PrintInfo: The print settings to use when printing the view.

Returns the new `NSPrintOperation` object. You must run the operation to print the view.

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.3 Constructor 806
- 12.4.4 Constructor(other as `NSPrintOperationMBS`) 807
- 12.4.5 Constructor(view as `DesktopHTMLViewer`, printInfo as `NSPrintInfoMBS = nil`) 807
- 12.4.7 Constructor(view as `NSViewMBS`) 809

12.4. CLASS NSPRINTOPERATIONMBS	809
• 12.4.8 Constructor(view as NSViewMBS, printInfo as NSPrintInfoMBS)	809

12.4.7 Constructor(view as NSViewMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an NSPrintOperation object ready to control the printing of the specified view.

Notes: View: The view whose contents you want to print.

Returns the new NSPrintOperation object. You must run the operation to print the view.

The new NSPrintOperation object uses the settings stored in the shared NSPrintInfo object. This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

• 12.4.3 Constructor	806
• 12.4.4 Constructor(other as NSPrintOperationMBS)	807
• 12.4.5 Constructor(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil)	807
• 12.4.6 Constructor(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil)	808
• 12.4.8 Constructor(view as NSViewMBS, printInfo as NSPrintInfoMBS)	809

12.4.8 Constructor(view as NSViewMBS, printInfo as NSPrintInfoMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an NSPrintOperation object ready to control the printing of the specified view using custom print settings.

Notes: View: The view whose contents you want to print.

PrintInfo: The print settings to use when printing the view.

Returns the new NSPrintOperation object. You must run the operation to print the view.

This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

• 12.4.3 Constructor	806
• 12.4.4 Constructor(other as NSPrintOperationMBS)	807

- 12.4.5 Constructor(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil) 807
- 12.4.6 Constructor(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil) 808
- 12.4.7 Constructor(view as NSViewMBS) 809

12.4.9 context as NSGraphicsMBS

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

Function: Returns the graphics context object used for generating output.

12.4.10 currentOperation as NSPrintOperationMBS

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

Function: Returns the current print operation for this thread.

Notes: The print operation object, or nil if there is no current operation.

12.4.11 currentPage as Integer

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

Function: Returns the current page number being printed.

12.4.12 data as Memoryblock

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

Function: Returns the data in PDF/EPS after the operation finished.

12.4.13 Destructor

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

Function: The destructor.

12.4.14 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new NSPrintOperation object ready to control the copying of EPS graphics from the specified view.

Notes: View: The view containing the data to be turned into EPS data.

rect: The portion of the view (specified in points in the view's coordinate space) to be rendered as EPS data.

After the job is run, use the Data function to get the EPS data.

Returns the new NSPrintOperation object. You must run the operation to generate the EPS data.

The new NSPrintOperation object uses the default NSPrintInfo object. This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.15 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS 811
- 12.4.16 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS 812
- 12.4.17 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string) as NSPrintOperationMBS 812

12.4.15 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new NSPrintOperation object ready to control the copying of EPS graphics from the specified view using the specified print settings.

Notes: View: The view containing the data to be turned into EPS data.

rect: The portion of the view (specified in points in the view's coordinate space) to be rendered as EPS data.

PrintInfo: The print settings to use when generating the EPS data.

After the job is run, use the Data function to get the EPS data.

Returns the new NSPrintOperation object. You must run the operation to generate the EPS data.

This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.14 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS)` as `NSPrintOperationMBS` 811
- 12.4.16 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem)` as `NSPrintOperationMBS` 812
- 12.4.17 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string)` as `NSPrintOperationMBS` 812

12.4.16 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem)` as `NSPrintOperationMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view and write the resulting data to the specified file.

Notes: View: The view containing the data to be turned into EPS data.

rect: The portion of the view (specified in points in the view's coordinate space) to be rendered as EPS data.

path: The path to a file. After the job is run, this file contains the EPS data.

PrintInfo: The print settings to use when generating the EPS data.

Returns the new `NSPrintOperation` object. You must run the operation to generate the EPS data.

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.14 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS)` as `NSPrintOperationMBS` 811
- 12.4.15 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS)` as `NSPrintOperationMBS` 811
- 12.4.17 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string)` as `NSPrintOperationMBS` 812

12.4.17 `EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string)` as `NSPrintOperationMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view and write the resulting data to the specified file.

Notes: View: The view containing the data to be turned into EPS data.

rect: The portion of the view (specified in points in the view's coordinate space) to be rendered as EPS data.

path: The path to a file. After the job is run, this file contains the EPS data.

PrintInfo: The print settings to use when generating the EPS data.

Returns the new NSPrintOperation object. You must run the operation to generate the EPS data. This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread. See also:

- 12.4.14 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS 811
- 12.4.15 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS 811
- 12.4.16 EPSOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS 812

12.4.18 isCopyingOperation as boolean

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

Function: Returns a Boolean value indicating whether the receiver is an EPS or PDF copy operation.

Notes: True if the receiver is an EPS or PDF copy operation; otherwise, false.

12.4.19 NSPrintOperationExistsException as string

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

Function: The name of an exception raised when there is already a print operation in process.

Notes: The methods that raise the NSExcptionMBS exception are the EPSOperation... and printOperation....

12.4.20 pageRange as NSRangeMBS

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

Function: Returns the print order for the pages.

Notes: The print order. For a list of possible values, see Constants.

12.4.21 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view.

Notes: `View`: The view containing the data to be turned into PDF data.

`rect`: The portion of the view (specified in points in the view's coordinate space) to be rendered as PDF data.

After the job is run, the data function gives you the PDF data.

Returns the new `NSPrintOperation` object. You must run the operation to generate the PDF data.

The new `NSPrintOperation` object uses the default `NSPrintInfo` object. This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.22 `PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS` 814
- 12.4.23 `PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS` 815
- 12.4.24 `PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string) as NSPrintOperationMBS` 815

12.4.22 `PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view using the specified print settings.

Notes: `View`: The view containing the data to be turned into PDF data.

`rect`: The portion of the view (specified in points in the view's coordinate space) to be rendered as PDF data.

`PrintInfo`: The print settings to use when generating the PDF data.

After the job is run, the data function returns the PDF data.

Returns the new `NSPrintOperation` object. You must run the operation to generate the PDF data.

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.21 `PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS` 813

12.4. CLASS NSPRINTOPERATIONMBS 815

- 12.4.23 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS 815
- 12.4.24 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string) as NSPrintOperationMBS 815

12.4.23 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new NSPrintOperation object ready to control the copying of PDF graphics from the specified view and write the resulting data to the specified file.

Notes: View: The view containing the data to be turned into PDF data.

rect: The portion of the view (specified in points in the view's coordinate space) to be rendered as PDF data.

path: The path to a file. After the job is run, this file contains the PDF data.

PrintInfo: The print settings to use when generating the PDF data.

Returns the new NSPrintOperation object. You must run the operation to generate the PDF data.

This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.21 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS 813
- 12.4.22 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS 814
- 12.4.24 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string) as NSPrintOperationMBS 815

12.4.24 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, path as string) as NSPrintOperationMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new NSPrintOperation object ready to control the copying of PDF graphics from the specified view and write the resulting data to the specified file.

Notes: View: The view containing the data to be turned into PDF data.

rect: The portion of the view (specified in points in the view's coordinate space) to be rendered as PDF data.

path: The path to a file. After the job is run, this file contains the PDF data.

PrintInfo: The print settings to use when generating the PDF data.

Returns the new NSPrintOperation object. You must run the operation to generate the PDF data.

This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.21 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS) as NSPrintOperationMBS 813
- 12.4.22 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS 814
- 12.4.23 PDFOperationWithView(view as NSViewMBS, rect as NSRectMBS, printInfo as NSPrintInfoMBS, file as folderitem) as NSPrintOperationMBS 815

12.4.25 preferredRenderingQuality as Integer

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

Function: Returns the printing quality.

Notes: The preferred printing quality. See constants for the possible values.

If the print sheet is unresponsive or sluggish due to the time it takes to fully render a page, you can check this method in drawRect: and other printing methods such as beginDocument and knowsPageRange: to determine if the print operation prefers speed over fidelity. Most applications render each page fast enough and do not need to call this method. Only use this method after establishing that best quality rendering does indeed make the user interface unresponsive.

12.4.26 printOperationWithView(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS

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

Function: Creates and returns an NSPrintOperation object ready to control the printing of the specified view using custom print settings.

Example:

```
// print a HTMLViewer
dim n as NSPrintOperationMBS = NSPrintOperationMBS.printOperationWithView(HTMLViewer1)

n.showsPrintPanel = true
n.showsProgressPanel = true
```

```
n.runOperationModalForWindow(self)
```

Notes: View: The view whose contents you want to print.
PrintInfo: The print settings to use when printing the view.

Returns the new NSPrintOperation object. You must run the operation to print the view.

This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.27 printOperationWithView(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS 817
- 12.4.28 printOperationWithView(view as NSViewMBS) as NSPrintOperationMBS 818
- 12.4.29 printOperationWithView(view as NSViewMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS 819

12.4.27 printOperationWithView(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an NSPrintOperation object ready to control the printing of the specified view using custom print settings.

Example:

```
// print a HTMLViewer
dim n as NSPrintOperationMBS = NSPrintOperationMBS.printOperationWithView(HTMLViewer1)

n.showsPrintPanel = true
n.showsProgressPanel = true
n.runOperationModalForWindow(self)
```

Notes: View: The view whose contents you want to print.
PrintInfo: The print settings to use when printing the view.

Returns the new NSPrintOperation object. You must run the operation to print the view.

This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.26 `printOperationWithView(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS` 816
- 12.4.28 `printOperationWithView(view as NSViewMBS) as NSPrintOperationMBS` 818
- 12.4.29 `printOperationWithView(view as NSViewMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS` 819

12.4.28 `printOperationWithView(view as NSViewMBS) as NSPrintOperationMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an `NSPrintOperation` object ready to control the printing of the specified view.

Example:

```
// print a text area
dim textView as NSTextViewMBS = TextArea1.NSTextViewMBS
dim o as NSPrintOperationMBS = NSPrintOperationMBS.printOperationWithView(textView)

o.showsPrintPanel = true
o.runOperationModalForWindow(Window1)
o = nil
```

Notes: View: The view whose contents you want to print.

Returns the new `NSPrintOperation` object. You must run the operation to print the view.

The new `NSPrintOperation` object uses the settings stored in the shared `NSPrintInfo` object. This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.26 `printOperationWithView(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS` 816
- 12.4.27 `printOperationWithView(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS` 817
- 12.4.29 `printOperationWithView(view as NSViewMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS` 819

12.4.29 printOperationWithView(view as NSViewMBS, printInfo as NSPrintInfoMBS) as NSPrintOperationMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an NSPrintOperation object ready to control the printing of the specified view using custom print settings.

Notes: View: The view whose contents you want to print.

PrintInfo: The print settings to use when printing the view.

Returns the new NSPrintOperation object. You must run the operation to print the view.

This method raises an NSPrintOperationExistsException if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

See also:

- 12.4.26 printOperationWithView(view as DesktopHTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS 816
- 12.4.27 printOperationWithView(view as HTMLViewer, printInfo as NSPrintInfoMBS = nil) as NSPrintOperationMBS 817
- 12.4.28 printOperationWithView(view as NSViewMBS) as NSPrintOperationMBS 818

12.4.30 runOperation as boolean

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

Function: Runs the print operation on the current thread.

Notes: Returns true if the operation was successful; otherwise, false.

The operation runs to completion in the current thread, blocking the application. A separate thread is not spawned, even if canSpawnSeparateThread is true. Use runOperationModalForWindow to use document-modal sheets and to allow a separate thread to perform the operation.

12.4.31 runOperationModalForWindow(win as DesktopWindow)

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

Function: Runs the print operation, calling your custom delegate method upon completion.

Notes: win: The document window to receive a print progress sheet.

Calls the printOperationDidRun event.

See also:

- 12.4.32 `runOperationModalForWindow(win as NSWindowMBS)` 820
- 12.4.33 `runOperationModalForWindow(win as window)` 820

12.4.32 `runOperationModalForWindow(win as NSWindowMBS)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Runs the print operation, calling your custom delegate method upon completion.

Notes: win: The document window to receive a print progress sheet.

Calls the `printOperationDidRun` event.

See also:

- 12.4.31 `runOperationModalForWindow(win as DesktopWindow)` 819
- 12.4.33 `runOperationModalForWindow(win as window)` 820

12.4.33 `runOperationModalForWindow(win as window)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Runs the print operation, calling your custom delegate method upon completion.

Notes: win: The document window to receive a print progress sheet.

Calls the `printOperationDidRun` event.

See also:

- 12.4.31 `runOperationModalForWindow(win as DesktopWindow)` 819
- 12.4.32 `runOperationModalForWindow(win as NSWindowMBS)` 820

12.4.34 `setCurrentOperation(operation as NSPrintOperationMBS)`

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

Function: Sets the current print operation for this thread.

Notes: operation: The print operation to make current. You may specify nil to clear the current print operation.

12.4.35 `view as NSViewMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns the view object that generates the actual data for the print operation.

12.4.36 Properties

12.4.37 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

12.4.38 canSpawnSeparateThread as boolean

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

Function: Whether the receiver is allowed to spawn a separate printing thread.

Notes: canSpawnSeparateThread: True if the receiver is allowed to spawn a separate thread; otherwise, false.

If canSpawnSeparateThread is true, an NSThread object is detached when the print panel is dismissed (or immediately, if the panel is not to be displayed). The new thread performs the print operation, so that control can return to your application. A thread is detached only if the print operation is run using the runOperationModalForWindow:delegate:didRunSelector:contextInfo: method. If canSpawnSeparateThread is false, the operation runs on the current thread, blocking the application until the operation completes.

If you send setCanSpawnSeparateThread: to an NSPrintOperation object with an argument of true, then the delegate specified in a subsequent invocation of runOperationModalForWindow may be messaged in that spawned, non-main thread.

(Read and Write computed property)

12.4.39 jobTitle as string

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

Function: The title of the print job.

Notes: A string containing the print job title. If set, this value overrides the title returned by the printing view.

Available in OS X v10.5 and later.

(Read and Write computed property)

12.4.40 `pageOrder` as Integer

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

Function: Returns the print order for the pages.

Notes: The print order. For a list of possible values, see Constants.
(Read and Write computed property)

12.4.41 `printInfo` as `NSPrintInfoMBS`

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

Function: Returns the receiver's `NSPrintInfo` object.

Notes: (Read and Write computed property)

12.4.42 `printPanel` as `NSPrintPanelMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns the `NSPrintPanel` object used when running the operation.

Notes: (Read and Write computed property)

12.4.43 `showsPrintPanel` as boolean

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

Function: Boolean value indicating whether a print panel is displayed during the operation,

Notes: True if the operation displays a print panel; otherwise, false.

Operations that generate EPS or PDF data do not display a print panel (instance of `NSPrintPanel`), regardless of the value returned by this method.

(Read and Write computed property)

12.4.44 `showsProgressPanel` as boolean

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

Function: Whether the receiver displays a progress panel for this operation.

Notes: True if you want to display a progress panel; otherwise, false.

This method does not affect the display of a print panel; that operation is controlled by the `ShowsPrintPanel` method.

Operations that generate EPS or PDF data do not display a progress panel, regardless of the value in the flag parameter.

(Read and Write computed property)

12.4.45 Events

12.4.46 `printOperationDidRun(success as boolean)`

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: The event called when the print operation ends.

12.4.47 Constants

Page Order Constants

Constant	Value	Description
<code>NSAscendingPageOrder</code>	1	Ascending (back to front) page order.
<code>NSDescendingPageOrder</code>	-1	Descending (front to back) page order.
<code>NSSpecialPageOrder</code>	0	The spooler does not rearrange pages—they are printed in the order received by the spooler.
<code>NSUnknownPageOrder</code>	2	No page order specified.

Print Quality Constants

Constant	Value	Description
<code>NSPrintRenderingQualityBest</code>	0	Renders the printing at the best possible quality, regardless of speed. Available in OS X v10.7 and later.
<code>NSPrintRenderingQualityResponsive</code>	1	Sacrifices the least possible amount of rendering quality for speed to maintain a responsive user interface. This option should be used only after establishing that best quality rendering does indeed make the user interface unresponsive. Available in OS X v10.7 and later.

12.5 class NSPrintPanelMBS

12.5.1 class NSPrintPanelMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: An NSPrintPanel object creates the Print panel used to query the user for information about a print job.

Notes: This panel may lets the user select the range of pages to print and the number of copies before executing the Print command.

Print panels can display a simplified interface when printing certain types of data. For example, the panel can display a list of print-setting presets, which lets the user enable print settings in groups as opposed to individually. The JobStyleHint property activates the simplified interface and identifies which presets to display.

Blog Entries

- [MBS Real Studio Plugins, version 12.4pr1](#)

12.5.2 Methods

12.5.3 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow)

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

Function: Displays a Print panel sheet and runs it modally for the specified window.

Notes: printInfo: The printing information for the current job.

win: The window on which to display the sheet.

When the modal session ends, if printPanelDidEnd event is invoked on the object.

See also:

- 12.5.4 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS) 824
- 12.5.5 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window) 825

12.5.4 beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Displays a Print panel sheet and runs it modally for the specified window.

Notes: printInfo: The printing information for the current job.

win: The window on which to display the sheet.

When the modal session ends, if `printPanelDidEnd` event is invoked on the object.

See also:

- 12.5.3 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow)` 824
- 12.5.5 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window)` 825

12.5.5 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as window)`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Displays a Print panel sheet and runs it modally for the specified window.

Notes: `printInfo`: The printing information for the current job.

win: The window on which to display the sheet.

When the modal session ends, if `printPanelDidEnd` event is invoked on the object.

See also:

- 12.5.3 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as DesktopWindow)` 824
- 12.5.4 `beginSheetWithPrintInfo(printInfo as NSPrintInfoMBS, win as NSWindowMBS)` 824

12.5.6 Constructor

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

Function: Initializes a new print panel.

12.5.7 `NSPrintAllPresetsJobStyleHint` as string

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

Function: One of the values can be passed to the `jobStyleHint` property to activate the simplified Print panel interface and specify which presets to display.

Notes: Output appropriate to all graphics types. Equivalent to Core Printing's `kMPMPresetGraphicsTypeAll`.

Available in OS X v10.6 and later.

12.5.8 `NSPrintNoPresetsJobStyleHint` as string

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

Function: One of the values can be passed to the `jobStyleHint` property to activate the simplified Print panel interface and specify which presets to display.

Notes: Output excludes all graphics printing. Equivalent to Core Printing's `kPMPresetGraphicsTypeNone`. Available in OS X v10.6 and later.

12.5.9 `NSPrintPhotoJobStyleHint` as string

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

Function: One of the values can be passed to the `jobStyleHint` property to activate the simplified Print panel interface and specify which presets to display.

Notes: Output contains photographic data.

12.5.10 `printInfo` as `NSPrintInfoMBS`

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

Function: Returns the printing information associated with the running Print panel.

Notes: The current printing information. May return nil if the Print panel is not currently running.

This method is a convenience method that your delegate can use to get the printing information while the Print Panel is visible.

Available in OS X v10.5 and later.

12.5.11 `printPanel` as `NSPrintPanelMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns a new `PrintPanel` object.

12.5.12 `runModal` as Integer

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

Function: Displays the receiver's Print panel and begins the modal loop.

Notes: `NSCancelButton` (0) if the user clicks the Cancel button; otherwise `NSOKButton` (1).

This method uses the printing information associated with the current printing operation.

12.5.13 runModalWithPrintInfo(printInfo as NSPrintInfoMBS) as Integer

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

Function: Displays the receiver's Print panel and runs the modal loop using the specified printing information.

Notes: printInfo: The printing information to use while displaying the Print panel.

Returns NSCancelButton (0) if the user clicks the Cancel button; otherwise NSOKButton (1).

12.5.14 Properties

12.5.15 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

12.5.16 defaultButtonTitle as string

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

Function: The title of the Print panel's default button.

Notes: defaultButtonTitle: The string to use for the button title.

You can use this method to change the default button title from "Print" to something more appropriate for your usage of the panel. For example, if you are using the Print panel to save a representation of the document to a file, you might change the title to "Save".

Available in OS X v10.5 and later.

(Read and Write computed property)

12.5.17 helpAnchor as string

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

Function: The HTML help anchor for the print panel.

Notes: helpAnchor: The anchor name in your Apple Help file. This parameter should contain just the name portion of the HTML anchor element.

For information on how to insert anchors into your Apple Help files, see Authoring User Help in Apple Help

Programming Guide.

Available in OS X v10.5 and later.

(Read and Write computed property)

12.5.18 `jobStyleHint` as string

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

Function: The type of content the Print panel is representing.

Notes: hint: For a list of supported job style hints, see Job Style Hints functions. Pass nil to this method to deactivate the simplified Print panel interface and use the standard interface instead (the equivalent of Core Printing's `kPMPresetGraphicsTypeGeneral`).

This method controls the set of items that appear in the Presets menu of the simplified Print panel interface. (Read and Write computed property)

12.5.19 `options` as Integer

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

Function: The configuration options for the Print panel.

Notes: The configuration options, which you specify by adding together the appropriate constant values. Available in OS X v10.5 and later.

(Read and Write computed property)

12.5.20 `Events`

12.5.21 `printPanelDidEnd(returnCode as Integer)`

Plugin Version: 12.4, Platform: macOS, Targets: .

Function: The event called when the sheet ends.

Notes: The value passed as `returnCode` is either `NSCancelButton` or `NSOKButton`. The value `NSOKButton` is returned even if the user clicked the Preview button.

12.5.22 `Constants`

Print Panel Option Constants

Constant	Value	Description
NSPrintPanelShowsCopies	1	The Print panel includes a field for manipulating the number of copies printed. This field is separate from any accessory views. Available in OS X v10.5 and later.
NSPrintPanelShowsOrientation	8	The Print panel includes a control for manipulating the page orientation. This control is separate from any accessory views. Available in OS X v10.5 and later.
NSPrintPanelShowsPageRange	2	The Print panel includes a set of fields for manipulating the range of pages being printed. These fields are separate from any accessory views. Available in OS X v10.5 and later.
NSPrintPanelShowsPageSetupAccessory	256	The Print panel includes a separate accessory view for manipulating the page size, orientation, and scaling attributes. Page setup fields that are configured for display on the main portion of the Print panel appear there and not on this accessory panel. Available in OS X v10.5 and later.
NSPrintPanelShowsPaperSize	4	The Print panel includes a control for manipulating the paper size of the document. This control is separate from any accessory views. Available in OS X v10.5 and later.
NSPrintPanelShowsPreview	131072	The Print panel displays a built-in preview of the document content. This option is only appropriate when the Print panel is used in conjunction with an NSPrintOperation object to print a document. Available in OS X v10.5 and later.
NSPrintPanelShowsPrintSelection	32	The Print panel includes an additional selection option for paper range. This control is separate from any accessory views. Available in OS X v10.6 and later.
NSPrintPanelShowsScaling	16	The Print panel includes a control for scaling the printed output. This control is separate from any accessory views. Available in OS X v10.5 and later.

Chapter 13

Cocoa Regular Expressions

13.1 class NSDataDetectorMBS

13.1.1 class NSDataDetectorMBS

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: A specialized regular expression object that matches natural language text for predefined data patterns.

Example:

```
// create detector for links and emails
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypeLink
Dim Error As NSErrorMBS
Dim DataDetector As New NSDataDetectorMBS(types, error)

Dim s As String = "Email us at john@mbs.test today!"

// look for the links now
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show first link
Dim r As NSTextCheckingResultMBS = m(0)
MsgBox r.URL
```

Notes: Currently the NSDataDetectorMBS class can match dates, addresses, links, phone numbers and transit information.

(links includes email addresses)

The results of matching content is returned as NSTextCheckingResultMBS objects. However, the NS-

TextCheckingResultMBS objects returned by NSDataDetectorMBS are different from those returned by the base class NSRegularExpressionMBS. Results returned by NSDataDetector will be of one of the data detectors types, depending on the type of result being returned, and they will have corresponding properties. For example, results of type NSTextCheckingTypeDate have a date, timeZone, and duration; results of type NSTextCheckingTypeLink have a URL, and so forth.

Subclass of the NSRegularExpressionMBS class.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr1](#)
- [Data Detectors as context menu](#)
- [XDC Anywhere - MBS Xojo Plugins](#)
- [MBS Xojo Plugin, June 2021 News](#)
- [News from the MBS Xojo Plugins Version 21.1](#)
- [Video about MBS Xojo Plugins 21.1](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 21.1](#)
- [MBS Xojo Plugins, version 21.1pr6](#)
- [Data Detectors for Xojo](#)

Videos

- [MBS Xojo Plugins 21.1](#)
- [XDC Anywhere - MBS Xojo Plugins](#)
- [MBS Xojo Videos - MBS Xojo Plugin, June 2021 News](#)

13.1.2 Methods

13.1.3 Constructor(checkingTypes as Integer, byref error as NSErrorMBS)

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Initializes and returns a data detector instance.

Example:

```
// create detector for phone number
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypePhoneNumber
Dim Error As NSErrorMBS
Dim DataDetector As New NSDataDetectorMBS(types, error)

Dim s As String = "Call us at (555)-555-555 today!"
```

```
// look for the phone number now
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show first phone number found
Dim r As NSTextCheckingResultMBS = m(0)
MsgBox r.phoneNumber
```

Notes: checkingTypes: The checking types. The supported checking types are a subset of the types NSTextCheckingType. Those constants can be combined using the BitWiseOR() function.
error: An out parameter that if an error occurs during initialization contains the encountered error.

Returns the newly initialized data detector. If an error was encountered returns nil, and error contains the error.

Currently, the supported data detectors checkingTypes are: NSTextCheckingTypeDate, NSTextCheckingTypeAddress, NSTextCheckingTypeLink, NSTextCheckingTypePhoneNumber, and NSTextCheckingTypeTransitInformation.

See also:

- 13.1.4 Constructor(script as string, options as Integer, byref error as NSErrorMBS) 833

13.1.4 Constructor(script as string, options as Integer, byref error as NSErrorMBS)

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Private constructor.

Notes: Here to prevent you from accidentally calling the constructor for NSRegularExpressionMBS class.

See also:

- 13.1.3 Constructor(checkingTypes as Integer, byref error as NSErrorMBS) 832

13.1.5 copy as NSDataDetectorMBS

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a copy of the object.

13.1.6 dataDetectorWithTypes(checkingTypes as Integer, byref error as NSErrorMBS) as NSDataDetectorMBS

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Creates and returns a new data detector instance.

Example:

```
// create detector for time, date and duration
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypeDate
Dim Error As NSErrorMBS
Dim DataDetector As NSDataDetectorMBS = NSDataDetectorMBS.dataDetectorWithTypes(types, error)

Dim s As String = "Let's meet sunday at 3 pm!"

// look for it:
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show time
Dim r As NSTextCheckingResultMBS = m(0)

Dim d As date = r.date
MsgBox d.SQLiteDateTime
```

Notes: `checkingTypes`: The checking types. The supported checking types are a subset of the types specified in `NSTextCheckingType`. Those constants can be combined using the bitwise OR operator.

`error`: An out parameter that if an error occurs during initialization contains the encountered error.

Returns the newly initialized data detector. If an error was encountered returns nil, and `error` contains the error.

Discussion

Currently, the supported data detectors `checkingTypes` are: `NSTextCheckingTypeDate`, `NSTextCheckingTypeAddress`, `NSTextCheckingTypeLink`, `NSTextCheckingTypePhoneNumber`, and `NSTextCheckingTypeTransitInformation`.

13.1.7 Properties

13.1.8 `checkingTypes` as Integer

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the checking types for this data detector.

Notes: The supported subset of checking types are specified in `NSTextCheckingType`. Those constants can be combined using the bitwise OR operator.

Currently, the supported data detectors `checkingTypes` are: `NSTextCheckingTypeDate`, `NSTextCheckingTypeAddress`, `NSTextCheckingTypeLink`, `NSTextCheckingTypePhoneNumber`, and `NSTextCheckingTypeTransitInformation`.

(Read only property)

13.2 class NSRegularExpressionMBS

13.2.1 class NSRegularExpressionMBS

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: An immutable representation of a compiled regular expression that you apply to Unicode strings.
Notes: The fundamental matching method for NSRegularExpressionMBS is a Block iterator method that allows clients to supply a Block object which will be invoked each time the regular expression matches a portion of the target string. There are additional convenience methods for returning all the matches as an array, the total number of matches, the first match, and the range of the first match.

An individual match is represented by an instance of the NSTextCheckingResultMBS class, which carries information about the overall matched range (via its range property), and the range of each individual capture group (via the rangeAtIndex: method). For basic NSRegularExpression objects, these match results will be of type NSTextCheckingTypeRegularExpression, but subclasses may use other types.

see also

<https://developer.apple.com/documentation/foundation/nsregularexpression>

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 21.1](#)
- [MBS Xojo Plugins, version 21.1pr6](#)

13.2.2 Methods

13.2.3 Constructor(script as string, options as Integer, byref error as NSErrorMBS)

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns an initialized NSRegularExpression instance with the specified regular expression pattern and options.

Notes: pattern: The regular expression pattern to compile.

options: The regular expression options that are applied to the expression during matching. See NSRegularExpressionOptions for possible values.

error: An out value that returns any error encountered during initialization. Returns an NSErrorMBS object if the regular expression pattern is invalid; otherwise returns nil.

Returns an instance of NSRegularExpressionMBS for the specified regular expression and options

13.2.4 copy as `NSRegularExpressionMBS`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a copy of the object.

13.2.5 `enumerateMatches(text as string, options as Integer, range as NSRangeMBS = nil)`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Enumerates the string allowing the event to handle each regular expression match.

Notes: Calls `enumerateMatch` event.

`text`: The string.

`options`: The matching options to report. See `MatchingOptions` for the supported values.

`range`: The range of the string to test.

This method is the fundamental matching method for regular expressions and is suitable for overriding by subclasses. There are additional convenience methods for returning all the matches as an array, the total number of matches, the first match, and the range of the first match.

By default, the event iterator method calls the event precisely once for each match, with a non-nil result and the appropriate flags. The client may then stop the operation by setting the contents of `stop` to true. The `stop` argument is an out-only argument. You should only ever set this Boolean to true within the event.

If the `MatchingReportProgress` matching option is specified, the event will also be called periodically during long-running match operations, with nil result and `MatchingProgress` matching flag set in the event, `flags` parameter, at which point the client may again stop the operation by setting the contents of `stop` to true.

If the `MatchingReportCompletion` matching option is specified, the event object will be called once after matching is complete, with nil result and the `MatchingCompleted` matching flag is set in the flags passed to the event, plus any additional relevant `MatchingFlags` from among `MatchingHitEnd`, `MatchingRequiredEnd`, or `MatchingInternalError`.

`MatchingProgress` and `MatchingCompleted` matching flags have no effect for methods other than this method. The `MatchingHitEnd` matching flag is set in the flags passed to the event if the current match operation reached the end of the search range. The `MatchingRequiredEnd` matching flag is set in the flags passed to the event if the current match depended on the location of the end of the search range.

The `MatchingFlags` matching flag is set in the flags passed to the event if matching failed due to an internal error (such as an expression requiring exponential memory allocations) without examining the entire search

range.

The `MatchingAnchored`, `MatchingWithTransparentBounds`, and `MatchingWithoutAnchoringBounds` regular expression options, specified in the `options` property specified when the regular expression instance is created, can apply to any `match` or `replace` method.

If `MatchingAnchored` matching option is specified, matches are limited to those at the start of the search range.

If `MatchingWithTransparentBounds` matching option is specified, matching may examine parts of the string beyond the bounds of the search range, for purposes such as word boundary detection, lookahead, etc.

If `MatchingWithoutAnchoringBounds` matching option is specified, `^` and `$` will not automatically match the beginning and end of the search range, but will still match the beginning and end of the entire string.

`MatchingWithTransparentBounds` and `MatchingWithoutAnchoringBounds` matching options have no effect if the search range covers the entire string.

13.2.6 `escapedPatternForString(text as String) as String`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns a string by adding backslash escapes as necessary to protect any characters that would match as pattern metacharacters.

Notes: `text`: The string.

Returns the escaped string.

Returns a string by adding backslash escapes as necessary to the given string, to escape any characters that would otherwise be treated as pattern metacharacters. You typically use this method to match on a particular string within a larger pattern.

For example, the string `"(N/A)"` contains the pattern metacharacters `(`, `/`, and `)`. The result of adding backslash escapes to this string is `"\\(N\\/A\\)"`.

13.2.7 `escapedTemplateForString(text as String) as String`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns a template string by adding backslash escapes as necessary to protect any characters that would match as pattern metacharacters.

Notes: `text`: The template string

Returns the escaped template string.

Returns a string by adding backslash escapes as necessary to the given string, to escape any characters that would otherwise be treated as pattern metacharacters. You typically use this method to match on a particular string within a larger pattern.

For example, the string "(N/A)" contains the pattern metacharacters (, /, and). The result of adding backslash escapes to this string is "\\(N\\/A\\)".

See Flag Options for the format of the resulting template string.

13.2.8 firstMatch(text as string, options as Integer, range as NSRangeMBS = nil) as NSTextCheckingResultMBS

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the first match of the regular expression within the specified range of the string.

Notes: text: The string to search.

options: The matching options to use. See NSMatchingOptions for possible values.

range: The range of the string to search.

Returns an NSTextCheckingResultMBS object. This result gives the overall matched range via its range property, and the range of each individual capture group via its rangeAtIndex: method. The range { NSNotFound, 0 } is returned if one of the capture groups did not participate in this particular match.

This is a convenience method that calls enumerateMatches.

13.2.9 matches(text as string, options as Integer, range as NSRangeMBS = nil) as NSTextCheckingResultMBS()

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns an array containing all the matches of the regular expression in the string.

Notes: text: The string to search.

options: The matching options to use. See Matching Options for possible values.

range: The range of the string to search.

Returns an array of NSTextCheckingResultMBS objects. Each result gives the overall matched range via its range property, and the range of each individual capture group via its rangeAtIndex: method. The range { NSNotFound, 0 } is returned if one of the capture groups did not participate in this particular match.

This is a convenience method that calls enumerateMatches passing the appropriate string, options, and range.

13.2.10 numberOfMatches(text as string, options as Integer, range as NSRangeMBS = nil) as Integer

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the number of matches of the regular expression within the specified range of the string.

Notes: text: The string to search.

options: The matching options to use. See Matching Options for possible values.

range: The range of the string to search.

Returns the number of matches of the regular expression.

This is a convenience method that calls enumerateMatches.

13.2.11 rangeOfFirstMatch(text as string, options as Integer, range as NSRangeMBS = nil) as NSRangeMBS

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the range of the first match of the regular expression within the specified range of the string.

Notes: string: The string to search.

options: The matching options to use. See Matching Options for possible values.

range: The range of the string to search.

Returns the range of the first match. Returns { NSNotFound, 0 } if no match is found.

This is a convenience method that calls enumerateMatches.

13.2.12 regularExpressionWithPattern(pattern as String, options as Integer, byref error as NSErrorMBS) as NSDataDetectorMBS

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Creates an NSRegularExpression instance with the specified regular expression pattern and options.

Notes: pattern: The regular expression pattern to compile.

options: The matching options. See constants for possible values. The values can be combined using the bitwise OR operator.

error: An out value that returns any error encountered during initialization. Returns an NSErrorMBS object if the regular expression pattern is invalid; otherwise returns nil.

Returns an instance of `NSRegularExpression` for the specified regular expression and options.

13.2.13 `replaceMatches(byref text as string, options as Integer, range as NSRangeMBS = nil, template as String) as Integer`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Replaces regular expression matches within the mutable string using the template string.

Notes: `text`: The Text to search and replace values within.

`options`: The matching options to use. See `NSMatchingOptions` for possible values.

`range`: The range of the string to search.

`template`: The substitution template used when replacing matching instances.

Returns the number of matches.

See `Flag Options` for the format of template.

13.2.14 `replacementStringForResult(result as NSTextCheckingResultMBS, text as string, offset as Integer, template as String) as String`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Used to perform template substitution for a single result for clients implementing their own replace functionality.

Notes: `result`: The result of the single match.

`text`: The string from which the result was matched.

`offset`: The offset to be added to the location of the result in the string.

`template`: See `Flag Options` for the format of template.

Returns a replacement string.

For clients implementing their own replace functionality, this is a method to perform the template substitution for a single result, given the string from which the result was matched, an offset to be added to the location of the result in the string (for example, in cases that modifications to the string moved the result since it was matched), and a replacement template.

This is an advanced method that is used only if you wanted to iterate through a list of matches yourself and do the template replacement for each one, plus maybe some other calculation that you want to do in code, then you would use this at each step.

13.2.15 `stringByReplacingMatches(text as string, options as Integer, range as NSRangeMBS = nil, template as String) as String`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns a new string containing matching regular expressions replaced with the template string.

Notes: text: The string to search for values within.

options: The matching options to use. See Matching Options for possible values.

range: The range of the string to search.

template: The substitution template used when replacing matching instances.

Returns a string with matching regular expressions replaced by the template string.

See Flag Options for the format of template.

13.2.16 Properties

13.2.17 Handle as Integer

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: The internal object reference.

Notes: (Read only property)

13.2.18 `numberOfCaptureGroups as Integer`

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the number of capture groups in the regular expression.

Notes: A capture group consists of each possible match within a regular expression. Each capture group can then be used in a replacement template to insert that value into a replacement string.

This value puts a limit on the values of n for \$ n in templates, and it determines the number of ranges in the returned NSTextCheckingResultMBS instances returned in the match... methods.

An exception will be generated if you attempt to access a result with an index value exceeding numberOfCaptureGroups-1.

(Read only property)

13.2.19 options as Integer

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the options used when the regular expression option was created.

Notes: The options property specifies aspects of the regular expression matching that are always used when matching the regular expression. For example, if the expression is case sensitive, allows comments, ignores metacharacters, etc. See constants for a complete discussion of the possible constants and their meanings. (Read only property)

13.2.20 pattern as String

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the regular expression pattern.

Notes: (Read only property)

13.2.21 Events

13.2.22 enumerateMatch(text as String, options as Integer, result as NSTextCheckingResultMBS, flags as Integer, byref stop as Boolean)

Plugin Version: 21.1, Platforms: macOS, iOS, Targets: .

Function: The event for enumerateMatches callback.

Notes: We pass through text and options for debugging.

Set stop to true to exit early.

Can be used with addHandler if needed.

result: An NSTextCheckingResultMBS specifying the match. This result gives the overall matched range via its range property, and the range of each individual capture group via its rangeAtIndex: method. The range { NSNotFound, 0 } is returned if one of the capture groups did not participate in this particular match.

flags: The current state of the matching progress. See Matching Flags for the possible values.

stop: A reference to a Boolean value. The event can set the value to true to stop further processing of the array. The stop argument is an out-only argument. You should only ever set this Boolean to true within the event.

13.2.23 Constants

Matching Options

Constant	Value	Description
MatchingAnchored	4	Specifies that matches are limited to those at the start of the search range. See enumerateMatches for a description of the constant in context.
MatchingReportCompletion	2	Call the Block once after the completion of any matching. This option has no effect for methods other than enumerateMatches. See enumerateMatches for a description of the constant in context.
MatchingReportProgress	1	Call the Block periodically during long-running match operations. This option has no effect for methods other than enumerateMatches. See enumerateMatches for a description of the constant in context.
MatchingWithoutAnchoringBounds	16	Specifies that ^and \$ will not automatically match the beginning and end of the search range, but will still match the beginning and end of the entire string. This constant has no effect if the search range contains the entire string. See enumerateMatches for a description of the constant in context.
MatchingWithTransparentBounds	8	Specifies that matching may examine parts of the string beyond the bounds of the search range, for purposes such as word boundary detection, lookahead, etc. This constant has no effect if the search range contains the entire string. See enumerateMatches for a description of the constant in context.

Progress

Constant	Value	Description
MatchingCompleted	2	Set when the Block is called after matching has completed.
MatchingHitEnd	4	Set when the current match operation reached the end of the search range.
MatchingInternalError	16	Set when matching failed due to an internal error.
MatchingProgress	1	Set when the Block is called to report progress during a long-running match operation.
MatchingRequiredEnd	8	Set when the current match depended on the location of the end of the search range.

Options

Constant	Value	Description
RegularExpressionAllowCommentsAndWhitespace	2	Ignore whitespace and #-prefixed comments in the pattern.
RegularExpressionAnchorsMatchLines	16	Allow ^and \$ to match the start and end of lines.
RegularExpressionCaseInsensitive	1	Match letters in the pattern independent of case.
RegularExpressionDotMatchesLineSeparators	8	Allow . to match any character, including line separators.
RegularExpressionIgnoreMetacharacters	4	Treat the entire pattern as a literal string.
RegularExpressionUseUnicodeWordBoundaries	64	Use Unicode TR#29 to specify word boundaries (otherwise, POSIX-style expression word boundaries are used).
RegularExpressionUseUnixLineSeparators	32	Treat only \n as a line separator (otherwise, all standard line separators are used).

13.3 class NSTextCheckingResultMBS

13.3.1 class NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: An occurrence of textual content found during the analysis of a block of text, such as when matching a regular expression.

Example:

```
// create detector for Address
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypeAddress
Dim Error As NSErrorMBS
Dim DataDetector As New NSDataDetectorMBS(types, error)

Dim s As String = "Let's meet at Hauptstraße 123 in 12345 Berlin, Deutschland."

// look for it:
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show Address
Dim r As NSTextCheckingResultMBS = m(0)

Dim dic As Dictionary = r.components
Dim lines() As String
For Each key As Variant In dic.keys
lines.Append key.StringValue+": "+dic.Value(key).StringValue
Next

MsgBox Join(lines,EndOfLine)
```

Notes: On both iOS and macOS, instances of NSTextCheckingResultMBS are returned by the NSRegularExpression class and the NSDataDetectorMBS class to indicate the discovery of content. In those cases, what is found may be a match for a regular expression or a date, address, phone number, and so on. In macOS, instances of NSTextCheckingResultMBS are returned by the NSSpellCheckerMBS object to describe the results of spelling, grammar, or text-substitution actions.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [Data Detectors as context menu](#)
- [News from the MBS Xojo Plugins Version 21.1](#)
- [MBS Xojo Plugins, version 21.1pr6](#)
- [Data Detectors for Xojo](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 19.1](#)

- [MBS Xojo Plugins, version 19.1pr6](#)

Xojo Developer Magazine

- [17.3, page 11: News](#)

13.3.2 Methods

13.3.3 addressCheckingResult(Range as NSRangeMBS, components as Dictionary) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified address components.

Notes: components: A dictionary containing the address components. The dictionary keys are described in Keys for Address Components.

13.3.4 alternativeStrings as String()

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Returns alternative strings.

13.3.5 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The private constructor.

13.3.6 copy as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates a copy of the result.

13.3.7 correctionCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result after detecting a possible correction.

Notes: range: The range of the detected result.

replacementString: The suggested replacement string.

Returns an `NSTextCheckingResult` with the specified range and a `resultType` of `NSTextCheckingTypeSpelling`.

See also:

- 13.3.8 `correctionCheckingResult(Range as NSRangeMBS, replacementString as String, alternativeStrings() as String) as NSTextCheckingResultMBS` 846

13.3.8 `correctionCheckingResult(Range as NSRangeMBS, replacementString as String, alternativeStrings() as String) as NSTextCheckingResultMBS`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result after detecting a possible correction.

Notes: range: The range of the detected result.

replacementString: The suggested replacement string.

Returns an `NSTextCheckingResultMBS` with the specified range and a `resultType` of `NSTextCheckingTypeSpelling`.

See also:

- 13.3.7 `correctionCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS` 845

13.3.9 `dashCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified dash corrected replacement string.

Notes: range: The range of the detected result.

replacementString: The replacement string.

Returns an `NSTextCheckingResult` with the specified range and a `resultType` of `NSTextCheckingTypeDash`.

13.3.10 `dateCheckingResult(Range as NSRangeMBS, date as Date) as NSTextCheckingResultMBS`

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

Function: Creates and returns a text checking result with the specified date.

See also:

- 13.3.11 dateCheckingResult(Range as NSRangeMBS, date as Date, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS 847
- 13.3.12 dateCheckingResult(Range as NSRangeMBS, date as DateTime) as NSTextCheckingResultMBS 847
- 13.3.13 dateCheckingResult(Range as NSRangeMBS, date as DateTime, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS 848

13.3.11 dateCheckingResult(Range as NSRangeMBS, date as Date, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS

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

Function: Creates and returns a text checking result with the specified date, time zone, and duration.

Notes: range: The range of the detected result.

date: The detected date.

timeZone: The detected time zone.

duration: The detected duration.

Returns an NSTextCheckingResult with the specified range and a resultType of NSTextCheckingTypeDate.

See also:

- 13.3.10 dateCheckingResult(Range as NSRangeMBS, date as Date) as NSTextCheckingResultMBS 846
- 13.3.12 dateCheckingResult(Range as NSRangeMBS, date as DateTime) as NSTextCheckingResultMBS 847
- 13.3.13 dateCheckingResult(Range as NSRangeMBS, date as DateTime, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS 848

13.3.12 dateCheckingResult(Range as NSRangeMBS, date as DateTime) as NSTextCheckingResultMBS

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

Function: Creates and returns a text checking result with the specified date.

See also:

- 13.3.10 dateCheckingResult(Range as NSRangeMBS, date as Date) as NSTextCheckingResultMBS 846
- 13.3.11 dateCheckingResult(Range as NSRangeMBS, date as Date, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS 847
- 13.3.13 dateCheckingResult(Range as NSRangeMBS, date as DateTime, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS 848

13.3.13 `dateCheckingResult(Range as NSRangeMBS, date as DateTime, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS`

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

Function: Creates and returns a text checking result with the specified date, time zone, and duration.

Notes: range: The range of the detected result.

date: The detected date.

timeZone: The detected time zone.

duration: The detected duration.

Returns an `NSTextCheckingResult` with the specified range and a `resultType` of `NSTextCheckingTypeDate`. See also:

- 13.3.10 `dateCheckingResult(Range as NSRangeMBS, date as Date) as NSTextCheckingResultMBS` 846
- 13.3.11 `dateCheckingResult(Range as NSRangeMBS, date as Date, timeZone as NSTimeZoneMBS, duration as double) as NSTextCheckingResultMBS` 847
- 13.3.12 `dateCheckingResult(Range as NSRangeMBS, date as DateTime) as NSTextCheckingResultMBS` 847

13.3.14 `grammarCheckingResult(Range as NSRangeMBS, details() as Dictionary) as NSTextCheckingResultMBS`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified array of grammatical errors.

Notes: range: The range of the detected result.

details: An array of details regarding the grammatical errors. This array of strings is suitable for presenting to the user.

Returns an `NSTextCheckingResultMBS` with the specified range and a `resultType` of `NSTextCheckingTypeGrammar`.

13.3.15 `grammarDetails as Dictionary()`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The details of a located grammatical type checking result.

Notes: This array of strings is suitable for presenting to the user.

13.3.16 linkCheckingResult(Range as NSRangeMBS, URL as String) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified URL.

13.3.17 NSTextCheckingAirlineKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the components dictionary.

Notes: A key that corresponds to the airline of a transit result.

13.3.18 NSTextCheckingCityKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the city component of the address.

13.3.19 NSTextCheckingCountryKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the country component of the address.

13.3.20 NSTextCheckingFlightKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the components dictionary.

Example:

```
// create detector for transit information
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypeTransitInformation
Dim Error As NSErrorMBS
Dim DataDetector As New NSDataDetectorMBS(types, error)

Dim s As String = "Let's fly with LH444 to Atlanta!"
```

```
// look for it:
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show airline
Dim r As NSTextCheckingResultMBS = m(0)

Dim dic As Dictionary = r.components
Dim Airline As String = dic.Lookup(NSTextCheckingResultMBS.NSTextCheckingAirlineKey, "?")
Dim Flight As String = dic.Lookup(NSTextCheckingResultMBS.NSTextCheckingFlightKey, "?")

MsgBox Airline+" "+Flight
```

Notes: A key that corresponds to the flight component of a transit result.

13.3.21 NSTextCheckingJobTitleKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the job component of the address.

13.3.22 NSTextCheckingNameKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the name component of the address.

13.3.23 NSTextCheckingOrganizationKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the organization component of the address.

13.3.24 NSTextCheckingPhoneKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the phone number component of the address.

13.3.25 NSTextCheckingStateKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the state or province component of the address.

13.3.26 NSTextCheckingStreetKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the street address component of the address.

13.3.27 NSTextCheckingZIPKey as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: One of the constants to identify the possible keys returned in the addressComponents dictionary.

Notes: A key that corresponds to the zip code or postal code component of the address.

13.3.28 orthographyCheckingResult(Range as NSRangeMBS, orthography as NSOrthographyMBS) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified orthography.

Notes: range: The range of the detected result.

orthography: An orthography object that describes the script.

Returns an NSTextCheckingResult with the specified range and a resultType of NSTextCheckingType-Orthography.

13.3.29 `phoneNumberCheckingResult(Range as NSRangeMBS, phoneNumber as String) as NSTextCheckingResultMBS`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified phone number.

13.3.30 `quoteCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified quote-balanced replacement string.

Notes: range: The range of the detected result.

replacementString: The replacement string.

Returns an NSTextCheckingResult with the specified range and a resultType of NSTextCheckingTypeQuote.

13.3.31 `rangeAtIndex(index as Integer) as NSRangeMBS`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Returns the result type that the range represents.

Notes: index: The index of the result.

Returns the range of the result.

A result must have at least one range, but may optionally have more (for example, to represent regular expression capture groups).

Passing rangeAtIndex the value 0 always returns the value of the the range property. Additional ranges, if any, will have indexes from 1 to numberOfRanges-1.

13.3.32 `rangeWithName(name as string) as NSRangeMBS`

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Returns range with name.

13.3.33 replacementCheckingResult(Range as NSRangeMBS, replacementString as String) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified replacement string.

13.3.34 resultByAdjustingRangesWithOffset(offset as Integer) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Returns a new text checking result after adjusting the ranges as specified by the offset.

Notes: offset: The amount the ranges are adjusted.

Returns a new NSTextCheckingResultMBS instance with the adjusted range or ranges.

13.3.35 spellCheckingResult(Range as NSRangeMBS) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the range of a misspelled word.

Notes: range: The range of the detected result.

Returns an NSTextCheckingResult with the specified range and a resultType of NSTextCheckingTypeSpelling.

13.3.36 transitInformationCheckingResult(Range as NSRangeMBS, components as Dictionary) as NSTextCheckingResultMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates and returns a text checking result with the specified transit information.

Notes: components: A dictionary containing the transit components. The currently supported keys are NSTextCheckingAirlineKey and NSTextCheckingFlightKey.

13.3.37 Properties

13.3.38 addressComponents as Dictionary

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The address dictionary of a type checking result.

Notes: The dictionary keys are described in Keys for Address Components.

(Read only property)

13.3.39 components as Dictionary

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: A dictionary containing the components of a type checking result.

Example:

```
// create detector for Address
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypeAddress
Dim Error As NSErrorMBS
Dim DataDetector As New NSDataDetectorMBS(types, error)

Dim s As String = "Let's meet at Hauptstraße 123 in 12345 Berlin, Deutschland."

// look for it:
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show Address
Dim r As NSTextCheckingResultMBS = m(0)

Dim dic As Dictionary = r.components
Dim lines() As String
For Each key As Variant In dic.keys
lines.Append key.StringValue+": "+dic.Value(key).StringValue
Next

MsgBox Join(lines,EndOfLine)
```

Notes: Currently used by the transit checking result. The supported keys are located in Keys for Transit Components.

(Read only property)

13.3.40 date as Date

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

Function: The date component of a type checking result.

Example:

```
// create detector for time, date and duration
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypeDate
Dim Error As NSErrorMBS
Dim DataDetector As NSDataDetectorMBS = NSDataDetectorMBS.dataDetectorWithTypes(types, error)

Dim s As String = "Let's meet sunday at 3 pm!"

// look for it:
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show time
Dim r As NSTextCheckingResultMBS = m(0)

Dim d As date = r.date
MsgBox d.SQLiteDateTime
```

Notes: (Read only property)

13.3.41 dateTime as DateTime

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

Function: The date component of a type checking result.

Notes: (Read only property)

13.3.42 duration as Double

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The duration component of a type checking result.

Notes: (Read only property)

13.3.43 Handle as Integer

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

13.3.44 numberOfRanges as Integer

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Returns the number of ranges.

Notes: A result must have at least one range, but may optionally have more (for example, to represent regular expression capture groups).

Passing rangeAtIndex the value 0 always returns the value of the the range property. Additional ranges, if any, will have indexes from 1 to numberOfRanges-1.

(Read only property)

13.3.45 orthography as NSOrthographyMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The detected orthography of a type checking result.

Notes: (Read only property)

13.3.46 phoneNumber as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The phone number of a type checking result.

Example:

```
// create detector for phone number
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypePhoneNumber
Dim Error As NSErrorMBS
Dim DataDetector As New NSDataDetectorMBS(types, error)

Dim s As String = "Call us at (555)-555-555 today!"

// look for the phone number now
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show first phone number found
Dim r As NSTextCheckingResultMBS = m(0)
MsgBox r.phoneNumber
```

Notes: (Read only property)

13.3.47 range as NSRangeMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Returns the range of the result that the receiver represents.

Notes: This property will be present for all returned NSTextCheckingResultMBS instances.
(Read only property)

13.3.48 regularExpression as NSRegularExpressionMBS

Plugin Version: 21.1, Platform: macOS, Targets: All.

Function: The regular expression of a type checking result.

Notes: (Read only property)

13.3.49 replacementString as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: A replacement string from one of a number of replacement checking results.

Notes: (Read only property)

13.3.50 resultType as Integer

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Returns the text checking result type that the receiver represents.

Notes: The possible result types for the built in checking capabilities are described in NSTextCheckingType.
This property will be present for all returned NSTextCheckingResultMBS instances.
(Read only property)

13.3.51 timeZone as NSTimeZoneMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The time zone component of a type checking result.

Notes: (Read only property)

13.3.52 URL as String

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: The URL of a type checking result.

Example:

```
// create detector for links and emails
Dim types As Integer = NSTextCheckingResultMBS.NSTextCheckingTypeLink
Dim Error As NSErrorMBS
Dim DataDetector As New NSDataDetectorMBS(types, error)

Dim s As String = "Email us at john@mbs.test today!"

// look for the links now
Dim m() As NSTextCheckingResultMBS = DataDetector.matches(s, 0)

// show first link
Dim r As NSTextCheckingResultMBS = m(0)
MsgBox r.URL
```

Notes: (Read only property)

13.3.53 Constants

Checking Types

Constant	Value	Description
NSTextCheckingAllCustomTypes	&hfffffff00000000	Checking types that can be used by clients.
NSTextCheckingAllSystemTypes	&hfffffff	Checking types supported by the system. The first 32 types.
NSTextCheckingAllTypes	&hfffffffffffffff	All possible checking types, both system- and user-supported.
NSTextCheckingTypeAddress	16	Attempts to locate addresses.
NSTextCheckingTypeCorrection	512	Performs autocorrection on misspelled words.
NSTextCheckingTypeDash	128	Replaces dashes with em-dashes.
NSTextCheckingTypeDate	8	Attempts to locate dates.
NSTextCheckingTypeGrammar	4	Checks grammar.
NSTextCheckingTypeLink	32	Attempts to locate URL links or emails.
NSTextCheckingTypeOrthography	1	Attempts to identify the language.
NSTextCheckingTypePhoneNumber	2048	Matches a phone number.
NSTextCheckingTypeQuote	64	Replaces quotes with smart quotes.
NSTextCheckingTypeRegularExpression	1024	Matches a regular expression.
NSTextCheckingTypeReplacement	256	Replaces characters such as (c) with the appropriate symbol.
NSTextCheckingTypeSpelling	2	Checks spelling.
NSTextCheckingTypeTransitInformation	4096	Matches a transit information, for example, flight information.

Chapter 14

Cocoa Tasks

14.1 class NSFileHandleMBS

14.1.1 class NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: NSFileHandle objects provide an object-oriented wrapper for accessing open files or communications channels.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingAtFile(f)

if n<>Nil then
  MsgBox n.readDataToEndOfFile
end if
```

Notes: Please call closeFile on the end if you want to close the file.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.1](#)
- [MBS Xojo / Real Studio Plugins, version 14.1pr4](#)

14.1.2 Methods

14.1.3 `acceptConnectionInBackgroundAndNotify`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Accepts a socket connection (for stream-type sockets only) in the background and creates a file handle for the "near" (client) end of the communications channel.

Notes: This method is asynchronous. In a separate "safe" thread it accepts a connection, creates a file handle for the other end of the connection, and returns that object to the client by posting an `NSFileHandleConnectionAcceptedNotification` in the run loop of the client. The notification includes as data a `userInfo` dictionary containing the created `NSFileHandle` object; access this object using the `NSFileHandleNotificationFileHandleItem` key.

The receiver must be created by an `fileHandleWithFileDescriptor` message that takes as an argument a stream-type socket created by the appropriate system routine. The object that will write data to the returned file handle must add itself as an observer of `NSFileHandleConnectionAcceptedNotification`.

Note that this method does not continue to listen for connection requests after it posts `NSFileHandleConnectionAcceptedNotification`. If you want to keep getting notified, you need to call `acceptConnectionInBackgroundAndNotify` again in your observer method.

14.1.4 `AvailableBytes as Integer`

Plugin Version: 14.1, Platform: macOS, Targets: All.

Function: Queries number of available bytes.

Notes: Returns -1 if query failed.

You can use this value with `readDataOfLength` function to have it not block.

14.1.5 `availableData as MemoryBlock`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the data available through the receiver.

Notes: If the receiver is a file, returns the data obtained by reading the file from the file pointer to the end of the file. If the receiver is a communications channel, reads up to a buffer of data and returns it; if no data is available, the method blocks. Returns an empty data object if the end of file is reached. Raises `NSFileHandleOperationException` if attempts to determine file-handle type fail or if attempts to read from the file or channel fail.

14.1.6 `closeFile`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Disallows further access to the represented file or communications channel and signals end of file on communications channels that permit writing.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim e as NSErrorMBS
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingFromFile(f,e)

if e<>Nil then
  MsgBox e.localizedDescription
end if

if n<>Nil then
  MsgBox n.readDataOfLength(5)
  n.closeFile
end if
```

Notes: The file or communications channel is available for other uses after the file handle represented by the receiver is closed. Further read and write messages sent to a file handle to which closeFile has been sent raises an exception.

Sending closeFile to a file handle does not cause its deallocation. The deallocation of an NSFileHandle object deletes its descriptor and closes the represented file or channel unless the NSFileHandle object was created with fileHandleWithFileDescriptor with false as the parameter argument.

14.1.7 Constructor

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

Function: The private constructor.

14.1.8 fileDescriptor as Integer

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the file descriptor associated with the receiver.

Notes: Returns the POSIX file descriptor associated with the receiver.

You can send this message to file handles originating from both file descriptors and file handles and receive a valid file descriptor so long as the file handle is open. If the file handle has been closed by sending it closeFile, this method raises an exception.

14.1.9 fileHandleForReadingAtPath(path as folderitem) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading the file, device, or named socket at the specified path.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingAtPath(f)

if n<>Nil then
  MsgBox n.readDataToEndOfFile
end if
```

Notes: path: The path to the file, device, or named socket to access.

Returns the initialized file handle, or nil if no file exists at path.

The file pointer is set to the beginning of the file. The returned object responds only to NSFileHandle read... messages.

14.1.10 fileHandleForReadingAtPath(path as string) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading the file, device, or named socket at the specified path.

Notes: path: The path to the file, device, or named socket to access.

Returns the initialized file handle, or nil if no file exists at path.

The file pointer is set to the beginning of the file. The returned object responds only to NSFileHandle read... messages.

14.1.11 fileHandleForReadingFromFile(URL as folderitem, byref error as NSErrorMBS) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading the file, device, or named socket at the specified

URL.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim e as NSErrorMBS
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingFromFile(f,e)

if e<>Nil then
  MsgBox e.localizedDescription
end if

if n<>Nil then
  MsgBox n.readDataOfLength(5)
  MsgBox str(n.offsetInFile) // shows 5
end if
```

Notes: url: The URL of the file, device, or named socket to access.

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns the initialized file handle, or nil if no file exists at url.

The file pointer is set to the beginning of the file. The returned object responds only to NSFileHandleRead... messages.

14.1.12 fileHandleForReadingFromURL(URL as string, byref error as NSErrorMBS) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading the file, device, or named socket at the specified URL.

Notes: url: The URL of the file, device, or named socket to access.

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns the initialized file handle, or nil if no file exists at url.

The file pointer is set to the beginning of the file. The returned object responds only to NSFileHandleRead... messages.

14.1.13 fileHandleForUpdatingAtPath(path as folderitem) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading and writing to the file, device, or named socket at the specified path.

Notes: path: The path to the file, device, or named socket to access.

Returns the initialized file handle, or nil if no file exists at path.

The file pointer is set to the beginning of the file. The returned object responds to both NSFileHandle read... messages and writeData.

14.1.14 fileHandleForUpdatingAtPath(path as string) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading and writing to the file, device, or named socket at the specified path.

Notes: path: The path to the file, device, or named socket to access.

Returns the initialized file handle, or nil if no file exists at path.

The file pointer is set to the beginning of the file. The returned object responds to both NSFileHandle read... messages and writeData.

14.1.15 fileHandleForUpdatingFile(URL as folderitem, byref error as NSErrorMBS) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading and writing to the file, device, or named socket at the specified URL.

Notes: url: The URL of the file, device, or named socket to access.

error: If an error occurs, upon return contains an NSError object that describes the problem.

The initialized file handle, or nil if no file exists at url.

The file pointer is set to the beginning of the file. The returned object responds to both NSFileHandle read... messages and writeData.

14.1.16 fileHandleForUpdatingURL(URL as string, byref error as NSErrorMBS) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for reading and writing to the file, device, or named socket at the specified URL.

Notes: url: The URL of the file, device, or named socket to access.

error: If an error occurs, upon return contains an NSError object that describes the problem.

The initialized file handle, or nil if no file exists at url.

The file pointer is set to the beginning of the file. The returned object responds to both NSFileHandleRead... messages and writeData.

14.1.17 fileHandleForWritingAtPath(path as folderitem) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for writing to the file, device, or named socket at the specified path.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForWritingAtPath(f)

if n<>Nil then
n.writeData "Hello World"
n.closeFile
end if
```

Notes: path: The path to the file, device, or named socket to access.

Returns the initialized file handle, or nil if no file exists at path.

The file pointer is set to the beginning of the file. The returned object responds only to writeData.

14.1.18 fileHandleForWritingAtPath(path as string) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for writing to the file, device, or named socket at the specified path.

Notes: path: The path to the file, device, or named socket to access.

Returns the initialized file handle, or nil if no file exists at path.

The file pointer is set to the beginning of the file. The returned object responds only to writeData.

14.1.19 fileHandleForWritingToFile(URL as folderitem, byref error as NSErrorMBS) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for writing to the file, device, or named socket at the specified URL.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim e as NSErrorMBS
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForWritingToFile(f, e)

if e<>Nil then
  MsgBox e.localizedDescription
else
  n.writeData "Hello World"
  n.closeFile
end if
```

Notes: url: The URL of the file, device, or named socket to access.

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns the initialized file handle, or nil if no file exists at url.

The file pointer is set to the beginning of the file. The returned object responds only to writeData.

14.1.20 fileHandleForWritingToURL(URL as string, byref error as NSErrorMBS) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized for writing to the file, device, or named socket at the specified URL.

Notes: url: The URL of the file, device, or named socket to access.

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns the initialized file handle, or nil if no file exists at url.

The file pointer is set to the beginning of the file. The returned object responds only to writeData.

14.1.21 fileHandleWithFileDescriptor(fd as Integer) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized with a file descriptor.

Notes: You can create a file handle for a socket by using the result of a socket call as fileDescriptor.

The object creating a file handle using this method owns fileDescriptor and is responsible for its disposition.

See also:

- 14.1.22 fileHandleWithFileDescriptor(fd as Integer, closeOnDealloc as boolean) as NSFileHandleMBS
867

14.1.22 fileHandleWithFileDescriptor(fd as Integer, closeOnDealloc as boolean) as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle initialized with a file descriptor.

Notes: You can create a file handle for a socket by using the result of a socket call as fileDescriptor.

The object creating a file handle using this method owns fileDescriptor and is responsible for its disposition.

closeOnDealloc: True if the file descriptor should be closed when the receiver is deallocated, otherwise false.

See also:

- 14.1.21 fileHandleWithFileDescriptor(fd as Integer) as NSFileHandleMBS 867

14.1.23 fileHandleWithNullDevice as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns a file handle associated with a null device.

Notes: You can use null-device file handles as "placeholders" for standard-device file handles or in collection objects to avoid exceptions and other errors resulting from messages being sent to invalid file handles. Read messages sent to a null-device file handle return an end-of-file indicator (an empty NSData object) rather than raise an exception. Write messages are no-ops, whereas fileDescriptor returns an illegal value. Other methods are no-ops or return "sensible" values.

14.1.24 fileHandleWithStandardError as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the file handle associated with the standard error file.

Example:

```
// for GUI apps this ends on the console: (see console.app)

dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleWithStandardError

n.writeData "Hello World"
```

Notes: Conventionally this is a terminal device to which error messages are sent. There is one standard error file handle per process; it is a shared instance.

14.1.25 fileHandleWithStandardInput as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the file handle associated with the standard input file.

Notes: Conventionally this is a terminal device on which the user enters a stream of data. There is one standard input file handle per process; it is a shared instance.

14.1.26 fileHandleWithStandardOutput as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the file handle associated with the standard output file.

Example:

```
// for GUI apps this ends on the console: (see console.app)

dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleWithStandardOutput
```

```
n.writeData "Hello World"
```

Notes: Conventionally this is a terminal device that receives a stream of data from a program. There is one standard output file handle per process; it is a shared instance.

14.1.27 NSFileHandleConnectionAcceptedNotification as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: This notification is posted when an NSFileHandle object establishes a socket connection between two processes, creates an NSFileHandle object for one end of the connection, and makes this object available to observers by putting it in the userInfo dictionary.

Notes: To cause the posting of this notification, you must send either acceptConnectionInBackgroundAndNotify to an NSFileHandle object representing a server stream-type socket.

The notification object is the NSFileHandle object that sent the notification. The userInfo dictionary contains the following information:

NSFileHandleNotificationFileHandleItem	The NSFileHandle object representing the "near" end of a socket connection
NSFileHandleError	An integer representing the UNIX-type error which occurred

14.1.28 NSFileHandleDataAvailableNotification as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: This notification is posted when the background thread determines that data is currently available for reading in a file or at a communications channel.

Notes: The observers can then issue the appropriate messages to begin reading the data. To cause the posting of this notification, you must send either waitForDataInBackgroundAndNotify or waitForDataInBackgroundAndNotifyForModes: to an appropriate NSFileHandle object.

The notification object is the NSFileHandle object that sent the notification. This notification does not contain a userInfo dictionary.

14.1.29 NSFileHandleNotificationDataItem as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: A key in the userInfo dictionary in a NSFileHandleReadCompletionNotification and NSFileHandleReadToEndOfFileCompletionNotification.

Notes: The corresponding value is a memoryblock containing the available data read from a socket connection.

14.1.30 `NSFileHandleNotificationFileHandleItem` as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: A key in the userInfo dictionary in a `NSFileHandleConnectionAcceptedNotification` notification.

Notes: The corresponding value is the `NSFileHandle` object handle representing the "near" end of a socket connection.

14.1.31 `NSFileHandleNotificationMonitorModes` as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Currently unused.

14.1.32 `NSFileHandleOperationException` as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Constant that defines the name of a file operation exception.

Notes: Raised by `NSFileHandle` if attempts to determine file-handle type fail or if attempts to read from a file or channel fail.

14.1.33 `NSFileHandleReadCompletionNotification` as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: One of the notification strings for the file handle class.

Notes: This notification is posted when the background thread reads the data currently available in a file or at a communications channel. It makes the data available to observers by putting it in the userInfo dictionary. To cause the posting of this notification, you must send either `readInBackgroundAndNotify` to an appropriate `NSFileHandle` object.

The notification object is the `NSFileHandle` object that sent the notification. The userInfo dictionary contains the following information:

<code>NSFileHandleNotificationDataItem</code>	An string containing the available data read from a socket connection
<code>NSFileHandleError</code>	An integer representing the UNIX-type error which occurred

14.1.34 `NSFileHandleReadToEndOfFileCompletionNotification` as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: This notification is posted when the background thread reads all data in the file or, if a communications channel, until the other process signals the end of data.

Notes: It makes the data available to observers by putting it in the userInfo dictionary. To cause the posting of this notification, you must send either `readToEndOfFileInBackgroundAndNotify` to an appropriate `NSFileHandle` object.

The notification object is the `NSFileHandle` object that sent the notification. The userInfo dictionary contains the following information:

`NSFileHandleNotificationDataItem`: A string containing the available data read from a socket connection

`NSFileHandleError` An integer representing the UNIX-type error which occurred

14.1.35 `readDataOfLength(length as Integer) as MemoryBlock`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Reads data up to a specified number of bytes from the receiver.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingAtFile(f)

if n<>Nil then
  MsgBox n.readDataOfLength(5)
end if
```

Notes: length: The number of bytes to read from the receiver.

Returns the data available through the receiver up to a maximum of length bytes.

If the receiver is a file, returns the data obtained by reading from the file pointer to length or to the end of the file, whichever comes first. If the receiver is a communications channel, the method reads data from the channel up to length. Returns an empty memoryblock if the file is positioned at the end of the file or if an end-of-file indicator is returned on a communications channel. Raises `NSFileHandleOperationException` if attempts to determine file-handle type fail or if attempts to read from the file or channel fail.

14.1.36 readDataToEndOfFile as MemoryBlock

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the data available through the receiver up to the end of file or maximum number of bytes.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingAtFile(f)

if n<>Nil then
  MsgBox n.readDataToEndOfFile
end if
```

Notes: Returns the data available through the receiver up to UINT_MAX bytes (the maximum value for unsigned integers) or, if a communications channel, until an end-of-file indicator is returned.

This method invokes readDataOfLength as part of its implementation.

14.1.37 readInBackgroundAndNotify

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Reads from the file or communications channel in the background and posts a notification when finished.

Notes: This method performs an asynchronous availableData operation on a file or communications channel and posts an NSFileHandleReadCompletionNotification to the client process's run loop.

The length of the data is limited to the buffer size of the underlying operating system. The notification includes a userInfo dictionary that contains the data read; access this object using the NSFileHandleNotificationDataItem key.

Any object interested in receiving this data asynchronously must add itself as an observer of NSFileHandleReadCompletionNotification. In communication via stream-type sockets, the receiver is often the object returned in the userInfo dictionary of NSFileHandleConnectionAcceptedNotification.

Note that this method does not cause a continuous stream of notifications to be sent. If you wish to keep getting notified, you'll also need to call readInBackgroundAndNotify in your observer method.

14.1.38 readToEndOfFileInBackgroundAndNotify

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Reads to the end of file from the file or communications channel in the background and posts a notification when finished.

Example:

```
dim path as string = "/tmp/NSFileHandle async reading.txt"  
dim f as FolderItem = GetFolderItem(path, FolderItem.PathTypeShell)  
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingAtFile(f)
```

```
n.readToEndOfFileInBackgroundAndNotify
```

Notes: This method performs an asynchronous `readToEndOfFile` operation on a file or communications channel and posts an `NSFileHandleReadToEndOfFileCompletionNotification` to the client process's run loop. The notification includes a `userInfo` dictionary that contains the data read; access this object using the `NSFileHandleNotificationDataItem` key.

Any object interested in receiving this data asynchronously must add itself as an observer of `NSFileHandleReadToEndOfFileCompletionNotification`. In communication via stream-type sockets, the receiver is often the object returned in the `userInfo` dictionary of `NSFileHandleConnectionAcceptedNotification`.

14.1.39 seekToEndOfFile as UInt64

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Puts the file pointer at the end of the file referenced by the receiver and returns the new file offset.

Notes: Returns the file offset with the file pointer at the end of the file. This is therefore equal to the size of the file.

Raises an exception if the message is sent to an `NSFileHandle` object representing a pipe or socket or if the file descriptor is closed.

14.1.40 seekToFileOffset(offset as UInt64)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Moves the file pointer to the specified offset within the file represented by the receiver.

Notes: Raises an exception if the message is sent to an `NSFileHandle` object representing a pipe or socket, if the file descriptor is closed, or if any other error occurs in seeking.

14.1.41 synchronizeFile

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Causes all in-memory data and attributes of the file represented by the receiver to be written to permanent storage.

Notes: This method should be invoked by programs that require the file to always be in a known state. An invocation of this method does not return until memory is flushed.

14.1.42 truncateFileAtOffset(offset as UInt64)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Truncates or extends the file represented by the receiver to a specified offset within the file and puts the file pointer at that position.

Notes: offset: The offset within the file that will mark the new end of the file.

If the file is extended (if offset is beyond the current end of file), the added characters are null bytes.

14.1.43 waitForDataInBackgroundAndNotify

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Checks to see if data is available in a background thread.

Notes: When the data becomes available, the thread notifies all observers with `NSFileHandleDataAvailableNotification`. After the notification has been posted, the thread is terminated.

14.1.44 writeData(data as MemoryBlock)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Synchronously writes data to the file, device, pipe, or socket represented by the receiver.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForWritingAtFile(f)

if n<>Nil then
n.writeData "Hello World"
n.closeFile
end if
```

Notes: If the receiver is a file, writing takes place at the file pointer's current position. After it writes the

data, the method advances the file pointer by the number of bytes written. Raises an exception if the file descriptor is closed or is not valid, if the receiver represents an unconnected pipe or socket endpoint, if no free space is left on the file system, or if any other writing error occurs.

14.1.45 Properties

14.1.46 Handle as Integer

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The internal handle to the NSFileHandle object.

Notes: (Read and Write property)

14.1.47 offsetInFile as UInt64

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the position of the file pointer within the file represented by the receiver.

Example:

// file must exist for this sample:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim e as NSErrorMBS
dim n as NSFileHandleMBS = NSFileHandleMBS.fileHandleForReadingFromFile(f,e)

if e<>Nil then
  MsgBox e.localizedDescription
end if

if n<>Nil then
  MsgBox n.readDataOfLength(5)
  MsgBox str(n.offsetInFile) // shows 5
end if
```

Notes: The position of the file pointer within the file represented by the receiver.

Raises an exception if the message is sent to a file handle representing a pipe or socket or if the file descriptor is closed.

(Read and Write computed property)

14.2 class NSPipeMBS

14.2.1 class NSPipeMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: NSPipe objects provide an object-oriented interface for accessing pipes.

Notes: An NSPipe object represents both ends of a pipe and enables communication through the pipe. A pipe is a one-way communications channel between related processes; one process writes data, while the other process reads that data. The data that passes through the pipe is buffered; the size of the buffer is determined by the underlying operating system. NSPipe is an abstract class, the public interface of a class cluster.

Blog Entries

- [MBS Xojo Plugins, version 17.4pr2](#)

14.2.2 Methods

14.2.3 Constructor

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The constructor to create a new pipe.

Notes: Handle is not zero on success.

14.2.4 fileHandleForReading as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the receiver's read file handle.

Notes: You use the returned file handle to read from the pipe using NSFileHandle's read methods—availableData, readDataToEndOfFile, and readDataOfLength.

You don't need to send closeFile to this object or explicitly release the object after you have finished using it.

14.2.5 fileHandleForWriting as NSFileHandleMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the receiver's write file handle.

Notes: You use the returned file handle to write to the pipe using NSFileHandle's writeData: method. When you are finished writing data to this object, send it a closeFile message to delete the descriptor. Deleting the descriptor causes the reading process to receive an end-of-data signal (an empty memoryblock).

14.2.6 pipe as NSPipeMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns an initialized NSPipe object.

Notes: Returns nil if the method encounters errors while attempting to create the pipe or the NSFileHandle objects that serve as endpoints of the pipe.

14.2.7 Properties

14.2.8 Handle as Integer

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The internal reference to the NSPipe object.

Notes: (Read and Write property)

14.3 class NSTaskMBS

14.3.1 class NSTaskMBS

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Using the NSTask class, your program can run another program as a subprocess and can monitor that program's execution.

Example:

// Launch "ls -l -a -t" in the current directory, and then read the result into a string:

```
dim task as new NSTaskMBS

task.launchPath = "/bin/ls"

dim arguments(-1) as string = array("-l", "-a", "-t")

task.setArguments arguments

dim pipe as new NSPipeMBS

task.setStandardOutput pipe

dim file as NSFileHandleMBS = pipe.fileHandleForReading

task.launch

dim data as string = file.readDataToEndOfFile
dim text as string = DefineEncoding(data, encodings.UTF8)

MsgBox text
```

Notes: An NSTask object creates a separate executable entity; it differs from NSThread in that it does not share memory space with the process that creates it.

A task operates within an environment defined by the current values for several items: the current directory, standard input, standard output, standard error, and the values of any environment variables. By default, an NSTask object inherits its environment from the process that launches it. If there are any values that should be different for the task, for example, if the current directory should change, you must change the value before you launch the task. A task's environment cannot be changed while it is running.

An NSTask object can only be run once. Subsequent attempts to run the task raise an error.

This class is comparable to the shell classe built into Xojo.

See also WindowsProcessMBS and WindowsShellExecuteMBS (Windows only), ConsoleExecuteMBS and ShellMBS (cross platform).

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 15.3pr1](#)

14.3.2 Methods

14.3.3 arguments as string()

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The command arguments that should be used to launch the executable.

14.3.4 Constructor

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

Notes: On success the handle value is not zero.

14.3.5 Destructor

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The destructor.

14.3.6 interrupt

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sends an interrupt signal to the receiver and all of its subtasks.

Notes: If the task terminates as a result, which is the default behavior, an NSTaskDidTerminateNotification gets sent to the default notification center. This method has no effect if the receiver was already launched and has already finished executing. If the receiver has not been launched yet, this method raises an NSInvalidArgumentException.

It is not always possible to interrupt the receiver because it might be ignoring the interrupt signal. interrupt sends SIGINT.

14.3.7 launch

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Launches the task represented by the receiver.

Example:

```
// Launch "ls -l -a -t" in the current directory, and then read the result into a string:
```

```
dim task as new NSTaskMBS
task.launchPath = "/bin/ls"
dim arguments(-1) as string = array("-l", "-a", "-t")
task.setArguments arguments
dim pipe as new NSPipeMBS
task.setStandardOutput pipe
dim file as NSFileHandleMBS = pipe.fileHandleForReading
task.launch
dim data as string = file.readDataToEndOfFile
dim text as string = DefineEncoding(data, encodings.UTF8)
MsgBox text
```

Notes: Raises an `NSInvalidArgumentException` if the launch path has not been set or is invalid or if it fails to create a process.

If you get an exception with `posix_spawn` and error 13, that's a permission denied.

14.3.8 `launchedTaskWithLaunchPath(path as string, arguments() as string) as NSTaskMBS`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and launches a task with a specified executable and arguments.

Example:

```
dim args(-1) as string
dim task as NSTaskMBS = NSTaskMBS.launchedTaskWithLaunchPath("/bin/ls", args)
```

Notes: path: The path to the executable.

arguments: An array of strings that supplies the arguments to the task.

The task inherits its environment from the process that invokes this method.

The `NSTask` object converts both path and the strings in arguments to appropriate C-style strings (using

fileSystemRepresentation) before passing them to the task via `argv []` . The strings in arguments do not undergo shell expansion, so you do not need to do special quoting, and shell variables, such as `$ PWD`, are not resolved.

14.3.9 NSTaskDidTerminateNotification as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The notification name used to notify you that the task terminated.

Notes: Posted when the task has stopped execution. This notification can be posted either when the task has exited normally or as a result of `terminate` being sent to the NSTask object. If the NSTask object gets released, however, this notification will not get sent, as the port the message would have been sent on was released as part of the task release. The observer method can use `terminationStatus` to determine why the task died. See "Ending an NSTask" for an example.

The notification object is the NSTask object that was terminated. This notification does not contain a `userInfo` dictionary.

14.3.10 resume as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Resumes execution of the receiver task that had previously been suspended with a `suspend` message.

Notes: If multiple `suspend` messages were sent to the receiver, an equal number of `resume` messages must be sent before the task resumes execution.

14.3.11 setArguments(arguments()) as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the command arguments that should be used to launch the executable.

Example:

```
// Performing complex pipelines.
// You can create multiple NSTasks and a bunch of NSPipes and hook them together,
// or you can use the "sh -c" trick to feed a shell a command, and let it parse
// it and set up all the IPC. This pipeline cats /usr/share/dict/words, finds
// all the words with 'ham' in them, reverses them, and shows you the last 5.
```

```
dim task as new NSTaskMBS
```

```
task.LaunchPath="/bin/sh"
```

```

dim arguments(-1) as string
arguments.Append "-c"
arguments.Append "cat /usr/share/dict/words | grep -i ham | rev | tail -5"
task.setArguments arguments

```

14.3.12 setStandardError(p as NSFileHandleMBS)

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the standard error for the receiver.

Notes: This method can be used with NSPipeMBS or NSFileHandleMBS object.

If file is an NSPipe object, launching the receiver automatically closes the write end of the pipe in the current task. Don't create a handle for the pipe and pass that as the argument, or the write end of the pipe won't be closed automatically.

If this method isn't used, the standard error is inherited from the process that created the receiver. This method raises an NSInvalidArgumentException if the receiver has already been launched.

See also:

- 14.3.13 setStandardError(p as NSPipeMBS) 882

14.3.13 setStandardError(p as NSPipeMBS)

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the standard error for the receiver.

Notes: This method can be used with NSPipeMBS or NSFileHandleMBS object.

If file is an NSPipe object, launching the receiver automatically closes the write end of the pipe in the current task. Don't create a handle for the pipe and pass that as the argument, or the write end of the pipe won't be closed automatically.

If this method isn't used, the standard error is inherited from the process that created the receiver. This method raises an NSInvalidArgumentException if the receiver has already been launched.

See also:

- 14.3.12 setStandardError(p as NSFileHandleMBS) 882

14.3.14 setStandardInput(p as NSFileHandleMBS)

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the standard input for the receiver.

Notes: file: The standard input for the receiver, which can be either an NSFileHandle or an NSPipe object.

If file is an NSPipe object, launching the receiver automatically closes the read end of the pipe in the current task. Don't create a handle for the pipe and pass that as the argument, or the read end of the pipe won't be closed automatically.

If this method isn't used, the standard input is inherited from the process that created the receiver. This method raises an NSInvalidArgumentException if the receiver has already been launched.

See also:

- 14.3.15 setStandardInput(p as NSPipeMBS) 883

14.3.15 setStandardInput(p as NSPipeMBS)

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the standard input for the receiver.

Notes: file: The standard input for the receiver, which can be either an NSFileHandle or an NSPipe object.

If file is an NSPipe object, launching the receiver automatically closes the read end of the pipe in the current task. Don't create a handle for the pipe and pass that as the argument, or the read end of the pipe won't be closed automatically.

If this method isn't used, the standard input is inherited from the process that created the receiver. This method raises an NSInvalidArgumentException if the receiver has already been launched.

See also:

- 14.3.14 setStandardInput(p as NSFileHandleMBS) 882

14.3.16 setStandardOutput(p as NSFileHandleMBS)

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the standard output for the receiver.

Notes: file: The standard output for the receiver, which can be either an NSFileHandle or an NSPipe object.

If file is an NSPipe object, launching the receiver automatically closes the write end of the pipe in the current task. Don't create a handle for the pipe and pass that as the argument, or the write end of the pipe won't be closed automatically.

If this method isn't used, the standard output is inherited from the process that created the receiver. This method raises an NSInvalidArgumentException if the receiver has already been launched.

See also:

- 14.3.17 setStandardOutput(p as NSPipeMBS) 884

14.3.17 setStandardOutput(p as NSPipeMBS)

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the standard output for the receiver.

Notes: file: The standard output for the receiver, which can be either an NSFileHandle or an NSPipe object.

If file is an NSPipe object, launching the receiver automatically closes the write end of the pipe in the current task. Don't create a handle for the pipe and pass that as the argument, or the write end of the pipe won't be closed automatically.

If this method isn't used, the standard output is inherited from the process that created the receiver. This method raises an NSInvalidArgumentException if the receiver has already been launched.

See also:

- 14.3.16 setStandardOutput(p as NSFileHandleMBS)

883

14.3.18 standardError as Variant

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the standard error file used by the receiver.

Notes: The standard error file used by the receiver.

Standard error is where all diagnostic messages are sent. The object returned is either an NSFileHandle or an NSPipe instance, depending on what type of object was passed to setStandardError.

14.3.19 standardInput as Variant

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the standard input file used by the receiver.

Notes: The standard input file used by the receiver.

Standard input is where the receiver takes its input from unless otherwise specified. The object returned is either an NSFileHandle or an NSPipe instance, depending on what type of object was passed to the setStandardInput method.

14.3.20 standardOutput as Variant

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the standard output file used by the receiver.

Notes: Standard output is where the receiver displays its output. The object returned is either an NSFileHandle or an NSPipe instance, depending on what type of object was passed to the setStandardOutput method.

14.3.21 suspend as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Suspends execution of the receiver task.

Notes: Returns true if the receiver was successfully suspended, false otherwise.

Multiple suspend messages can be sent, but they must be balanced with an equal number of resume messages before the task resumes execution.

14.3.22 terminate

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sends a terminate signal to the receiver and all of its subtasks.

Notes: If the task terminates as a result, which is the default behavior, an NSTaskDidTerminateNotification gets sent to the default notification center. This method has no effect if the receiver was already launched and has already finished executing. If the receiver has not been launched yet, this method raises an NSInvalidArgumentException.

It is not always possible to terminate the receiver because it might be ignoring the terminate signal. terminate sends SIGTERM.

14.3.23 waitUntilExit

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Block until the receiver is finished.

Example:

```
dim args(-1) as string
dim task as NSTaskMBS = NSTaskMBS.launchedTaskWithLaunchPath("/bin/ls", args)
```

```
task.waitUntilExit
```

```
MsgBox "done"
```

Notes: This method first checks to see if the receiver is still running using isRunning. Then it polls the

current run loop using `NSDefaultRunLoopMode` until the task completes.

14.3.24 Properties

14.3.25 `currentDirectoryPath` as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The task's current directory.

Notes: (Read and Write property)

14.3.26 `Handle` as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal reference to the `NSTask` object.

Notes: (Read and Write property)

14.3.27 `isRunning` as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns whether the receiver is still running.

Notes: (Read only property)

14.3.28 `launchPath` as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The path of the receiver's executable.

Notes: (Read and Write property)

14.3.29 `processIdentifier` as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's process identifier.

Example:

```

dim args(-1) as string
dim task as NSTaskMBS = NSTaskMBS.launchedTaskWithLaunchPath("/bin/ls", args)

MsgBox "PID: "+str(task.processIdentifier)

```

Notes: (Read only property)

14.3.30 qualityOfService as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The quality of service setting for this application.

Notes: read-only after the task is launched.

(Read and Write property)

14.3.31 terminationReason as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the reason the task was terminated.

Example:

// Launch "ls -l -a -t" in the current directory, and then read the result into a string:

```

dim args(-1) as string
dim task as NSTaskMBS = NSTaskMBS.launchedTaskWithLaunchPath("/bin/ls", args)

if not task.isRunning then
dim status as Integer = Task.terminationReason

MsgBox "Task termination reason is: "+str(status)
end if

```

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

see this constants:

NSTaskTerminationReasonExit	= 1	The task exited normally.
NSTaskTerminationReasonUncaughtSignal	= 2	The task exited due to an uncaught signal.

(Read only property)

14.3.32 terminationStatus as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the exit status returned by the receiver's executable.

Example:

```
// Launch "ls -l -a -t" in the current directory, and then read the result into a string:

dim args(-1) as string
dim task as NSTaskMBS = NSTaskMBS.launchedTaskWithLaunchPath("/bin/ls", args)

if not task.isRunning then
dim status as Integer = Task.terminationStatus

MsgBox "Task return value is: "+str(status)
end if
```

Notes: The exit status returned by the receiver's executable.

Each task defines and documents how its return value should be interpreted. For example, many commands return 0 if they complete successfully or an error code if they don't. You'll need to look at the documentation for that task to learn what values it returns under what circumstances.

This method raises an `NSInvalidArgumentException` if the receiver is still running. Verify that the receiver is not running before you use it.

(Read only property)

14.3.33 environment as dictionary

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A dictionary of variables for the environment from which the receiver was launched.

Notes: The dictionary keys are the environment variable names.

(Read and Write computed property)

14.3.34 Events

14.3.35 Terminated

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

Function: Event called when task terminated.

14.3.36 Constants

Constants

Constant	Value	Description
NSTaskTerminationReasonExit	1	One of the constants used to specify the values that are returned by <code>terminationReason</code> . The task exited normally.
NSTaskTerminationReasonUncaughtSignal	2	One of the constants used to specify the values that are returned by <code>terminationReason</code> . The task exited due to an uncaught signal.

Quality of Service Constants

Constant	Value	Description
NSQualityOfServiceBackground	&h09	Background QoS is used for work that is not user initiated or visible. In general, a user is unaware that this work is even happening and it will run in the most efficient manner while giving the most deference to higher QoS work. For example, pre-fetching content, search indexing, backups, and syncing of data with external systems.
NSQualityOfServiceDefault	-1	Default QoS indicates the absence of QoS information. Whenever possible QoS information will be inferred from other sources. If such inference is not possible, a QoS between UserInitiated and Utility will be used.
NSQualityOfServiceUserInitiated	&h19	UserInitiated QoS is used for performing work that has been explicitly requested by the user and for which results must be immediately presented in order to allow for further user interaction. For example, loading an email after a user has selected it in a message list.
NSQualityOfServiceUserInteractive	&h21	UserInteractive QoS is used for work directly involved in providing an interactive UI such as processing events or drawing to the screen.
NSQualityOfServiceUtility	&h11	Utility QoS is used for performing work which the user is unlikely to be immediately waiting for the results. This work may have been requested by the user or initiated automatically, does not prevent the user from further interaction, often operates at user-visible timescales and may have its progress indicated to the user by a non-modal progress indicator. This work will run in an energy-efficient manner, in deference to higher QoS work when resources are constrained. For example, periodic content updates or bulk file operations such as media import.

Chapter 15

Cocoa Text

15.1 class NSMutableParagraphStyleMBS

15.1.1 class NSMutableParagraphStyleMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: NSMutableParagraphStyle adds methods to its superclass, NSParagraphStyle, for changing the values of the subattributes in a paragraph style attribute.

Example:

```
dim n as NSParagraphStyleMBS = NSParagraphStyleMBS.defaultParagraphStyle
MsgBox str(n.alignment) // 4 = natural
```

```
dim m as NSMutableParagraphStyleMBS = n.mutableCopy
m.setAlignment NSParagraphStyleMBS.NSCenterTextAlignment
MsgBox str(m.alignment) // 2 = center
```

Notes: See the NSParagraphStyle and NSAttributedString specifications for more information.

Important A paragraph style object should not be mutated after adding it to an attributed string; doing so can cause your program to crash.

Subclass of the NSParagraphStyleMBS class.

Blog Entries

- [News from the MBS Xojo Plugins Version 22.0](#)
- [MBS Xojo Plugins, version 18.1pr3](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr9](#)

- [MBS Real Studio Plugins, version 12.4pr1](#)

15.1.2 Methods

15.1.3 addTabStop(tabstop as NSTextTabMBS)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Adds tabStop to the receiver.

15.1.4 Constructor

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: The constructor.

15.1.5 removeTabStop(tabstop as NSTextTabMBS)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Removes the first text tab whose location and type are equal to those of tabStop.

15.1.6 setAlignment(alignment as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the alignment of the receiver to alignment.

Example:

```
dim n as NSParagraphStyleMBS = NSParagraphStyleMBS.defaultParagraphStyle
MsgBox str(n.alignment) // 4 = natural
```

```
dim m as NSMutableParagraphStyleMBS = n.mutableCopy
m.setAlignment NSParagraphStyleMBS.NSCenterTextAlignment
MsgBox str(m.alignment) // 2 = center
```

Notes: alignment may be one of:

NSLeftTextAlignment
NSRightTextAlignment

NSCenterTextAlignment
NSJustifiedTextAlignment
NSNaturalTextAlignment

15.1.7 setBaseWritingDirection(writingDirection as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the base writing direction for the receiver.

Notes: It can be NSWritingDirectionNaturalDirection, NSWritingDirectionLeftToRight, or NSWritingDirectionRightToLeft. If you specify NSWritingDirectionNaturalDirection, the receiver resolves the writing direction to either NSWritingDirectionLeftToRight or NSWritingDirectionRightToLeft, depending on the direction for the user's language preference setting.

15.1.8 setDefaultTabInterval(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the default tab interval for the receiver.

Notes: Tabs after the last specified in tabStops are placed at integral multiples of this distance. This value must be nonnegative.

15.1.9 setFirstLineHeadIndent(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the distance in points from the leading margin of a text container to the beginning of the paragraph's first line to value.

Notes: This value must be nonnegative.

15.1.10 setHeaderLevel(level as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Specifies whether the paragraph is to be treated as a header for purposes of HTML generation.

Notes: Should be set to 0 (the default value) if the paragraph is not a header, or from 1 through 6 if the paragraph is to be treated as a header.

15.1.11 setHeadIndent(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the distance in points from the leading margin of a text container to the beginning of lines other than the first to value.

Notes: This value must be nonnegative.

15.1.12 setHyphenationFactor(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Specifies the threshold for hyphenation.

Notes: Valid values lie between 0.0 and 1.0 inclusive. The default value is 0.0. Hyphenation is attempted when the ratio of the text width (as broken without hyphenation) to the width of the line fragment is less than the hyphenation factor. When the paragraph's hyphenation factor is 0.0, the layout manager's hyphenation factor is used instead. When both are 0.0, hyphenation is disabled.

15.1.13 setLineBreakMode(mode as Integer)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the mode used to break lines in a layout container to mode.

Notes: The mode parameter may be one of:

NSLineBreakByWordWrapping

NSLineBreakByCharWrapping

NSLineBreakByClipping

NSLineBreakByTruncatingHead

NSLineBreakByTruncatingTail

NSLineBreakByTruncatingMiddle

See the description of lineBreakMode in the NSParagraphStyle class specification for descriptions of these values.

15.1.14 setLineHeightMultiple(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the line height multiple for the receiver.

Notes: The natural line height of the receiver is multiplied by this factor before being constrained by minimum and maximum line height. This value must be nonnegative.

15.1.15 setLineSpacing(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the space in points added between lines within the paragraph to value.

Example:

```
// works for Labels, TextArea and TextFields

// control is either a textfield or a textview
dim n as NSTextFieldMBS = me.NSTextFieldMBS
dim v as NSTextViewMBS = me.NSTextViewMBS

// get text with attributes
dim a as NSAttributedStringMBS

if n<>Nil then
a = n.attributedStringValue
elseif v<>nil then
a = v.textStorage
end if

// get style
dim p as NSParagraphStyleMBS

try
p = a.attributeAtIndex(a.NSParagraphStyleAttributeName, 0)
catch ex as NSEExceptionMBS
// we have none, so make one
p = new NSParagraphStyleMBS
end try

// modify it
dim m as NSMutableParagraphStyleMBS = p.mutableCopy
m.setLineSpacing 5

// add back to styled text
dim s as NSMutableAttributedStringMBS = a.mutableCopy
s.addAttribute(a.NSParagraphStyleAttributeName, m, new NSRangeMBS(0, s.length))

// and apply to control
if n<>Nil then
n.attributedStringValue = s
elseif v<>nil then
v.textStorage.setAttributedString s
end if
```

Notes: This value must be nonnegative.

15.1.16 `setMaximumLineHeight(value as Double)`

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the maximum height that any line in the paragraph style will occupy, regardless of the font size or size of any attached graphic, to value.

Notes: Glyphs and graphics exceeding this height will overlap neighboring lines; however, a maximum height of 0 implies no line height limit. This value must be nonnegative.

Although this limit applies to the line itself, line spacing adds extra space between adjacent lines.

15.1.17 `setMinimumLineHeight(value as Double)`

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the minimum height that any line in the paragraph style will occupy, regardless of the font size or size of any attached graphic, to value.

Notes: This value must be nonnegative.

15.1.18 `setParagraphSpacing(value as Double)`

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the space added at the end of the paragraph to separate it from the following paragraph to value.

Notes: This value must be nonnegative.

15.1.19 `setParagraphSpacingBefore(value as Double)`

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the distance between the paragraph's top and the beginning of its text content.

Notes: This value must be nonnegative.

15.1.20 `setParagraphStyle(ParagraphStyle as NSParagraphStyleMBS)`

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Replaces the subattributes of the receiver with those in ParagraphStyle.

15.1.21 setTabStops(tabStops() as NSTextTabMBS)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Replaces the tab stops in the receiver with tabStops.

15.1.22 setTailIndent(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Sets the distance in points from the margin of a text container to the end of lines to value.

Notes: If positive, this is the distance from the leading margin (for example, the left margin in left-to-right text). That is, it's the absolute line width. If 0 or negative, it's the distance from the trailing margin—the value is added to the line width.

For example, to create a paragraph style that fits exactly in a 2-inch wide container, set its head indent to 0.0 and its tail indent to 0.0. To create a paragraph style with quarter-inch margins, set its head indent to 0.25 and its tail indent to -0.25.

15.1.23 setTextBlocks(TextBlocks() as NSTextBlockMBS)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the text blocks that contain the paragraph.

15.1.24 setTextLists(TextLists() as NSTextListMBS)

Plugin Version: 18.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the text lists containing the paragraph.

15.1.25 setTighteningFactorForTruncation(value as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Specifies the threshold for using tightening as an alternative to truncation.

Notes: When the line break mode specifies truncation, the text system attempts to tighten intercharacter

spacing as an alternative to truncation, provided that the ratio of the text width to the line fragment width does not exceed $1.0 +$ the value returned by `tighteningFactorForTruncation`. Otherwise the text is truncated at a location determined by the line break mode. The default value is 0.05. This method accepts positive and negative values. Values less than or equal to 0.0 result in not tightening.

15.2 class NSParagraphStyleMBS

15.2.1 class NSParagraphStyleMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: NSParagraphStyle and its subclass NSMutableParagraphStyle encapsulate the paragraph or ruler attributes used by the NSAttributedString classes.

Example:

```
dim n as NSParagraphStyleMBS = NSParagraphStyleMBS.defaultParagraphStyle
MsgBox str(n.alignment) // 4 = natural
```

Notes: Instances of these classes are often referred to as paragraph style objects or, when no confusion will result, paragraph styles.

The mutable subclass of NSParagraphStyle is NSMutableParagraphStyle.

Blog Entries

- [MBS Xojo Plugins, version 22.1pr3](#)
- [News from the MBS Xojo Plugins Version 22.0](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.1](#)
- [MBS Xojo Plugins, version 18.1pr3](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr9](#)
- [MBS Real Studio Plugins, version 12.5pr3](#)
- [MBS Real Studio Plugins, version 12.4pr1](#)

15.2.2 Methods

15.2.3 Constructor

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: The constructor.

15.2.4 copy as NSParagraphStyleMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Creates a copy of the paragraph style.

15.2.5 defaultParagraphStyle as NSParagraphStyleMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the default paragraph style.

Example:

```
dim n as NSParagraphStyleMBS = NSParagraphStyleMBS.defaultParagraphStyle
MsgBox str(n.alignment) // 4 = natural
```

Notes: The default paragraph style has the following default values:

Subattribute	Default Value
Alignment	NSNaturalTextAlignment
Tab stops	12 left-aligned tabs, spaced by 28.0 points
Line break mode	NSLineBreakByWordWrapping
All others	0.0

15.2.6 defaultWritingDirectionForLanguage(languageName as string) as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the default writing direction for the specified language.

Notes: languageName: The language specified in ISO language region format. Can be nil to return a default writing direction derived from the user's defaults database.

15.2.7 mutableCopy as NSMutableParagraphStyleMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Creates a mutable copy of the paragraph style.

15.2.8 NSCenterTextAlignment as Integer

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: One of the text alignment constants for the alignment property.

Notes: Visually centered

15.2.9 NSRightTextAlignment as Integer

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: One of the text alignment constants for the alignment property.

Notes: Visually right aligned

15.2.10 tabStops as NSTextTabMBS()

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the receiver's tab stops.

Notes: The NSTextTab objects, sorted by location, that define the tab stops for the paragraph style.

15.2.11 textBlocks as NSTextBlockMBS()

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The text blocks that contain the paragraph.

15.2.12 textLists as NSTextListMBS()

Plugin Version: 18.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array specifying the text lists containing the paragraph.

15.2.13 Properties**15.2.14 alignment as Integer**

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the text alignment of the receiver.

Example:

```
dim n as NSParagraphStyleMBS = NSParagraphStyleMBS.defaultParagraphStyle
MsgBox str(n.alignment) // 4 = natural
```

```
dim m as NSMutableParagraphStyleMBS = n.mutableCopy
m.setAlignment NSParagraphStyleMBS.NSCenterTextAlignment
MsgBox str(m.alignment) // 2 = center
```

Notes: Natural text alignment is realized as left or right alignment depending on the line sweep direction of the first script contained in the paragraph.
(Read only property)

15.2.15 `baseWritingDirection` as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the base writing direction for the receiver.

Notes: (Read only property)

15.2.16 `defaultTabInterval` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the document-wide default tab interval.

Notes: The default tab interval in points. Tabs after the last specified in `tabStops` are placed at integer multiples of this distance (if positive). Default return value is 0.0.
(Read only property)

15.2.17 `firstLineHeadIndent` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the indentation of the first line of the receiver.

Notes: The distance in points from the leading margin of a text container to the beginning of the paragraph's first line. This value is always nonnegative.
(Read only property)

15.2.18 `firstTabStop` as `NSTextTabMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: The first tab stop entry.

Notes: For debugging.
(Read only property)

15.2.19 firstTextList as NSTextListMBS

Plugin Version: 18.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The first text list entry.

Notes: For debugging.
(Read only property)

15.2.20 Handle as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

15.2.21 headerLevel as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Specifies whether the paragraph is to be treated as a header for purposes of HTML generation.

Notes: Returns 0 (the default value), if the paragraph is not a header, or from 1 through 6 if the paragraph is to be treated as a header.
(Read only property)

15.2.22 headIndent as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the indentation of the receiver's lines other than the first.

Notes: The distance in points from the leading margin of a text container to the beginning of lines other than the first. This value is always nonnegative.
(Read only property)

15.2.23 hyphenationFactor as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the paragraph's threshold for hyphenation.

Notes: A value between 0.0 and 1.0 inclusive. The default value is 0.0.

Hyphenation is attempted when the ratio of the text width (as broken without hyphenation) to the width of the line fragment is less than the hyphenation factor. When the paragraph's hyphenation factor is 0.0, the layout manager's hyphenation factor is used instead. When both are 0.0, hyphenation is disabled.

(Read only property)

15.2.24 `lineBreakMode` as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the mode that should be used to break lines in the receiver.

Notes: The line break mode to be used laying out the paragraph's text.

(Read only property)

15.2.25 `lineHeightMultiple` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the line height multiple.

Notes: The line height multiple. The natural line height of the receiver is multiplied by this factor (if positive) before being constrained by minimum and maximum line height. Default return value is 0.0.

(Read only property)

15.2.26 `lineSpacing` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the space between lines in the receiver (commonly known as leading).

Notes: The space in points added between lines within the paragraph. This value is always nonnegative.

(Read only property)

15.2.27 `maximumLineHeight` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the receiver's maximum line height.

Notes: The maximum height in points that any line in the receiver will occupy, regardless of the font size or size of any attached graphic. This value is always nonnegative. The default value is 0.

Glyphs and graphics exceeding this height will overlap neighboring lines; however, a maximum height of 0 implies no line height limit. Although this limit applies to the line itself, line spacing adds extra space between adjacent lines.

(Read only property)

15.2.28 `minimumLineHeight` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the receiver's minimum height.

Notes: The minimum height in points that any line in the receiver will occupy, regardless of the font size or size of any attached graphic. This value is always nonnegative.

(Read only property)

15.2.29 `paragraphSpacing` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the space after the end of the paragraph.

Notes: The space in points added at the end of the paragraph to separate it from the following paragraph. This value is always nonnegative.

This value is determined by adding the previous paragraph's `paragraphSpacing` and the current paragraph's `paragraphSpacingBefore`.

(Read only property)

15.2.30 `paragraphSpacingBefore` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the distance between the paragraph's top and the beginning of its text content.

Notes: The distance in points between the paragraph's top and the beginning of its text content. Default return value is 0.0.

(Read only property)

15.2.31 `tailIndent` as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the trailing indentation of the receiver.

Notes: The distance in points from the margin of a text container to the end of lines.

If positive, this value is the distance from the leading margin (for example, the left margin in left-to-right

text). If 0 or negative, it's the distance from the trailing margin.

For example, a paragraph style designed to fit exactly in a 2-inch wide container has a head indent of 0.0 and a tail indent of 0.0. One designed to fit with a quarter-inch margin has a head indent of 0.25 and a tail indent of -0.25.

(Read only property)

15.2.32 tighteningFactorForTruncation as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the threshold for using tightening as an alternative to truncation.

Notes: The tightening threshold value. The default value is 0.05.

When the line break mode specifies truncation, the text system attempts to tighten intercharacter spacing as an alternative to truncation, provided that the ratio of the text width to the line fragment width does not exceed 1.0 + the tightening factor returned by this method. Otherwise the text is truncated at a location determined by the line break mode.

(Read only property)

15.2.33 Constants

Alignment Modes

Constant	Value	Description
<code>NSJustifiedTextAlignment</code>	3	Fully-justified. The last line in a paragraph is natural-aligned.
<code>NSLeftTextAlignment</code>	0	Visually left aligned
<code>NSNaturalTextAlignment</code>	4	Indicates the default alignment for script.

Line Break Modes

Constant	Value	Description
<code>NSLineBreakByCharWrapping</code>	1	Wrapping occurs before the first character that doesn't fit.
<code>NSLineBreakByClipping</code>	2	Lines are simply not drawn past the edge of the text container.
<code>NSLineBreakByTruncatingHead</code>	3	Each line is displayed so that the end fits in the container and the missing text is indicated by some kind of ellipsis glyph.
<code>NSLineBreakByTruncatingMiddle</code>	5	Each line is displayed so that the beginning and end fit in the container and the missing text is indicated by some kind of ellipsis glyph in the middle.
<code>NSLineBreakByTruncatingTail</code>	4	Each line is displayed so that the beginning fits in the container and the missing text is indicated by some kind of ellipsis glyph.
<code>NSLineBreakByWordWrapping</code>	0	Wrapping occurs at word boundaries, unless the word itself doesn't fit on single line.

Writing Direction

Constant	Value	Description
NSWritingDirectionLeftToRight	0	The writing direction is left to right. Available in Mac OS X v10.2 and later.
NSWritingDirectionNatural	-1	The writing direction is determined using the Unicode Bidi Algorithm rules P2 and P3. Default. Available in Mac OS X v10.4 and later.
NSWritingDirectionRightToLeft	1	The writing direction is right to left. Available in Mac OS X v10.2 and later.

15.3 class NSTextContainerMBS

15.3.1 class NSTextContainerMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The NSTextContainer class defines a region where text is laid out.

Notes: An NSLayoutManager uses NSTextContainer to determine where to break lines, lay out portions of text, and so on. NSTextContainer defines rectangular regions, but you can create subclasses that define regions of other shapes, such as circular regions, regions with holes in them, or regions that flow alongside graphics.

Blog Entries

- [MBS Xojo Plugins, version 18.6pr1](#)
- [Line Wrap for Textarea in Xojo Mac applications](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr1](#)
- [MBS Real Studio Plugins, version 12.4pr4](#)

15.3.2 Methods

15.3.3 Constructor

Plugin Version: 19.0, Platform: macOS, Targets: Desktop only.

Function: Initializes a text container with no bounds.

Notes: Returns the newly initialized text container.

See also:

- 15.3.4 Constructor(size as NSSizeMBS)

908

15.3.4 Constructor(size as NSSizeMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Initializes a text container with a specified bounding rectangle.

Notes: size: The size of the text container's bounding rectangle.

Returns the newly initialized text container.

The new text container must be added to an NSLayoutManager object before it can be used. The text container must also have an NSTextView object set for text to be displayed. This method is the designated initializer for the NSTextContainer class.

See also:

- 15.3.3 Constructor

15.3.5 containsPoint(p as NSPointMBS) as boolean

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Overridden by subclasses to return whether a point lies within the receiver's region or on the region's edge—not simply within its bounding rectangle.

Notes: True if aPoint lies within the receiver's region or on the region's edge—not simply within its bounding rectangle - false otherwise.

For example, if the receiver defines a donut shape and aPoint lies in the hole, this method returns false. This method can be used for hit testing of mouse events.

The default NSTextContainer implementation merely checks that aPoint lies within its bounding rectangle.

15.3.6 isSimpleRectangularTextContainer as boolean

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Overridden by subclasses to return whether the receiver's region is a rectangle with no holes or gaps and whose edges are parallel to the text view's coordinate system axes.

Notes: True if the receiver's region is a rectangle with no holes or gaps and whose edges are parallel to the text view's coordinate system axes, false otherwise.

A text container whose shape changes can return true if its region is currently a simple rectangle, but when its shape does change it must send textContainerChangedGeometry to its layout manager so the layout can be recalculated.

The default NSTextContainer implementation of this method returns true.

15.3.7 replaceLayoutManager(l as NSLayoutManagerMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Replaces the layout manager for the group of text system objects containing the receiver.

Notes: All text containers and text views sharing the original layout manager share the new layout manager. This method makes all the adjustments necessary to keep these relationships intact, unlike setLayoutManager.

15.3.8 Properties

15.3.9 Handle as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.

Notes: (Read and Write property)

15.3.10 containerSize as NSSizeMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The size of the receiver's bounding rectangle.

Notes: size: The new size of the text container's bounding rectangle.

This method also sends textContainerChangedGeometry to the text container's layout manager.

(Read and Write computed property)

15.3.11 heightTracksTextView as boolean

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Controls whether the receiver adjusts the height of its bounding rectangle when its text view is resized.

Notes: value: True if the receiver should follow changes to the height of its text view, false otherwise.

(Read and Write computed property)

15.3.12 layoutManager as NSLayoutManagerMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The layout manager.

Notes: This method is invoked automatically when you add a text container to a layout manager; you should never need to invoke it directly, but might want to override it. If you want to replace the layout manager for an established group of text system objects, use replaceLayoutManager.

(Read and Write computed property)

15.3.13 lineFragmentPadding as Double

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The amount by which text is inset within line fragment rectangles.

Notes: Value: The amount by which text is inset within line fragment rectangles, in points.

This method also sends `textContainerChangedGeometry` to the text container's layout manager.

Line fragment padding is not designed to express text margins. Instead, use the `NSTextView` method `setTextContainerInset`, paragraph margin attributes, or the position of the text view within a superview. (Read and Write computed property)

15.3.14 `textView` as `NSTextViewMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: The text view.

Notes: This method sends `setTextContainer` to a `TextView` to complete the association of the text container and text view.

Because you usually specify a text container when you create a text view, you should rarely need to invoke this method. A text container doesn't need a text view to calculate line fragment rectangles, but must have one to display text.

You can use this method to disconnect a text view from a group of text system objects by sending this message to its text container and passing `nil` as a `TextView`. (Read and Write computed property)

15.3.15 `widthTracksTextView` as `boolean`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Controls whether the receiver adjusts the width of its bounding rectangle when its text view is resized.

Notes: Returns true if the receiver adjusts the width of its bounding rectangle when its text view is resized, false otherwise.

If the receiver does track the text view width, its width is adjusted to the width of the text view minus twice the inset width (as given by `NSTextView`'s `textContainerInset` method). (Read and Write computed property)

15.3.16 Constants

Line Movement Constants

Constant	Value	Description
<code>NSLineDoesntMove</code>	0	Line has no movement.
<code>NSLineMovesDown</code>	3	Lines move from top to bottom.
<code>NSLineMovesLeft</code>	1	Lines move from right to left.
<code>NSLineMovesRight</code>	2	Lines move from left to right.
<code>NSLineMovesUp</code>	4	Lines move from bottom to top.

Line Sweep Direction Constants

Constant	Value	Description
<code>NSLineSweepDown</code>	2	Characters move from top to bottom.
<code>NSLineSweepLeft</code>	0	Characters move from right to left.
<code>NSLineSweepRight</code>	1	Characters move from left to right.
<code>NSLineSweepUp</code>	3	Characters move from bottom to top.

15.4 class NSTextMBS

15.4.1 class NSTextMBS

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The Cocoa class for text controls.

Notes: You may prefer to use the NSTextViewMBS class which is a subclass from NSTextMBS.

You can embed this view in a CustomNSViewMBS to get more events for mouse and keyboard.
Subclass of the NSViewMBS class.

Blog Entries

- [RTF functions in MBS Plugins](#)
- [MBS Xojo Plugins, version 22.1pr3](#)
- [MBS Xojo / Real Studio Plugins, version 16.5pr6](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr7](#)

15.4.2 Methods

15.4.3 alignCenter

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method applies center alignment to selected paragraphs (or all text if the receiver is a plain text object).

15.4.4 alignLeft

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method applies left alignment to selected paragraphs (or all text if the receiver is a plain text object).

15.4.5 alignRight

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method applies right alignment to selected paragraphs (or all text if the receiver is a plain text object).

15.4.6 changeFont

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method changes the font of the selection for a rich text object, or of all text for a plain text object.

Notes: If the receiver doesn't use the Font panel, this method does nothing.

This method changes the font by sending a `convertFont` message to the shared `NSFontManager` and applying each `NSFont` returned to the appropriate text. See the `NSFontManager` class specification for more information on font conversion.

15.4.7 checkSpelling

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: This action method searches for a misspelled word in the receiver's text.

Notes: The search starts at the end of the selection and continues until it reaches a word suspected of being misspelled or the end of the text. If a word isn't recognized by the spelling server, a `showGuessPanel` message then opens the Guess panel and allows the user to make a correction or add the word to the local dictionary.

15.4.8 Constructor

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new text with size 100/100 and position 0/0

Example:

```
dim t as new NSTextMBS
```

Notes: On success the handle property is not zero.

See also:

- 15.4.9 Constructor(Handle as Integer) 914
- 15.4.10 Constructor(left as Double, top as Double, width as Double, height as Double) 915

15.4.9 Constructor(Handle as Integer)

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSText handle.

Example:

```
dim t as new NSTextMBS(0, 0, 100, 100)
dim v as new NSTextMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSText and the plugin retains this handle.

See also:

- 15.4.8 Constructor 914
- 15.4.10 Constructor(left as Double, top as Double, width as Double, height as Double) 915

15.4.10 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Creates a new text with the given size and position.

Example:

```
dim x as new NSTextMBS(0, 0, 100, 100)
```

Notes: On success the handle property is not zero.

See also:

- 15.4.8 Constructor 914
- 15.4.9 Constructor(Handle as Integer) 914

15.4.11 copy

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method copies the selected text onto the general pasteboard, in as many formats as the receiver supports.

15.4.12 copyFont

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method copies the font information for the first character of the selection (or for the insertion point) onto the font pasteboard, as `NSFontPboardType`.

15.4.13 `copyRuler`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method copies the paragraph style information for first selected paragraph onto the ruler pasteboard, as `NSRulerPboardType`, and expands the selection to paragraph boundaries.

15.4.14 `cut`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method deletes the selected text and places it onto the general pasteboard, in as many formats as the receiver supports.

15.4.15 `delete`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method deletes the selected text.

15.4.16 `isRulerVisible` as `boolean`

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver's enclosing scroll view shows its ruler.

15.4.17 `maxSizeHeight` as `Double`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's maximum height.

15.4.18 `maxSizeWidth` as `Double`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's maximum width.

15.4.19 minSizeHeight as Double

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's minimum height.

15.4.20 minSizeWidth as Double

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's maximum width.

15.4.21 NSCenterTextAlignment as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: One of the text alignment constants for the alignment property.

Notes: Visually centered

15.4.22 NSRightTextAlignment as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: One of the text alignment constants for the alignment property.

Notes: Visually right aligned

15.4.23 paste

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method pastes text from the general pasteboard at the insertion point or over the selection.

15.4.24 pasteFont

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method pastes font information from the font pasteboard onto the selected text or insertion point of a rich text object, or over all text of a plain text object.

15.4.25 pasteRuler

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method pastes paragraph style information from the ruler pasteboard onto the selected paragraphs of a rich text object.

15.4.26 readRTFDFromFile(file as folderitem) as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Attempts to read the RTFD file, returning true if successful and false if not.

Notes: file should be the path for an .rtf file or an .rtfd file wrapper, not for the RTF file within an .rtfd file wrapper.

15.4.27 replaceCharactersInRangeWithRTF(start as Integer, length as Integer, rtfData as MemoryBlock)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Replaces the characters in the given range with RTF text interpreted from the given RTF data.

Notes: This method applies only to rich text objects.

This method does not include undo support by default. Clients must invoke `shouldChangeTextInRange` to include this method in an undoable action.

This method is designed for transferring text from out-of-process sources such as the pasteboard. In most cases, programmatic modification of the text is best done by operating on the text storage directly, using the general methods of `NSMutableAttributedString`.

15.4.28 replaceCharactersInRangeWithRTFD(start as Integer, length as Integer, rtfData as MemoryBlock)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Replaces the characters in the given range with RTFD text interpreted from the given RTFD data.

Notes: This method applies only to rich text objects.

This method does not include undo support by default. Clients must invoke `shouldChangeTextInRange` to include this method in an undoable action.

This method is designed for transferring text from out-of-process sources such as the pasteboard. In most cases, programmatic modification of the text is best done by operating on the text storage directly, using the general methods of `NSMutableAttributedString`.

15.4.29 `replaceCharactersInRangeWithString(start as Integer, length as Integer, text as string)`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Replaces the characters in the given range with those in the given string.

Notes: For a rich text object, the text is assigned the formatting attributes of the first character of the text it replaces, or of the character immediately before `aRange` if the range's length is 0. If the range's location is 0, the formatting attributes of the first character in the receiver are used.

This method does not include undo support by default. Clients must invoke `shouldChangeTextInRange` to include this method in an undoable action.

In most cases, programmatic modification of the text is best done by operating on the text storage directly, using the general methods of `NSMutableAttributedString`.

15.4.30 `RTFDFromRange(start as Integer, length as Integer) as MemoryBlock`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a string that contains an RTFD stream corresponding to the characters and attributes within `aRange`.

Notes: Raises an `NSRangeException` if any part of `aRange` lies beyond the end of the receiver's characters.

When writing data to the pasteboard, you can use the `memoryblock` object as the first argument to `NSPasteboard's setDataForType` method, with a second argument of `NSRTFDPboardType`.

15.4.31 `RTFFFromRange(start as Integer, length as Integer) as MemoryBlock`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: returns a string object that contains an RTF stream corresponding to the characters and attributes within aRange, omitting any attachment characters and attributes.

Notes: Raises an NSRangeException if any part of aRange lies beyond the end of the receiver's characters.

When writing data to the pasteboard, you can use the memoryblock as the first argument to NSPasteboard's setDataForType method, with a second argument of NSRTFPboardType.

15.4.32 scrollRangeToVisible(start as Integer, length as Integer)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Scrolls the receiver in its enclosing scroll view so the first characters of aRange are visible.

15.4.33 selectAll

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method selects all of the receiver's text.

15.4.34 setFontForRange(font as NSFontMBS, start as Integer, length as Integer)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Sets the font of characters within the given range to font.

Notes: This method applies only to a rich text object.

This method does not include undo support by default. Clients must invoke shouldChangeTextInRanges to include this method in an undoable action.

15.4.35 setMaxSize(width as Double, height as Double)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Sets the receiver's maximum size.

15.4.36 setMinSize(width as Double, height as Double)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Sets the receiver's minimum size.

15.4.37 setTextColorForRange(colorValue as NSColorMBS, start as Integer, length as Integer)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Sets the text color of characters within the given range to colorValue.

Notes: Removes the text color attribute if colorValue is nil. This method applies only to rich text objects.

This method does not include undo support by default. Clients must invoke shouldChangeTextInRange to include this method in an undoable action.

15.4.38 showGuessPanel

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: This action method opens the Spelling panel, allowing the user to make a correction during spell checking.

15.4.39 sizeToFit

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Resizes the receiver to fit its text.

Notes: The text view will not be sized any smaller than its minimum size, however.

15.4.40 subscript

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method applies a subscript attribute to selected text (or all text if the receiver is a plain text object), lowering its baseline offset by a predefined amount.

15.4.41 superscript

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method applies a superscript attribute to selected text (or all text if the receiver is a plain text object), raising its baseline offset by a predefined amount.

15.4.42 textLength as Integer

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Returns the length of the text in unicode chars.

15.4.43 toggleRuler

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method shows or hides the ruler, if the receiver is enclosed in a scroll view.

15.4.44 underline

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Adds the underline attribute to the selected text attributes if absent; removes the attribute if present.

Notes: If there is a selection and the first character of the selected range has any form of underline on it, or if there is no selection and the typing attributes have any form of underline, then underline is removed; otherwise a single simple underline is added.

Operates on the selected range if the receiver contains rich text. For plain text the range is the entire contents of the receiver.

15.4.45 unscript

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: This action method removes any superscripting or subscripting from selected text (or all text if the receiver is a plain text object).

15.4.46 writeRTFDToFile(file as folderitem, atomically as boolean) as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Writes the receiver's text as RTF with attachments to a file or directory at file.

Notes: Returns true on success and false on failure. If atomically is true, attempts to write the file safely so that an existing file at path is not overwritten, nor does a new file at path actually get created, unless the write is successful.

15.4.47 Properties**15.4.48 alignment as Integer**

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The text alignment.

Notes: (Read and Write computed property)

15.4.49 backgroundColor as NSColorMBS

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The background color.

Notes: (Read and Write computed property)

15.4.50 baseWritingDirection as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The initial writing direction used to determine the actual writing direction for text.

Notes: The Text system uses this value as a hint for calculating the actual direction for displaying Unicode characters. You should not need to call this method directly. If no writing direction is set, returns `NSWritingDirectionNatural`.

Available in Mac OS X v10.4 and later.
(Read and Write computed property)

15.4.51 drawsBackground as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the scrollview draws its background.

Notes: (Read and Write computed property)

15.4.52 Enabled as boolean

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

Function: Enables/disables the control.

Notes: If enabled, text is in default text color.

if disabled, you can't edit or select and the text color is gray.

(Read and Write computed property)

15.4.53 font as NSFontMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The font of the first character in the receiver's text, or of the insertion point if there's no text.

Notes: (Read and Write computed property)

15.4.54 importsGraphics as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver allows the user to import files by dragging.

Notes: (Read and Write computed property)

15.4.55 isEditable as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver allows the user to edit text, false if it doesn't.

Notes: You can change the receiver's text programmatically regardless of this setting.

If the receiver is editable, it's also selectable.

(Read and Write computed property)

15.4.56 isFieldEditor as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver interprets Tab, Shift-Tab, and Return (Enter) as cues to end editing and possibly to change the first responder.

Notes: True if the receiver interprets Tab, Shift-Tab, and Return (Enter) as cues to end editing and possibly to change the first responder; false if it accepts them as text input.

See the `NSWindow` class specification for more information on field editors. By default, `NSText` objects don't behave as field editors.

(Read and Write computed property)

15.4.57 `isHorizontallyResizable` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: True if the receiver automatically changes its width to accommodate the width of its text, false if it doesn't.

Notes: By default, an `NSText` object is not horizontally resizable.

(Read and Write computed property)

15.4.58 `isRichText` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the `NSText` allows the user to apply attributes to specific ranges of the text.

Notes: (Read and Write computed property)

15.4.59 `isSelectable` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the text allows the user to select text, false if it doesn't.

Notes: (Read and Write computed property)

15.4.60 `isVerticallyResizable` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: True if the receiver automatically changes its height to accommodate the height of its text, false if it doesn't.

Notes: By default, an `NSText` object is vertically resizable.

(Read and Write computed property)

15.4.61 `selectedRange` as `NSRangeMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The range of selected characters.

Example:

```
dim t as NSTextViewMBS = TextArea1.NSTextViewMBS
dim s as NSTextStorageMBS = t.textStorage

const NSUnderlineStyleSingle = 1

dim r as NSRangeMBS = t.selectedRange
s.addAttribute NSAttributedStringMBS.NSStrikethroughStyleAttributeName, NSUnderlineStyleSingle, r
```

Notes: (Read and Write computed property)

15.4.62 `text` as string

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The characters of the text.

Notes: For performance reasons, this method returns the current backing store of the text object. If you want to maintain a snapshot of this as you manipulate the text storage, you should make a copy of the appropriate substring.

(Read and Write computed property)

15.4.63 `textColor` as `NSColorMBS`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The text color.

Notes: (Read and Write computed property)

15.4.64 `usesFontPanel` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver uses the Font panel.

Notes: (Read and Write computed property)

15.4.65 Events

15.4.66 textDidBeginEditing

Plugin Version: 8.4, Platform: macOS, Targets: .

Function: Informs you that the text object has begun editing (that the user has begun changing it).

15.4.67 textDidChange

Plugin Version: 8.4, Platform: macOS, Targets: .

Function: Informs you that the text object has changed its characters or formatting attributes.

15.4.68 textDidEndEditing

Plugin Version: 8.4, Platform: macOS, Targets: .

Function: Informs you that the text object has finished editing (that it has resigned first responder status).

15.4.69 textShouldBeginEditing as boolean

Plugin Version: 8.4, Platform: macOS, Targets: .

Function: Invoked when a text object begins to change its text, this method requests permission to begin editing.

Notes: If the delegate returns false, the text object proceeds to make changes. If the delegate returns true, the text object abandons the editing operation. This method is also invoked when the user drags and drops a file onto the text object.

15.4.70 textShouldEndEditing as boolean

Plugin Version: 8.4, Platform: macOS, Targets: .

Function: Invoked from a text object's implementation of resignFirstResponder, this method requests permission to end editing.

Notes: If the delegate returns false, the text object proceeds to finish editing and resign first responder status. If the delegate returns true, the text object selects all of its text and remains the first responder.

15.4.71 Constants

Constants

Constant	Value	Description
NSBackspaceCharacter	8	One of the constants to specify commonly used Unicode characters.
NSBackTabCharacter	&h19	One of the constants to specify commonly used Unicode characters.
NSBacktabTextMovement	&h12	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The Backtab (Shift-Tab) key was pressed.
NSCancelTextMovement	&h17	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The user cancelled the completion. Available in Mac OS X v10.3 and later.
NSCarriageReturnCharacter	13	One of the constants to specify commonly used Unicode characters.
NSDeleteCharacter	&h7F	One of the constants to specify commonly used Unicode characters.
NSDownTextMovement	&h16	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The down arrow key was pressed.
NSEnterCharacter	3	One of the constants to specify commonly used Unicode characters.
NSFormFeedCharacter	12	One of the constants to specify commonly used Unicode characters.
NSIllegalTextMovement	0	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" Currently unused.
NSLeftTextMovement	&h13	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The left arrow key was pressed.
NSLineSeparatorCharacter	&h2028	One of the constants to specify commonly used Unicode characters.
NSNewlineCharacter	10	One of the constants to specify commonly used Unicode characters.
NSOtherTextMovement	0	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The user performed some undefined action. Available in Mac OS X v10.3 and later.
NSParagraphSeparatorCharacter	&h2029	One of the constants to specify commonly used Unicode characters.
NSReturnTextMovement	&h10	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The Return key was pressed.
NSRightTextMovement	&h14	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The right arrow key was pressed.
NSTabCharacter	9	One of the constants to specify commonly used Unicode characters.
NSTabTextMovement	&h11	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The Tab key was pressed.
NSTextWritingDirectionEmbedding	0	An additional constant to specify the writing direction. Direction is embedded. Available in Mac OS X v10.6 and later.
NSTextWritingDirectionOverride	1	An additional constant to specify the writing direction. Direction override Available in Mac OS X v10.6 and later.
NSUpTextMovement	&h15	One of the constants used to specify the reason for a change of editing among text fields, in essence answering the question "why am I leaving this field?" The up arrow key was pressed.

Alignment Modes

Constant	Value	Description
<code>NSJustifiedTextAlignment</code>	3	Fully-justified. The last line in a paragraph is natural-aligned.
<code>NSLeftTextAlignment</code>	0	Visually left aligned
<code>NSNaturalTextAlignment</code>	4	Indicates the default alignment for script.

Writing Direction

Constant	Value	Description
<code>NSWritingDirectionLeftToRight</code>	0	The writing direction is left to right. Available in Mac OS X v10.2 and later.
<code>NSWritingDirectionNatural</code>	-1	The writing direction is determined using the Unicode Bidi Algorithm rules P2 and P3. Default. Available in Mac OS X v10.4 and later.
<code>NSWritingDirectionRightToLeft</code>	1	The writing direction is right to left. Available in Mac OS X v10.2 and later.

15.5 class NSTextStorageMBS

15.5.1 class NSTextStorageMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: The class for the text storage of a text view.

Notes: NSTextStorage is a semiconcrete subclass of NSMutableAttributedString that manages a set of client NSLayoutManager objects, notifying them of any changes to its characters or attributes so that they can relay and redisplay the text as needed. NSTextStorage defines the fundamental storage mechanism of the Application Kit's extended text-handling system.

NSTextStorage also defines a set of methods, listed under "Getting and setting scriptable properties" in the Method Types section, useful for getting and setting scriptable properties of NSTextStorage objects. Unless you are dealing with scriptability, you do not normally need to invoke these methods directly. In particular, using the characters, words or paragraphs methods or their corresponding setter methods is an inefficient way to manipulate the text storage, since these methods create and return many objects. Instead, use the text access methods defined by NSMutableAttributedString, NSAttributedString, NSMutableString, and NSString to perform character-level manipulation.

Subclass of the NSMutableAttributedStringMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 12.2pr4](#)

15.5.2 Methods

15.5.3 addLayoutManager(l as NSLayoutManagerMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Adds a layout manager to the receiver's set of layout managers.

Notes: l: The layout manager to add.

15.5.4 changeInLength as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the difference between the current length of the edited range and its length before editing began.

Notes: Returns the difference between the current length of the edited range and its length before editing began. That is, before the receiver was sent the first beginEditing message or a single edited:range:changeInLength: message.

This difference is accumulated with each invocation of edited:range:changeInLength:, until a final message

processes the changes.

The receiver's delegate and layout managers can use this information to determine the nature of edits in their respective notification methods.

15.5.5 Constructor

Plugin Version: 12.2, Platform: macOS, Targets: Desktop only.

Function: The constructor to create a new empty text storage object.

15.5.6 `editedMask` as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the kinds of edits pending for the receiver.

Notes: Returns a mask describing the kinds of edits pending for the receiver.

15.5.7 `editedRange` as NSRangeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: returns the range of the receiver to which pending changes have been made, whether of characters or of attributes.

Notes: The receiver's delegate and layout managers can use this information to determine the nature of edits in their respective notification methods.

15.5.8 `ensureAttributesAreFixedInRange`(Range as NSRangeMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Ensures that attributes are fixed in the given range.

Notes: range: The range of characters whose attributes might be examined.

An NSTextStorage object using lazy attribute fixing is required to call this method before accessing any attributes within range. This method gives attribute fixing a chance to occur if necessary. NSTextStorage subclasses wishing to support laziness must call this method from all attribute accessors they implement.

15.5.9 fixesAttributesLazily as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns whether the receiver fixes attributes lazily.

Notes: By default, custom NSTextStorage subclasses are not lazy, but the provided concrete subclass is lazy by default.

15.5.10 invalidateAttributesInRange(Range as NSRangeMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Invalidates attributes in the specified range.

Notes: range: The range of characters whose attributes should be invalidated.

Called from processEditing to invalidate attributes when the text storage changes. If the receiver is not lazy, this method simply calls fixAttributesInRange. If lazy attribute fixing is in effect, this method instead records the range needing fixing.

15.5.11 processEditing

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Cleans up changes made to the receiver and notifies its delegate and layout managers of changes.

15.5.12 removeLayoutManager(l as NSLayoutManagerMBS)

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Removes a layout manager from the receiver's set of layout managers.

Notes: l: The layout manager to remove.

15.5.13 Constants

Constants

Constant	Value	Description
<code>NSTextStorageEditedAttributes</code>	1	The constants for text messages. Attributes were added, removed, or changed.
<code>NSTextStorageEditedCharacters</code>	2	The constants for text messages. Characters were added, removed, or replaced.

15.6 class NSTextTabMBS

15.6.1 class NSTextTabMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: An NSTextTab object represents a tab in an NSParagraphStyle object, storing an alignment type and location.

Notes: NSTextTab objects are most frequently used with the Application Kit's text system and with NSRulerView and NSRulerMarker objects. See the appropriate class specifications for more information on these uses.

The text system supports four alignment types: left, center, right, and decimal (based on the decimal separator character of the locale in effect). These alignment types are absolute, not based on the line sweep direction of text. For example, tabbed text is always positioned to the left of a right-aligned tab, whether the line sweep direction is left to right or right to left. A tab's location, on the other hand, is relative to the back margin. A tab set at 1.5", for example, is at 1.5" from the right in right to left text.

Blog Entries

- [MBS Real Studio Plugins, version 12.4pr1](#)

15.6.2 Methods

15.6.3 Constructor

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: The default constructor.

See also:

- 15.6.4 Constructor(alignment as Integer, location as Double, options as dictionary) 935
- 15.6.5 Constructor(type as Integer, location as Double) 936

15.6.4 Constructor(alignment as Integer, location as Double, options as dictionary)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Initializes a text tab with the text alignment, location, and options.

Notes: The text alignment is used to determine the position of text inside the tab column. See NSTextTabType for a mapping between alignments and tab stop types.

See also:

- 15.6.3 Constructor 935

- 15.6.5 Constructor(type as Integer, location as Double) 936

15.6.5 Constructor(type as Integer, location as Double)

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Initializes a newly allocated NSTextTab with an alignment of type at location on the paragraph.

Notes: The location is relative to the back margin, based on the line sweep direction of the paragraph. type can be any of the values described in NSTextTabType.

See also:

- 15.6.3 Constructor 935
- 15.6.4 Constructor(alignment as Integer, location as Double, options as dictionary) 935

15.6.6 copy as NSTextTabMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Creates a copy of the text tab.

15.6.7 Properties

15.6.8 alignment as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the text alignment of the receiver.

Notes: (Read only property)

15.6.9 Handle as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

15.6.10 location as Double

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the receiver's ruler location relative to the back margin.

Notes: (Read only property)

15.6.11 options as Dictionary

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the dictionary of attributes associated with the receiver.

Notes: (Read only property)

15.6.12 tabStopType as Integer

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Returns the receiver's tab stop type.

Notes: (Read only property)

15.6.13 Constants

Tab Stop Types

Constant	Value	Description
NSCenterTabStopType	2	A center-aligned tab stop.
NSDecimalTabStopType	3	Aligns columns of numbers by the decimal point.
NSLeftTabStopType	0	A left-aligned tab stop.
NSRightTabStopType	1	A right-aligned tab stop.

Chapter 16

Controls

16.1 class Control

16.1.1 class Control

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

Function: The built in Control class in Xojo.

16.1.2 Methods

16.1.3 NSControlMBS as NSControlMBS

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Creates a NSControlMBS object for the given control.

Example:

```
BevelButton1.NSControlMBS.StringValue = "Hello"
```

Notes: This way you can manipulate Cocoa controls directly.

16.2 class DesktopControl

16.2.1 class DesktopControl

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

Function: The built in Control class in Xojo.

16.2.2 Methods

16.2.3 NSControlMBS as NSControlMBS

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

Function: Creates a NSControlMBS object for the given control.

Example:

```
BevelButton1.NSControlMBS.StringValue = "Hello"
```

Notes: This way you can manipulate Cocoa controls directly.

16.3 class DesktopLabel

16.3.1 class DesktopLabel

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

Function: An extension of Xojo's internal control.

16.3.2 Methods

16.3.3 NSTextFieldMBS as NSTextFieldMBS

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

Function: Creates a NSTextFieldMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

16.4 class DesktopTextField

16.4.1 class DesktopTextField

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

Function: The built in TextField class in Xojo.

16.4.2 Methods

16.4.3 NSTextFieldMBS as NSTextFieldMBS

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

Function: Creates a NSTextFieldMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

Seems like Xojo 2011r1 uses a NSTextField, so this method should return an object on Cocoa targets.

16.4.4 NSTextViewMBS as NSTextViewMBS

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

Function: Creates a NSTextViewMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

Seems like Xojo 2011r1 uses a NSTextField, so this method should return nil on Cocoa targets. Please use NSTextFieldMBS method in this case.

16.5 class Label

16.5.1 class Label

Plugin Version: 13.0, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: An extension of Xojo's internal control.

16.5.2 Methods

16.5.3 NSTextFieldMBS as NSTextFieldMBS

Plugin Version: 13.3, Platform: macOS, Targets: Desktop only.

Function: Creates a NSTextFieldMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

16.6 class `TabPanel`

16.6.1 class `TabPanel`

Platform: macOS, Targets: Desktop only.

Function: Extends the `TabPanel` control inside Xojo.

16.6.2 Methods

16.6.3 `NSTabViewMBS` as `NSTabViewMBS`

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a `NSTabViewMBS` object for the given control.

Example:

```
MsgBox TabPanel1.NSTabViewMBS.className
```

Notes: This way you can manipulate Cocoa controls directly.

16.7 class TextField

16.7.1 class TextField

Plugin Version: 10.0, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The built in TextField class in Xojo.

16.7.2 Methods

16.7.3 NSTextFieldMBS as NSTextFieldMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Creates a NSTextFieldMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

Seems like Xojo 2011r1 uses a NSTextField, so this method should return an object on Cocoa targets.

16.7.4 NSTextViewMBS as NSTextViewMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Creates a NSTextViewMBS object for the given control.

Notes: This way you can manipulate Cocoa controls directly.

Seems like Xojo 2011r1 uses a NSTextField, so this method should return nil on Cocoa targets. Please use NSTextFieldMBS method in this case.

Chapter 17

Drag & Drop

17.1 class DragItem

17.1.1 class DragItem

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

Function: The dragitem class from Xojo framework.

17.1.2 Methods

17.1.3 NSDraggingInfoMBS as NSDraggingInfoMBS

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

Function: Provides the NSDraggingInfo object for the DragItem.

17.2 class NSDraggingImageComponentMBS

17.2.1 class NSDraggingImageComponentMBS

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

Function: The NSDraggingImageComponent class represents a single object in a dragging item.

Notes: An array of NSDraggingImageComponent instances are composited together to create the dragging image for an NSDraggingItem. NSDraggingImageComponent instances can simply be considered as named images with a location used by an NSDraggingItem instance.

Blog Entries

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

17.2.2 Methods

17.2.3 Constructor(key as string)

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

Function: Initializes a dragging image component with the specified key.

Notes: key: The key.

Available in OS X v10.7 and later.

17.2.4 draggingImageComponentWithKey(key as string) as NSDraggingImageComponentMBS

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

Function: Creates and returns a dragging image component with the specified key.

Notes: Available in OS X v10.7 and later.

17.2.5 NSDraggingImageComponentIconKey as string

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

Function: One of the keys for Constructor.

Notes: Key with a corresponding value that is an image of the item being dragged.

Available in OS X v10.7 and later.

17.2.6 NSDraggingImageComponentLabelKey as string

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

Function: One of the keys for Constructor.

Notes: Key with a corresponding value that represents a textual label associate with the item, for example, a file name.

Available in OS X v10.7 and later.

17.2.7 Properties

17.2.8 Handle as Integer

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

Function: The internal object handle.

Notes: (Read and Write property)

17.2.9 contents as Variant

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

Function: An object providing the image contents of the component.

Notes: Typically you set an NSImage instance as content.

Available in OS X v10.7 and later.

(Read and Write computed property)

17.2.10 frame as NSRectMBS

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

Function: The coordinate space is the bounds of the parent dragging item.

Notes: The frame is { { 0,0 } , { draggingFrame.size.width, draggingFrame.size.height } } .

The coordinate space is the bounds of the parent NSDraggingItem instance's draggingFrame.

Available in OS X v10.7 and later.

(Read and Write computed property)

17.2.11 key as string

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

Function: The unique name of this image component instance.

Notes: The key must be unique for each component in an NSDraggingItem instance. You can create your own named components, however the keys described in NSDragImage Component Keys have special meanings.

When an NSDraggingItem instances imageComponents are changed by one of the enumerateDraggingItemsWithOptions methods the image associated with this key is morphed into the new image component's image associated with the same key.

Available in OS X v10.7 and later.
(Read and Write computed property)

17.3 class NSDraggingInfoMBS

17.3.1 class NSDraggingInfoMBS

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

Function: The NSDraggingInfo protocol declares methods that supply information about a dragging session.
Notes: NSDraggingInfo protocol methods are designed to be invoked from within a class's implementation of NSDraggingDestination protocol methods. The Application Kit automatically passes an object that conforms to the NSDraggingInfo protocol as the argument to each of the methods defined by NSDraggingDestination. NSDraggingInfo messages should be sent to this object; you never need to create a class that implements the NSDraggingInfo protocol.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 14.3pr7](#)
- [MBS Real Studio Plugins, version 13.1pr9](#)

17.3.2 Methods

17.3.3 Constructor

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

Function: The private constructor.

See also:

- [17.3.4 Constructor\(Handle as Integer\)](#) 951

17.3.4 Constructor(Handle as Integer)

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

Function: The constructor to build an NSDraggingInfo object with a handle.

Notes: Handle should be a Cocoa object reference implementing the NSDraggingInfo protocol.

See also:

- [17.3.3 Constructor](#) 951

17.3.5 namesOfPromisedFilesDroppedAtDestination(dropDestination as FolderItem) as string()

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

Function: Sets the drop location for promised files and returns the names of the files that the receiver promises to create there.

Notes: dropDestination: A folderitem specifying the drop location for promised files.

Return an array of file names, which are not full paths.

Drag destinations should invoke this method within their performDragOperation method. The source may or may not have created the files by the time this method returns.

17.3.6 promisedFilesDroppedAtDestination(dropDestination as FolderItem) as FolderItem()

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

Function: Sets the drop location for promised files and returns the files that the receiver promises to create there.

Notes: dropDestination: A folderitem specifying the drop location for promised files.

Return an array of folderitems for the files.

The files may not yet exist.

Drag destinations should invoke this method within their performDragOperation method. The source may or may not have created the files by the time this method returns.

17.3.7 slideDraggedImageTo(screenPoint as NSPointMBS)

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

Function: Slides the image to a specified location.

Notes: screenPoint: A point that specifies a location in the screen coordinate system.

This method can be used to adjust the location to which the dragged image will slide back if the drag is rejected.

It should only be invoked from within the destination's implementation of prepareForDragOperation, and will only have effect if the destination rejects the drag.

This method is invoked after the user has released the image but before it is removed from the screen.

17.3.8 Properties

17.3.9 animatesToDestination as boolean

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

Function: Whether the dragging formation animates while the drag is over this destination.

Notes: During the conclusion of an accepted drag, if this property is set to true, the drag manager will animate each dragging image to their NSDraggingFormationNone locations. Otherwise, the drag images are removed without any animation.

This property is inspected between `prepareForDragOperation` and `performDragOperation`. You should enumerate through the dragging items during `performDragOperation` to set the item's `draggingFrame` to the correct destinations.

Available in OS X v10.7 and later.
(Read and Write property)

17.3.10 draggedImage as Variant

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

Function: Returns the image being dragged.

Notes: Value is a `NSImageMBS` object. Returned as Variant to reduce plugin dependencies.

This image object visually represents the data put on the pasteboard during the drag operation; however, it is the pasteboard data and not this image that is ultimately utilized in the dragging operation.

This method returns non-nil for a local drag, but nil for a cross-process drag. With the new multi-image dragging capabilities, a cross-process destination may participate and change the drag image. But it still cannot get the current drag image.

(Read only property)

17.3.11 draggedImageLocation as NSPointMBS

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

Function: Returns the current location of the dragged image's origin.

Notes: Returns the dragged image's origin, in the base coordinate system of the destination object's window.

The image moves along with the mouse pointer (the position of which is given by `draggingLocation`) but may be positioned at some offset.

(Read only property)

17.3.12 draggingDestinationWindow as Variant

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

Function: Returns the destination window for the dragging operation.

Notes: Value is a NSWindowMBS object. Returned as Variant to reduce plugin dependencies.

Either this window is the destination itself, or it contains the view object that is the destination.

(Read only property)

17.3.13 draggingFormation as Integer

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

Function: Returns the dragging formation while the drag is over this destination.

Notes: Set this property to change the formation of the drag items. This is generally done during the updateDraggingItemsForDrag method or whenever you enumerate the dragging items.

The default value is the current drag formation.

Note: Set this property before or after the NSDraggingInfo or NSDraggingSession class's method enumerateDraggingItemsWithOptions not inside the enumeration Block.

Available in OS X v10.7 and later.

(Read and Write property)

17.3.14 draggingLocation as NSPointMBS

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

Function: Returns the current location of the mouse pointer in the base coordinate system of the destination object's window.

Notes: (Read only property)

17.3.15 draggingPasteboard as Variant

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

Function: Returns the pasteboard object that holds the data being dragged.

Notes: Value is a NSPasteboardMBS object. Returned as Variant to reduce plugin dependencies.

The dragging operation that is ultimately performed utilizes this pasteboard data and not the image returned by the draggedImage method.

(Read only property)

17.3.16 draggingSequenceNumber as Integer

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

Function: Returns a number that uniquely identifies the dragging session.

Notes: (Read only property)

17.3.17 draggingSource as Variant

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

Function: Returns the source, or owner, of the dragged data.

Notes: This method returns nil if the source is not in the same application as the destination. The dragging source implements methods from the NSDraggingSource protocol.

(Read only property)

17.3.18 draggingSourceOperationMask as Integer

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

Function: Returns the dragging operation mask of the dragging source.

Notes: The dragging operation mask, which is declared by the dragging source through the NSDraggingSource sourceOperationMaskForDraggingContext method (preferred) or the NSDraggingSource draggingSourceOperationMaskForLocal method. If the source does not permit any dragging operations, this method should return NSDragOperationNone.

If the source permits dragging operations, the elements in the mask are one or more of the constants described in "Obtaining Information About the Dragging Session", combined using the C bitwise OR operator.

If the user is holding down a modifier key during the dragging session and the source does not prohibit modifier keys from affecting the drag operation (through its ignoreModifierKeysWhileDragging method), then the operating system combines the dragging operation value that corresponds to the modifier key (see the descriptions below) with the source's mask using the C bitwise AND operator.

The modifier keys are associated with the dragging operation options shown below:

Modifier Key	Dragging Operation
Control	NSDragOperationLink
Option	NSDragOperationCopy
Command	NSDragOperationGeneric

(Read only property)

17.3.19 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

17.3.20 numberOfValidItemsForDrop as Integer

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

Function: Specifies the number of valid items for a drop operation.

Notes: During `draggingEntered` or `draggingUpdated`, you are responsible for returning the drag operation. In some cases, you may accept some, but not all items on the dragging pasteboard. (For example, your application may only accept image files.)

If you only accept some of the items, set this property to the number of items accepted so the drag manager can update the drag count badge.

When `updateDraggingItemsForDrag` is called, you should set the image of non-valid dragging items to `nil`. If none of the drag items are valid then you should not `updateItems`, simply return `NSDragOperationNone` from your implementation of `draggingEntered` and, or `draggingUpdated` and do not modify any drag item properties.

Available in OS X v10.7 and later.

(Read and Write property)

17.3.21 Constants

Visual Format Constants

Constant	Value	Description
NSDraggingFormationDefault	0	The system determined formation. Available in OS X v10.7 and later.
NSDraggingFormationList	3	Drag images are laid out vertically, non-overlapping with the left edges aligned. Available in OS X v10.7 and later.
NSDraggingFormationNone	1	Drag images maintain their set positions relative to each other/ Available in OS X v10.7 and later.
NSDraggingFormationPile	2	Drag images are placed on top of each other with random rotations. Available in OS X v10.7 and later.
NSDraggingFormationStack	4	Drag images are laid out overlapping diagonally. Available in OS X v10.7 and later.

Drag Operations Constants

Constant	Value	Description
NSDragOperationAll_Obsolete	15	The NSDragOperationAll constant is deprecated. Use NSDragOperationEvery instead.
NSDragOperationCopy	1	The data represented by the image can be copied.
NSDragOperationDelete	32	The data can be deleted.
NSDragOperationEvery	-1	All of the above.
NSDragOperationGeneric	4	The operation can be defined by the destination.
NSDragOperationLink	2	
NSDragOperationMove	16	The data can be moved.
NSDragOperationNone	0	No drag operations are allowed.
NSDragOperationPrivate	8	The operation is negotiated privately between the source and the destination.

17.4 class NSDraggingItemMBS

17.4.1 class NSDraggingItemMBS

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

Function: The NSDraggingItem class encompasses a single dragged item within an NSDraggingSession instance.

Notes: See NSDraggingSessionMBS Class Reference for more information

When the NSDraggingSession method `beginDraggingSessionWithItems` is called, the dragging items passed to the method are consumed immediately and are not retained.

Blog Entries

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

17.4.2 Methods

17.4.3 Constructor(item as NSPasteboardItemMBS)

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

Function: Initializes a dragging item using the specified content.

Notes: item: The object that provides the dragging content.

When the developer creates an NSDraggingItem instance, it is for use with the view method `beginDraggingSessionWithItems`. During the invocation of that method, the item is placed onto the dragging pasteboard for the NSDraggingSession that contains the dragging item instance.

Available in OS X v10.7 and later.

17.4.4 item as Variant

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

Function: Returns the pasteboard reader or writer object dependent on the context of where this dragging item is used. (read-only)

Notes: When you create an NSDraggingItem instance, item is the pasteboardWriter passed to Constructor.

17.4.5 setDraggingFrame(frame as NSRectMBS, contents as Variant)

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

Function: Sets the item's dragging frame and contents.

Notes: frame: The item content frame in the same coordinate space that the draggingFrame.

contents: The item contents to display when dragging. Typically this is an NSImage, but a CGImageRef will also work.

Alternate single image component setter.

This method simplifies modifying the components of an NSDraggingItem when there is only one component.

This is a convenience method. This method sets the draggingFrame and creates a single NSDraggingImageComponent instance with one image corresponding to the NSDraggingImageComponentIconKey key. You should only use this method under the following conditions: the drag image for this item is composed of a single image., or there are a reasonable number of dragging item instances being created or enumerated.

This method will set the draggingFrame and imageComponents properties.
Available in OS X v10.7 and later.

17.4.6 Properties

17.4.7 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

17.4.8 draggingFrame as NSRectMBS

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

Function: The frame of the dragging item.

Notes: The dragging frame provides the spatial relationship between NSDraggingItem instances when the dragging formation is set to NSDraggingFormationNone.

The exact coordinate space of this rectangle is dependent on where it is used. The view that initiated the drag using beginDraggingSessionWithItems or the view your pass to the NSDraggingSession instance implantation of enumerateDraggingItemsWithOptions.

Available in OS X v10.7 and later.
(Read and Write computed property)

17.5 class NSDraggingSessionMBS

17.5.1 class NSDraggingSessionMBS

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

Function: The NSDraggingSession class encompasses a drag and drop action and allows modification of the drag while in progress.

Notes: You start a new dragging session by calling the NSView method `beginDraggingSessionWithItems` method. This method immediately returns and you can further modify the properties of the dragging session. The actual drag begins at the next turn of the run loop.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MonkeyBread Software Releases the MBS Real Studio plug-ins in version 13.1](#)
- [MBS Real Studio Plugins, version 13.1pr9](#)

17.5.2 Methods

17.5.3 Constructor

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

Function: The private constructor.

17.5.4 `draggingLeaderIndex` as Integer

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

Function: The index of the dragging item under the cursor.

Notes: The index is to an element in the array passed as the first parameter to the NSView method `beginDraggingSessionWithItem`.

The default is the NSDraggingItem closest to the location field in the event parameter that was passed to the `beginDraggingSessionWithItems` method.

Available in OS X v10.7 and later.

17.5.5 `draggingLocation` as NSPointMBS

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

Function: The current cursor location of the drag in screen coordinates. (read-only)

Notes: Available in OS X v10.7 and later.

17.5.6 draggingPasteboard as NSPasteboardMBS

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

Function: Returns the pasteboard object that contains the data being dragged. (read-only)

Notes: Available in OS X v10.7 and later.

17.5.7 draggingSequenceNumber as Integer

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

Function: Returns a number that uniquely identifies the dragging session. (read-only)

Notes: Available in OS X v10.7 and later.

17.5.8 Properties

17.5.9 Handle as Integer

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

Function: The internal object reference.

Notes: (Read and Write property)

17.5.10 animatesToStartingPositionsOnCancelOrFail as boolean

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

Function: Controls whether the dragging image animates back to its starting point on a cancelled or failed drag.

Notes: This property should be set immediately after creating the dragging session.

The default value is true.

Available in OS X v10.7 and later.

(Read and Write computed property)

17.5.11 draggingFormation as Integer

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

Function: Controls the dragging formation when the drag is not over the source or a valid destination.

Notes: Setting this value causes the dragging formation to change immediately, provided a valid destination has not overridden the behavior. If the dragging session hasn't started yet, the dragging items will animate into formation immediately upon start. It is highly recommended to never change the formation when starting a drag.

The default value is NSDraggingFormationNone.

Available in OS X v10.7 and later.

(Read and Write computed property)

17.5.12 Constants

Constants

Constant	Value	Description
NSDraggingContextOutsideApplication	0	Whether a drag terminates within or outside the application. The dragging terminates outside the application. Available in OS X v10.7 and later.
NSDraggingContextWithinApplication	1	Whether a drag terminates within or outside the application. The dragging terminates within the application. Available in OS X v10.7 and later.

Visual Format Constants

Constant	Value	Description
NSDraggingFormationDefault	0	The system determined formation. Available in OS X v10.7 and later.
NSDraggingFormationList	3	Drag images are laid out vertically, non-overlapping with the left edges aligned. Available in OS X v10.7 and later.
NSDraggingFormationNone	1	Drag images maintain their set positions relative to each other. Available in OS X v10.7 and later.
NSDraggingFormationPile	2	Drag images are placed on top of each other with random rotations. Available in OS X v10.7 and later.
NSDraggingFormationStack	4	Drag images are laid out overlapping diagonally. Available in OS X v10.7 and later.

Chapter 18

iCloud

18.1 class NSFileManagerMBS

18.1.1 class NSFileManagerMBS

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: The NSFileManager class enables you to perform many generic file-system operations and insulates an application from the underlying file system.

Notes: In Xojo you can do the same with folderitem. But for some iCloud related tasks you need to use this class.

In Cocoa applications, a file manager object is usually your first interaction with the file system. You use this object to locate, create, copy, and move files and directories. You also use this object to get information about files and directories, such as its size, modification date, and BSD permissions. You can also use a file manager object to change the values of many file and directory attributes.

The NSFileManagerMBS class supports both the folderitem and path string as ways to specify the location of a file or directory. The use of the folderitem is generally preferred for specifying file-system items because they can convert path information to a more efficient representation internally. You can also obtain a bookmark from an folderitem, which is similar to an alias and offers a more sure way of locating the file or directory later.

In iOS 5.0 and later and in Mac OS X v10.7 and later, NSFileManager includes methods for managing items stored in the cloud. Files and directories tagged for cloud storage are synced to the user's MobileMe account so that they can be made available to the user's iOS devices and Macintosh computers. Changes to an item in one location are propagated to all other locations to ensure the items stay in sync.

Blog Entries

- [CallKit extension for Xojo iOS application](#)

- [Multithreaded plugin functions can increase speed of Xojo application](#)
- [MBS Xojo Plugins, version 19.5pr6](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 19.3](#)
- [MBS Xojo Plugins, version 19.3pr3](#)
- [MBS Xojo Plugins, version 18.5pr1](#)
- [MBS Xojo Plugins, version 18.2pr1](#)
- [MBS Xojo / Real Studio Plugins, version 16.5pr7](#)
- [MBS Xojo / Real Studio Plugins, version 16.4pr4](#)
- [Notes...](#)

Xojo Developer Magazine

- [17.5, page 10: News](#)

18.1.2 Methods

18.1.3 `attributesOfFileSystemForPath(item as folderitem, byref error as NSErrorMBS) as Dictionary`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns a dictionary that describes the attributes of the mounted file system on which a given path resides.

Notes: path: A folderitem pointing to the mounted file system.

error: On input, a pointer to an error object. If an error occurs, this pointer is set to an actual error object containing the error information. You may specify nil for this parameter if you do not want the error information.

Returns an Dictionary object that describes the attributes of the mounted file system on which path resides, or nil if an error occurs. See File-System Attribute Keys for a description of the keys available in the dictionary.

See also:

- [18.1.4 `attributesOfFileSystemForPath\(path as string, byref error as NSErrorMBS\) as Dictionary`](#) 966

18.1.4 `attributesOfFileSystemForPath(path as string, byref error as NSErrorMBS) as Dictionary`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns a dictionary that describes the attributes of the mounted file system on which a given path resides.

Notes: path: Any pathname within the mounted file system.

error: On input, a pointer to an error object. If an error occurs, this pointer is set to an actual error object containing the error information. You may specify nil for this parameter if you do not want the error information.

Returns an Dictionary object that describes the attributes of the mounted file system on which path resides, or nil if an error occurs. See File-System Attribute Keys for a description of the keys available in the dictionary.

See also:

- 18.1.3 attributesOfFileSystemForPath(item as folderitem, byref error as NSErrorMBS) as Dictionary 966

18.1.5 attributesOfItemAtPath(item as folderitem, byref error as NSErrorMBS) as Dictionary

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

Function: Returns the attributes of the item at a given path.

Example:

```
dim n as new NSFileManagerMBS
dim e as NSErrorMBS
dim d as Dictionary = n.attributesOfItemAtPath(SpecialFolder.Desktop, e)

break // check dictionary in debugger
```

Notes: item: The folderitem of a file or directory.

Error: If an error occurs, this is set to an actual error object containing the error information.

Return a Dictionary object that describes the attributes (file, directory, symlink, and so on) of the file specified by path. The keys in the dictionary are described in File Attribute Keys. (see NSFile* methods)

Special Considerations

This method does not traverse symbolic links. If the item at the path is a symbolic link—that is, the value of the NSFileType key in the attributes dictionary is NSFileTypeSymbolicLink—you can use the destinationOfSymbolicLinkAtPath method to retrieve the path of the item pointed to by the link. You can also use the stringByResolvingSymlinksInPath method of NSString to resolve links in the path before retrieving the item's attributes.

Available in Mac OS X v10.5 and later.

See also:

- 18.1.6 `attributesOfItemAtPath(path as string, byref error as NSErrorMBS) as Dictionary` 968

18.1.6 `attributesOfItemAtPath(path as string, byref error as NSErrorMBS) as Dictionary`

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

Function: Returns the attributes of the item at a given path.

Notes: path: The path of a file or directory.

Error: If an error occurs, this is set to an actual error object containing the error information.

Return a Dictionary object that describes the attributes (file, directory, symlink, and so on) of the file specified by path. The keys in the dictionary are described in File Attribute Keys. (see `NSFile*` methods)

Special Considerations

This method does not traverse symbolic links. If the item at the path is a symbolic link—that is, the value of the `NSFileType` key in the attributes dictionary is `NSFileTypeSymbolicLink`—you can use the `destinationOfSymbolicLinkAtPath` method to retrieve the path of the item pointed to by the link. You can also use the `stringByResolvingSymlinksInPath` method of `NSString` to resolve links in the path before retrieving the item's attributes.

Available in Mac OS X v10.5 and later.

See also:

- 18.1.5 `attributesOfItemAtPath(item as folderitem, byref error as NSErrorMBS) as Dictionary` 967

18.1.7 `changeCurrentDirectory(folder as folderitem) as boolean`

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Changes the path of the current working directory to the specified path.

Example:

```
dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Applications
if m.changeCurrentDirectory(f) then
MsgBox m.currentDirectory.NativePath
end if
```

Notes: folder: The path of the directory to which to change.

Returns true if successful, otherwise false.

All relative pathnames refer implicitly to the current working directory. Changing the current working directory affects only paths created in the current process.

See also:

- 18.1.8 `changeCurrentDirectory(path as string)` as boolean 969

18.1.8 `changeCurrentDirectory(path as string)` as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Changes the path of the current working directory to the specified path.

Example:

```
dim m as new NSFileManagerMBS
if m.changeCurrentDirectory("/Users") then
MsgBox m.currentDirectoryPath
end if
```

Notes: path: The path of the directory to which to change.

Returns true if successful, otherwise false.

All relative pathnames refer implicitly to the current working directory. Changing the current working directory affects only paths created in the current process.

See also:

- 18.1.7 `changeCurrentDirectory(folder as folderitem)` as boolean 968

18.1.9 Constructor

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: The constructor.

18.1.10 `containerFolderForSecurityApplicationGroupIdentifier(groupIdentifier as string)` as folderItem

Plugin Version: 14.3, Platform: macOS, Targets: All.

Function: Returns the container directory associated with the specified security application group ID.

Notes: As explained in App Sandbox Design Guide, groups of sandboxed apps that need to share files and other information can request a container directory as part of their entitlements. These directories are stored

in
textasciitilde /Library/Group Containers/.

When called with a valid group identifier, this method returns the location of that directory as an `folderitem`. This method also creates the directory if it does not yet exist.

Important: Your app must have a `com.apple.security.application-groups` entitlement for the specified application group.

Available in OS X v10.8 and later.

18.1.11 `containerURLForSecurityApplicationGroupIdentifier(groupIdentifier as string) as string`

Plugin Version: 14.3, Platform: macOS, Targets: All.

Function: Returns the container directory associated with the specified security application group ID.

Notes: As explained in App Sandbox Design Guide, groups of sandboxed apps that need to share files and other information can request a container directory as part of their entitlements. These directories are stored in
textasciitilde /Library/Group Containers/.

When called with a valid group identifier, this method returns the location of that directory as an URL. This method also creates the directory if it does not yet exist.

Important: Your app must have a `com.apple.security.application-groups` entitlement for the specified application group.

Available in OS X v10.8 and later.

18.1.12 `contentsEqual(path1 as folderitem, path2 as folderitem) as boolean`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the files or directories in specified paths have the same contents.

Notes: `path1`: The path of a file or directory to compare with the contents of `path2`.

`path2`: The path of a file or directory to compare with the contents of `path1`.

Returns true if file or directory specified in `path1` has the same contents as that specified in `path2`, otherwise

false.

If path1 and path2 are directories, the contents are the list of files and subdirectories each contains—contents of subdirectories are also compared. For files, this method checks to see if they’re the same file, then compares their size, and finally compares their contents. This method does not traverse symbolic links, but compares the links themselves.

18.1.13 `copyItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean`

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Copies the item at the specified path to a new location synchronously.

Notes: Source: The path to the file or directory you want to move.

Dest: The path at which to place the copy of srcPath. This path must include the name of the file or directory in its new location.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was copied successfully or the file manager’s delegate aborted the operation deliberately. Returns false if an error occurred.

When copying items, the current process must have permission to read the file or directory at Source and write the parent directory of Dest. If the item at Source is a directory, this method copies the directory and all of its contents, including any hidden files. If a file with the same name already exists at Dest, this method aborts the copy attempt and returns an appropriate error. If the last component of srcPath is a symbolic link, only the link is copied to the new path.

Available in OS X v10.5 and later.

See also:

- 18.1.14 `copyItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean`
971

18.1.14 `copyItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean`

Plugin Version: 19.5, Platform: macOS, Targets: All.

Function: Copies the item at the specified path to a new location synchronously.

Notes: Source: The path to the file or directory you want to move.

Dest: The path at which to place the copy of srcPath. This path must include the name of the file or directory in its new location.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was copied successfully or the file manager's delegate aborted the operation deliberately. Returns false if an error occurred.

When copying items, the current process must have permission to read the file or directory at Source and write the parent directory of Dest. If the item at Source is a directory, this method copies the directory and all of its contents, including any hidden files. If a file with the same name already exists at Dest, this method aborts the copy attempt and returns an appropriate error. If the last component of srcPath is a symbolic link, only the link is copied to the new path.

Available in OS X v10.5 and later.

See also:

- 18.1.13 copyItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean 971

18.1.15 copyItemMT(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean

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

Function: Copies the item at the specified path to a new location synchronously.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim d as FolderItem = SpecialFolder.Desktop.Child("output.rtf")
dim e as NSErrorMBS

if not NSFileManagerMBS.defaultManager.copyItemMT(f,d,e) then
  MsgBox e.LocalizedDescription
else
  MsgBox "OK"
end if
```

Notes: Source: The path to the file or directory you want to move.

Dest: The path at which to place the copy of srcPath. This path must include the name of the file or directory in its new location.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was copied successfully or the file manager's delegate aborted the operation deliberately. Returns false if an error occurred.

When copying items, the current process must have permission to read the file or directory at Source and

write the parent directory of Dest. If the item at Source is a directory, this method copies the directory and all of its contents, including any hidden files. If a file with the same name already exists at Dest, this method aborts the copy attempt and returns an appropriate error. If the last component of srcPath is a symbolic link, only the link is copied to the new path.

Available in OS X v10.5 and later.

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.

See also:

- 18.1.16 copyItemMT(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean
973

18.1.16 copyItemMT(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean

Plugin Version: 19.5, Platform: macOS, Targets: All.

Function: Copies the item at the specified path to a new location synchronously.

Notes: Source: The path to the file or directory you want to move.

Dest: The path at which to place the copy of srcPath. This path must include the name of the file or directory in its new location.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was copied successfully or the file manager's delegate aborted the operation deliberately. Returns false if an error occurred.

When copying items, the current process must have permission to read the file or directory at Source and write the parent directory of Dest. If the item at Source is a directory, this method copies the directory and all of its contents, including any hidden files. If a file with the same name already exists at Dest, this method aborts the copy attempt and returns an appropriate error. If the last component of srcPath is a symbolic link, only the link is copied to the new path.

Available in OS X v10.5 and later.

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.

See also:

- 18.1.15 copyItemMT(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean
972

18.1.17 createDirectory(Path as folderItem, createIntermediates as boolean = true, attrs as Dictionary = nil, byref error as NSErrorMBS) as boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a directory with given attributes at the specified path.

Notes: path: A folderitme identifying the directory to create. You may specify a full path or a path that is relative to the current working directory. This parameter must not be nil.

createIntermediates: If true, this method creates any non-existent parent directories as part of creating the directory in path. If NO, this method fails if any of the intermediate parent directories does not exist. This method also fails if any of the intermediate path elements corresponds to a file and not a directory.

attributes: The file attributes for the new directory and any newly created intermediate directories. You can set the owner and group numbers, file permissions, and modification date. If you specify nil for this parameter or omit a particular value, one or more default values are used as described in the discussion. For a list of keys you can include in this dictionary, see Constants. Some of the keys, such as NSFileHFSCreatorCode and NSFileHFSTypeCode, do not apply to directories.

error: If an error occurs, this variable is set to an actual error object containing the error information.

Returns true if the directory was created, true if createIntermediates is set and the directory already exists, or false if an error occurred.

If you specify nil for the attributes parameter, this method uses a default set of values for the owner, group, and permissions of any newly created directories in the path. Similarly, if you omit a specific attribute, the default value is used. The default values for newly created directories are as follows:

- Permissions are set according to the umask of the current process. For more information, see umask.
- The owner ID is set to the effective user ID of the process.
- The group ID is set to that of the parent directory.

See also:

- 18.1.18 createDirectory(Path as String, createIntermediates as boolean = true, attrs as Dictionary = nil, byref error as NSErrorMBS) as boolean 974

18.1.18 createDirectory(Path as String, createIntermediates as boolean = true, attrs as Dictionary = nil, byref error as NSErrorMBS) as boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a directory with given attributes at the specified path.

Notes: path: A path string identifying the directory to create. You may specify a full path or a path that is relative to the current working directory. This parameter must not be nil.

createIntermediates: If true, this method creates any non-existent parent directories as part of creating the

directory in path. If NO, this method fails if any of the intermediate parent directories does not exist. This method also fails if any of the intermediate path elements corresponds to a file and not a directory.

attributes: The file attributes for the new directory and any newly created intermediate directories. You can set the owner and group numbers, file permissions, and modification date. If you specify nil for this parameter or omit a particular value, one or more default values are used as described in the discussion. For a list of keys you can include in this dictionary, see Constants. Some of the keys, such as NSFileHFSCreatorCode and NSFileHFSTypeCode, do not apply to directories.

error: If an error occurs, this variable is set to an actual error object containing the error information.

Returns true if the directory was created, true if createIntermediates is set and the directory already exists, or false if an error occurred.

If you specify nil for the attributes parameter, this method uses a default set of values for the owner, group, and permissions of any newly created directories in the path. Similarly, if you omit a specific attribute, the default value is used. The default values for newly created directories are as follows:

- Permissions are set according to the umask of the current process. For more information, see umask.
- The owner ID is set to the effective user ID of the process.
- The group ID is set to that of the parent directory.

See also:

- 18.1.17 createDirectory(Path as folderItem, createIntermediates as boolean = true, attrs as Dictionary = nil, byref error as NSErrorMBS) as boolean 974

18.1.19 createFile(Path as folderItem, contents as MemoryBlock, attrs as Dictionary = nil) as boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a file with the specified content and attributes at the given location.

Notes: path: The folderitem for the new file.

contents: A data object containing the contents of the new file.

attributes: A dictionary containing the attributes to associate with the new file. You can use these attributes to set the owner and group numbers, file permissions, and modification date. For a list of keys, see NSFileAttributeKey. If you specify nil for attributes, the file is created with a set of default attributes.

Returns true if the operation was successful or if the item already exists, otherwise false.

If you specify nil for the attributes parameter, this method uses a default set of values for the owner, group, and permissions of any newly created directories in the path. Similarly, if you omit a specific attribute, the default value is used. The default values for newly created files are as follows:

- Permissions are set according to the umask of the current process. For more information, see umask.
- The owner ID is set to the effective user ID of the process.
- The group ID is set to that of the parent directory.

If a file already exists at path, this method overwrites the contents of that file if the current process has the appropriate privileges to do so.

See also:

- 18.1.20 createFile(Path as String, contents as MemoryBlock, attrs as Dictionary = nil) as boolean 976

18.1.20 createFile(Path as String, contents as MemoryBlock, attrs as Dictionary = nil) as boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a file with the specified content and attributes at the given location.

Notes: path: The path for the new file.

contents: A data object containing the contents of the new file.

attributes: A dictionary containing the attributes to associate with the new file. You can use these attributes to set the owner and group numbers, file permissions, and modification date. For a list of keys, see NSFileAttributeKey. If you specify nil for attributes, the file is created with a set of default attributes.

Returns true if the operation was successful or if the item already exists, otherwise false.

If you specify nil for the attributes parameter, this method uses a default set of values for the owner, group, and permissions of any newly created directories in the path. Similarly, if you omit a specific attribute, the default value is used. The default values for newly created files are as follows:

- Permissions are set according to the umask of the current process. For more information, see umask.
- The owner ID is set to the effective user ID of the process.
- The group ID is set to that of the parent directory.

If a file already exists at path, this method overwrites the contents of that file if the current process has the appropriate privileges to do so.

See also:

- 18.1.19 createFile(Path as folderItem, contents as MemoryBlock, attrs as Dictionary = nil) as boolean 975

18.1.21 createSymbolicLink(file as folderitem, destFile as folderitem, byref error as NSErrorMBS) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates a symbolic link at the specified URL that points to an item at the given URL.

Example:

```
dim m as new NSFileManagerMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim destfile as FolderItem = SpecialFolder.Desktop.Child("Notes.rtf")
dim error as NSErrorMBS

if m.createSymbolicLink(file, destfile, error) then
  MsgBox "OK"
else
  MsgBox "Error: "+error.localizedDescription
end if
```

Notes: file: The path at which to create the new symbolic link. The last path component is used as the name of the link.

destFile: The path that contains the item to be pointed to by the link. In other words, this is the destination of the link.

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns true if the symbolic link was created or false if an error occurred. This method also returns false if a file, directory, or link already exists at path.

This method does not traverse symbolic links contained in either path or destPath.

Available in Mac OS X v10.5 and later.

As of Mac OS X 10.7 tests here, it seems like an alias file is created, not a symbolic link.

See also:

- 18.1.22 createSymbolicLink(path as string, destPath as string, byref error as NSErrorMBS) as boolean
977

18.1.22 createSymbolicLink(path as string, destPath as string, byref error as NSErrorMBS) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates a symbolic link that points to the specified destination.

Notes: path: The path at which to create the new symbolic link. The last path component is used as the

name of the link.

destPath: The path that contains the item to be pointed to by the link. In other words, this is the destination of the link.

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns true if the symbolic link was created or false if an error occurred. This method also returns false if a file, directory, or link already exists at path.

This method does not traverse symbolic links contained in either path or destPath.

Available in Mac OS X v10.5 and later.

As of Mac OS X 10.7 tests here, it seems like an alias file is created, not a symbolic link.

See also:

- 18.1.21 createSymbolicLink(file as folderitem, destFile as folderitem, byref error as NSErrorMBS) as boolean 977

18.1.23 destinationOfSymbolicLinkAtPath(file as folderitem, byref error as NSErrorMBS) as string

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

Function: Returns the path of the item pointed to by a symbolic link.

Notes: file: The folderitem of a file or directory. Be aware that Xojo may already have resolved the symlink!

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns a string containing the path of the directory or file to which the symbolic link path refers, or "" upon failure. If the symbolic link is specified as a relative path, that relative path is returned.

See also:

- 18.1.24 destinationOfSymbolicLinkAtPath(path as string, byref error as NSErrorMBS) as string 978

18.1.24 destinationOfSymbolicLinkAtPath(path as string, byref error as NSErrorMBS) as string

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

Function: Returns the path of the item pointed to by a symbolic link.

Example:

```
dim f as new NSFileManagerMBS
```

```
dim e as NSErrorMBS
```

```
dim p as string = f.destinationOfSymbolicLinkAtPath("/tmp", e)
```

```

if e = nil then
  MsgBox p
else
  MsgBox e.LocalizedDescription
end if

```

Notes: path: The path of a file or directory.

error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns a string containing the path of the directory or file to which the symbolic link path refers, or "" upon failure. If the symbolic link is specified as a relative path, that relative path is returned.

See also:

- 18.1.23 destinationOfSymbolicLinkAtPath(file as folderitem, byref error as NSErrorMBS) as string 978

18.1.25 displayName(path as folderitem) as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns the display name of the file or directory at a specified path.

Notes: path: The path of a file or directory.

The name of the file or directory at path in a localized form appropriate for presentation to the user. If there is no file or directory at path, or if an error occurs, returns path as is.

Display names are user-friendly names for files. They are typically used to localize standard file and directory names according to the user's language settings. They may also reflect other modifications, such as the removal of filename extensions. Such modifications are used only when displaying the file or directory to the user and do not reflect the actual path to the item in the file system. For example, if the current user's preferred language is French, the following code fragment logs the name Biblioth@que and not the name Library, which is the actual name of the directory.

18.1.26 evictUbiquitousItem(item as folderitem, byref error as NSErrorMBS) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Removes the local copy of the specified cloud-based item.

Notes: item: Specify the file or directory in iCloud storage.

error: If an error occurs, this pointer is set to an NSError object containing information about the error.

Returns true if the local item was removed successfully or false if it was not. If false is returned, an NSError object describing the error is returned in the error parameter.

This method does not remove the item from the cloud. It removes only the local version. You can use this method to force iCloud to download a new version of the file or directory from the server.

To delete a file permanently from the user's iCloud storage, use the regular NSFileManager routines for deleting files and directories. Remember that deleting items from iCloud cannot be undone. Once deleted, the item is gone forever.

Available in Mac OS X v10.7 and later.

18.1.27 fileExists(path as folderitem) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a file or directory exists at a specified path.

Example:

```
dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Desktop.Child("notes.rtf")
```

```
MsgBox "File exists: "+str(m.fileExists(f))
```

Notes: path: The path of the file or directory. I

Returns true if a file at the specified path exists or false if the file's does not exist or its existence could not be determined.

If the file at path is inaccessible to your application, perhaps because one or more parent directories are inaccessible, this method returns false. If the final element in path specifies a symbolic link, this method traverses the link and returns true or false based on the existence of the file at the link destination.

Note: Attempting to predicate behavior based on the current state of the file system or a particular file on the file system is not recommended. Doing so can cause odd behavior or race conditions. It's far better to attempt an operation (such as loading a file or creating a directory), check for errors, and handle those errors gracefully than it is to try to figure out ahead of time whether the operation will succeed. For more information on file system race conditions, see Race Conditions, File Operations, and Interprocess Communication in Secure Coding Guide.

http://developer.apple.com/library/mac/documentation/Security/Conceptual/SecureCodingGuide/Articles/Race-Conditions.html#//apple_ref/doc/uid/TP40002585

See also:

- 18.1.28 fileExists(path as folderitem, byref isDirectory as boolean) as boolean 981

18.1.28 fileExists(path as folderitem, byref isDirectory as boolean) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a file or directory exists at a specified path.

Example:

```
dim m as new NSFileManagerMBS
```

```
dim f as FolderItem = SpecialFolder.Desktop
```

```
dim directory as Boolean
```

```
MsgBox "File exists: "+str(m.fileExists(f, directory))+ " and is Directory: "+str(directory)
```

Notes: path: The path of a file or directory.

isDirectory: Upon return, contains true if path is a directory or if the final path element is a symbolic link that points to a directory, otherwise contains false. If path doesn't exist, the return value is undefined.

Returns true if a file at the specified path exists or false if the file's does not exist or its existence could not be determined.

If the file at path is inaccessible to your application, perhaps because one or more parent directories are inaccessible, this method returns false. If the final element in path specifies a symbolic link, this method traverses the link and returns true or false based on the existence of the file at the link destination.

If you need to further determine if path is a package, use the isFilePackageAtPath method of NSWorkspaceMBS.

Note: Attempting to predicate behavior based on the current state of the file system or a particular file on the file system is not recommended. Doing so can cause odd behavior or race conditions. It's far better to attempt an operation (such as loading a file or creating a directory), check for errors, and handle those errors gracefully than it is to try to figure out ahead of time whether the operation will succeed. For more information on file system race conditions, see Race Conditions, File Operations, and Interprocess Communication in Secure Coding Guide.

http://developer.apple.com/library/mac/documentation/Security/Conceptual/SecureCodingGuide/Articles/Race-Conditions.html#//apple_ref/doc/uid/TP40002585

See also:

- 18.1.27 fileExists(path as folderitem) as boolean 980

18.1.29 FileForUbiquityContainerIdentifier(containerIdentifier as string) as folderitem

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the iCloud directory associated with the specified container ID.

Notes: containerID: Specify the container ID of the cloud-based storage container. The string you specify must not contain wildcards and must be of the form <TEAMID>.<CONTAINER>, where <TEAMID> is your development team ID and <CONTAINER> describes the bundle identifier of the container you want to access. The container identifiers for your application must be declared in the com.apple.developer.ubiquity-container-identifiers entitlement.

If you specify "", this method returns the first container listed in the com.apple.developer.ubiquity-container-identifiers entitlement.

Returns a folderitem pointing to the specified container directory or nil if the container could not be located or if iCloud storage is unavailable for the current user or device.

You can use the folderitem returned by this method to build paths to files and directories in the user's iCloud storage. Each application that syncs documents to the cloud must have at least one associated container directory in which to put those files. This container directory can be unique to the application or shared by multiple applications. You use this method to retrieve the folderitem for that container directory.

In addition to writing to its own container directory, an application can write to any container directory for which it has the appropriate permission. Each additional container directory should be listed as an additional value in the com.apple.developer.ubiquity-container-identifiers entitlement.

Note: The development team ID that precedes each container ID string is the unique identifier associated with your development team. You can find this string in the Member Center of the Apple Developer website (<http://developer.apple.com/membercenter>). From the Member Center home page, select the Your Account tab and then select Organization Profile from the column on the left of that tab. Your team's identifier is in the Company/Organization ID field.

The first time you call this method for a given container directory, iOS extends your application sandbox to include that container directory. Thus, it is important that you call this method at least once before trying to search for files in iCloud. And if your application accesses multiple container directories, you should call the method once for each directory.

Available in Mac OS X v10.7 and later.

FileForUbiquityContainerIdentifier returns folderitem while URLForUbiquityContainerIdentifier returns URL string.

18.1.30 fileManagerWithAuthorization(authorization as NSError) as NSError

Plugin Version: 19.3, Platform: macOS, Targets: Desktop only.

Function: Initializes a file manager object that is authorized to perform privileged file system operations.

Notes: Returns nil in case of errors.

This method returns an `NSFileManagerMBS` instance that can perform file system operations previously allowed by the user via `NSWorkspaceMBS.requestAuthorization`. Each `NSWorkspaceAuthorizationMBS` you receive requires creating a new `NSFileManagerMBS` instance using this method.

Only the following `NSFileManagerMBS` methods currently take advantage of an authorization:

- `createSymbolicLink` (`NSWorkspaceAuthorizationTypeCreateSymbolicLink`)
- `setAttributes` (`NSWorkspaceAuthorizationTypeSetAttributes`)
- `replaceItem` (`NSWorkspaceAuthorizationTypeReplaceFile`)

Note that an `NSWorkspaceAuthorizationTypeSetAttributes` authorization only enables `setAttributes` to modify the following attributes:

- `NSFileOwnerAccountID`
- `NSFileGroupOwnerAccountID`
- `NSFilePosixPermissions`

Also note that for `replaceItem`, the `backupItemName` and `options` parameters will be ignored.

These methods may also fail with any of the following errors:

- `NSWorkspaceAuthorizationInvalidError`: The provided `NSWorkspaceAuthorization` expired or is invalid.
- `NSFileWriteUnknownError`: The application failed to communicate with a helper process, or a file system error occurred.
- `NSFileWriteNoPermissionError`: The operation failed for any other reason, including the user denying access to the resource, or access to a resource is denied by system policy.

All other `NSFileManager` methods invoked on this instance will behave normally.

The app using this may need code signature with `com.apple.developer.security.privileged-file-operations` entitlement for the sandboxed application and may need to be in applications folder.

18.1.31 `homeDirectoryForUser(Name as string) as FolderItem`

Plugin Version: 18.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the home directory for the specified user.

Notes: `userName`: The username of the owner of the desired home directory.

Returns the `FolderItem` containing the location of the specified user, the user's home directory, or `nil` if no such user exists or the user's home directory is not available.

18.1.32 `isDeletableFile(path as folderitem) as boolean`

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the invoking object appears able to delete a specified file.

Example:

```
dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Desktop.Child("notes.rtf")
```

```
MsgBox "Can delete: " + str(m.isDeletableFile(f))
```

Notes: `path`: A file path.

Returns true if the current process has delete privileges for the file at `path`; otherwise false if the process does not have delete privileges or the existence of the file could not be determined.

For a directory or file to be deletable, the current process must either be able to write to the parent directory of `path` or it must have the same owner as the item at `path`. If `path` is a directory, every item contained in `path` must be deletable by the current process.

If the file at `path` is inaccessible to your application, perhaps because it does not have search privileges for one or more parent directories, this method returns false. This method does not traverse symbolic links in the path.

Note: Attempting to predicate behavior based on the current state of the file system or a particular file on the file system is not recommended. Doing so can cause odd behavior or race conditions. It's far better to attempt an operation (such as loading a file or creating a directory), check for errors, and handle those errors gracefully than it is to try to figure out ahead of time whether the operation will succeed. For more information on file system race conditions, see Race Conditions, File Operations, and Interprocess Communication in Secure Coding Guide.

http://developer.apple.com/library/mac/documentation/Security/Conceptual/SecureCodingGuide/Articles/Race-Conditions.html#//apple_ref/doc/uid/TP40002585

18.1.33 isExecutableFile(path as folderitem) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the operating system appears able to execute a specified file.

Example:

```
dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Desktop.Child("notes.rtf")

MsgBox "Can execute: "+str(m.isExecutableFile(f))
```

Notes: path: A file path.

Returns true if the current process has execute privileges for the file at path; otherwise false if the process does not have execute privileges or the existence of the file could not be determined.

If the file at path is inaccessible to your application, perhaps because it does not have search privileges for one or more parent directories, this method returns false. This method traverses symbolic links in the path. This method also uses the real user ID and group ID, as opposed to the effective user and group IDs, to determine if the file is executable.

Note: Attempting to predicate behavior based on the current state of the file system or a particular file on the file system is not recommended. Doing so can cause odd behavior or race conditions. It's far better to attempt an operation (such as loading a file or creating a directory), check for errors, and handle those errors gracefully than it is to try to figure out ahead of time whether the operation will succeed. For more information on file system race conditions, see Race Conditions, File Operations, and Interprocess Communication in Secure Coding Guide:

http://developer.apple.com/library/mac/documentation/Security/Conceptual/SecureCodingGuide/Articles/Race-Conditions.html#//apple_ref/doc/uid/TP40002585

18.1.34 isReadableFile(path as folderitem) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the invoking object appears able to read a specified file.

Example:

```

dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Desktop.Child("notes.rtf")

MsgBox "Can read: "+str(m.isReadableFile(f))

```

Notes: path: A file path.

Returns true if the current process has read privileges for the file at path; otherwise false if the process does not have read privileges or the existence of the file could not be determined.

If the file at path is inaccessible to your application, perhaps because it does not have search privileges for one or more parent directories, this method returns false. This method traverses symbolic links in the path. This method also uses the real user ID and group ID, as opposed to the effective user and group IDs, to determine if the file is readable.

Note: Attempting to predicate behavior based on the current state of the file system or a particular file on the file system is not recommended. Doing so can cause odd behavior or race conditions. It's far better to attempt an operation (such as loading a file or creating a directory), check for errors, and handle those errors gracefully than it is to try to figure out ahead of time whether the operation will succeed. For more information on file system race conditions, see Race Conditions, File Operations, and Interprocess Communication in Secure Coding Guide.

http://developer.apple.com/library/mac/documentation/Security/Conceptual/SecureCodingGuide/Articles/Race-Conditions.html#//apple_ref/doc/uid/TP40002585

18.1.35 isUbiquitousItem(item as folderitem) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a Boolean indicating whether the item is targeted for storage in iCloud.

Example:

```

dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Desktop.Child("notes.rtf")

MsgBox "Is in iCloud: "+str(m.isUbiquitousItem(f))

```

Notes: item: Specify the folderitem for the file or directory whose status you want to check.

Returns true if the item is targeted for iCloud storage or false if it is not. This method also returns false if no item exists at url.

This method reflects only whether the item should be stored in iCloud because a call was made to the `setUbiquitous` method with a value of `true` for its flag parameter. This method does not reflect whether the file has actually been uploaded to any iCloud servers. To determine a file's upload status, check the `IsUploadedMBS` function in the `folderitem` class.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.36 `isUbiquitousItem(URL as string)` as boolean

987

18.1.36 `isUbiquitousItem(URL as string)` as boolean

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a Boolean indicating whether the item is targeted for storage in iCloud.

Notes: `item`: Specify the URL for the file or directory whose status you want to check.

Returns `true` if the item is targeted for iCloud storage or `false` if it is not. This method also returns `false` if no item exists at `url`.

This method reflects only whether the item should be stored in iCloud because a call was made to the `setUbiquitous` method with a value of `true` for its flag parameter. This method does not reflect whether the file has actually been uploaded to any iCloud servers. To determine a file's upload status, check the `IsUploadedMBS` function in the `folderitem` class.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.35 `isUbiquitousItem(item as folderitem)` as boolean

986

18.1.37 `isWritableFile(path as folderitem)` as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the invoking object appears able to write to a specified file.

Example:

```
dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Desktop.Child("notes.rtf")
```

```
MsgBox "Can write: " + str(m.isWritableFile(f))
```

Notes: path: A file path.

Returns true if the current process has write privileges for the file at path; otherwise false if the process does not have write privileges or the existence of the file could not be determined.

If the file at path is inaccessible to your application, perhaps because it does not have search privileges for one or more parent directories, this method returns false. This method traverses symbolic links in the path. This method also uses the real user ID and group ID, as opposed to the effective user and group IDs, to determine if the file is writable.

Note: Attempting to predicate behavior based on the current state of the file system or a particular file on the file system is not recommended. Doing so can cause odd behavior or race conditions. It's far better to attempt an operation (such as loading a file or creating a directory), check for errors, and handle those errors gracefully than it is to try to figure out ahead of time whether the operation will succeed. For more information on file system race conditions, see Race Conditions, File Operations, and Interprocess Communication in Secure Coding Guide:

http://developer.apple.com/library/mac/documentation/Security/Conceptual/SecureCodingGuide/Articles/Race-Conditions.html#//apple_ref/doc/uid/TP40002585

18.1.38 lastPathComponent(pathOrURL as string) as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the last path component of a file URL or file path.

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

18.1.39 linkItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Creates a hard link between the items at the specified paths.

Notes: Source: The path that specifies the item you wish to link to. The value in this parameter must not be nil.

Dest: The path that identifies the location where the link will be created. The value in this parameter must not be nil.

error: If an error occurs, this pointer is set to an actual error object containing the error information.

Returns true if the hard link was created or NO if an error occurred. This method also returns false if a file, directory, or link already exists at dstPath.

Use this method to create hard links between files in the current file system. If Source is a directory, this method creates a new directory at Dest and then creates hard links for the items in that directory. If Source is (or contains) a symbolic link, the symbolic link is copied to the new location and not converted to a hard link.

Available in OS X v10.5 and later.

See also:

- 18.1.40 linkItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean 989

18.1.40 linkItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean

Plugin Version: 19.5, Platform: macOS, Targets: All.

Function: Creates a hard link between the items at the specified paths.

Notes: Source: The path that specifies the item you wish to link to. The value in this parameter must not be nil.

Dest: The path that identifies the location where the link will be created. The value in this parameter must not be nil.

error: If an error occurs, this pointer is set to an actual error object containing the error information.

Returns true if the hard link was created or NO if an error occurred. This method also returns false if a file, directory, or link already exists at dstPath.

Use this method to create hard links between files in the current file system. If Source is a directory, this method creates a new directory at Dest and then creates hard links for the items in that directory. If Source is (or contains) a symbolic link, the symbolic link is copied to the new location and not converted to a hard link.

Available in OS X v10.5 and later.

See also:

- 18.1.39 linkItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean 988

18.1.41 moveItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Moves the file or directory at the specified path to a new location synchronously.

Notes: source: The path to the file or directory you want to move. This parameter must not be nil.

dest: The new path for the item in Source. This path must include the name of the file or directory in its new location. This parameter must not be nil.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was moved successfully or the file manager's delegate aborted the operation deliberately. Returns NO if an error occurred.

When moving items, the current process must have permission to read the item at Source and write the parent directory of Dest. If the item at Source is a directory, this method moves the directory and all of its contents, including any hidden files. If an item with the same name already exists at Dest, this method aborts the move attempt and returns an appropriate error. If the last component of Source is a symbolic link, only the link is moved to the new path; the item pointed to by the link remains at its current location.

If the source and destination of the move operation are not on the same volume, this method copies the item first and then removes it from its current location. This behavior may trigger additional delegate notifications related to copying and removing individual items.

Available in OS X v10.5 and later.

See also:

- 18.1.42 `moveItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean`
990

18.1.42 `moveItem(sourcePath as String, destPath as String, byref error as NSErrorMBS) as boolean`

Plugin Version: 19.5, Platform: macOS, Targets: All.

Function: Moves the file or directory at the specified path to a new location synchronously.

Notes: `source`: The path to the file or directory you want to move. This parameter must not be nil.

`dest`: The new path for the item in Source. This path must include the name of the file or directory in its new location. This parameter must not be nil.

`error`: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was moved successfully or the file manager's delegate aborted the operation deliberately. Returns NO if an error occurred.

When moving items, the current process must have permission to read the item at Source and write the parent directory of Dest. If the item at Source is a directory, this method moves the directory and all of its contents, including any hidden files. If an item with the same name already exists at Dest, this method aborts the move attempt and returns an appropriate error. If the last component of Source is a symbolic link, only the link is moved to the new path; the item pointed to by the link remains at its current location.

If the source and destination of the move operation are not on the same volume, this method copies the item first and then removes it from its current location. This behavior may trigger additional delegate notifications related to copying and removing individual items.

Available in OS X v10.5 and later.

See also:

- 18.1.41 `moveItem(source as folderItem, dest as folderItem, byref error as NSErrorMBS) as boolean` 989

18.1.43 `NSFileAppendOnly` as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates whether the file is read-only.

The corresponding value is a number containing a Boolean value.

Available in Mac OS X v10.2 and later.

18.1.44 `NSFileBusy` as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates whether the file is busy.

The corresponding value is a number containing a Boolean value.

Available in Mac OS X v10.4 and later.

18.1.45 `NSFileCreationDate` as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's creation date.

The corresponding value is a date.

Available in Mac OS X v10.2 and later.

18.1.46 `NSFileDeviceIdentifier` as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the identifier for the device on which the

file resides.

The corresponding value is a number containing an Int32.
Available in Mac OS X v10.0 and later.

18.1.47 NSFileExtensionHidden as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates whether the file's extension is hidden.
The corresponding value is a number containing a Boolean value.
Available in Mac OS X v10.1 and later.

18.1.48 NSFileGroupOwnerAccountID as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Example:

```
Dim file As FolderItem = SpecialFolder.Desktop.Child("test.rtf")
Dim fileManager As New NSFileManagerMBS
```

```
dim error as NSErrorMBS
Dim Values As Dictionary = fileManager.attributesOfItemAtPath(file, error)
Dim Permissions As Integer = Values.Value(fileManager.NSFilePosixPermissions)
```

```
MsgBox "Group: "+_
values.Value(fileManager.NSFileGroupOwnerAccountID)+_
" "+_
values.Value(fileManager.NSFileGroupOwnerAccountName)
```

Notes: The key in a file attribute dictionary whose value indicates the file's group ID.
The corresponding value is a number containing an Int32.
Available in Mac OS X v10.2 and later.

18.1.49 NSFileGroupOwnerAccountName as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Example:

```
Dim file As FolderItem = SpecialFolder.Desktop.Child("test.rtf")
Dim fileManager As New NSFileManagerMBS

dim error as NSErrorMBS
Dim Values As Dictionary = fileManager.attributesOfItemAtPath(file, error)
Dim Permissions As Integer = Values.Value(fileManager.NSFilePosixPermissions)

MsgBox "Group: "+_
values.Value(fileManager.NSFileGroupOwnerAccountID)+_
" "+_
values.Value(fileManager.NSFileGroupOwnerAccountName)
```

Notes: The key in a file attribute dictionary whose value indicates the group name of the file's owner. The corresponding value is a string. Available in Mac OS X v10.0 and later.

18.1.50 NSFileHFSCreatorCode as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's HFS creator code. The corresponding value is a number containing an `Int32`. See "HFS File Types" for possible values. Available in Mac OS X v10.1 and later.

18.1.51 NSFileHFSTypeCode as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's HFS type code. The corresponding value is a number containing an `Int32`. See "HFS File Types" for possible values. Available in Mac OS X v10.1 and later.

18.1.52 NSFileImmutable as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Example:

```
Dim n As New NSFileManagerMBS
dim e as NSErrorMBS
Dim d As Dictionary = n.attributesOfItemAtPath("/Users/cs/Desktop/test.rtf", e)
dim locked as Boolean = d.Lookup(n.NSFileImmutable, false)
```

Break // inspect in debugger

Notes: The key in a file attribute dictionary whose value indicates whether the file is mutable. The corresponding value is a number containing a Boolean value. Available in Mac OS X v10.2 and later.

18.1.53 NSFileManagerUnmountDissentingProcessIdentifierErrorKey as string

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The key for the error dictionary containing process ID of the process blocking the unmount.

18.1.54 NSFileModificationDate as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's last modified date. The corresponding value is a date.

Available in Mac OS X v10.0 and later.

18.1.55 NSFileOwnerAccountID as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Example:

```

Dim file As FolderItem = SpecialFolder.Desktop.Child("test.rtf")
Dim fileManager As New NSFileManagerMBS

dim error as NSErrorMBS
Dim Values As Dictionary = fileManager.attributesOfItemAtPath(file, error)
Dim Permissions As Integer = Values.Value(fileManager.NSFilePosixPermissions)

MsgBox "Owner: "+_
values.Value(fileManager.NSFileOwnerAccountID)+_
" "+_
values.Value(fileManager.NSFileOwnerAccountName)

```

Notes: The key in a file attribute dictionary whose value indicates the file's owner's account ID. The corresponding value is a number containing an Int32. Available in Mac OS X v10.2 and later.

18.1.56 NSFileOwnerAccountName as string

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

Function: One of the keys to access file attribute values contained in dictionary used by setAttributes, attributesOfItemAtPath, createDirectory, and createFile.

Example:

```

Dim file As FolderItem = SpecialFolder.Desktop.Child("test.rtf")
Dim fileManager As New NSFileManagerMBS

dim error as NSErrorMBS
Dim Values As Dictionary = fileManager.attributesOfItemAtPath(file, error)
Dim Permissions As Integer = Values.Value(fileManager.NSFilePosixPermissions)

'MsgBox "POSIX Permissions: "+Oct(Permissions)
MsgBox "Owner: "+_
values.Value(fileManager.NSFileOwnerAccountID)+_
" "+_
values.Value(fileManager.NSFileOwnerAccountName)

```

Notes: The key in a file attribute dictionary whose value indicates the name of the file's owner. The corresponding value is a string. Available in Mac OS X v10.0 and later.

18.1.57 NSFilePosixPermissions as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Example:

```
Dim file As FolderItem = SpecialFolder.Desktop.Child("test.rtf")
Dim fileManager As New NSFileManagerMBS

dim error as NSErrorMBS
Dim Values As Dictionary = fileManager.attributesOfItemAtPath(file, error)
Dim Permissions As Integer = Values.Value(fileManager.NSFilePosixPermissions)

MsgBox "POSIX Permissions: "+oct(Permissions)
```

Notes: The key in a file attribute dictionary whose value indicates the file's Posix permissions. The corresponding value is a number. Available in Mac OS X v10.0 and later.

18.1.58 NSFileReferenceCount as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's reference count. The corresponding value is a number containing an `Int32`. The number specifies the number of hard links to a file. Available in Mac OS X v10.0 and later.

18.1.59 NSFileSize as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's size in bytes. The corresponding value is a number containing an `Int64`. Important If the file has a resource fork, the returned value does not include the size of the resource fork.

18.1.60 NSFileSystemFileNumber as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's filesystem file number.

The corresponding value is a number containing an `Int32`. The value corresponds to the value of `st_ino`, as returned by `stat(2)`.

Available in Mac OS X v10.0 and later.

18.1.61 NSFileSystemFreeNodes as string

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

Function: One of the keys to access the file attribute values contained in the dictionary object returned from the `attributesOfFileSystemForPath` function.

Notes: The key in a file system attribute dictionary whose value indicates the number of free nodes in the file system.

The corresponding value is a number that specifies the number of free nodes in the file system.

Available in Mac OS X v10.0 and later.

18.1.62 NSFileSystemFreeSize as string

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

Function: One of the keys to access the file attribute values contained in the dictionary object returned from the `attributesOfFileSystemForPath` function.

Notes: The key in a file system attribute dictionary whose value indicates the amount of free space on the file system.

The corresponding value is a number that specifies the amount of free space on the file system in bytes. The value is determined by `statfs()`.

Available in Mac OS X v10.0 and later.

18.1.63 NSFileSystemNodes as string

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

Function: One of the keys to access the file attribute values contained in the dictionary object returned from the `attributesOfFileSystemForPath` function.

Notes: The key in a file system attribute dictionary whose value indicates the number of nodes in the file system.

The corresponding value is a number that specifies the number of nodes in the file system.

Available in Mac OS X v10.0 and later.

18.1.64 NSFileSystemNumber as string

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

Function: One of the keys to access the file attribute values contained in the dictionary object returned from the `attributesOfFileSystemForPath` function.

Notes: The key in a file system attribute dictionary whose value indicates the filesystem number of the file system.

The corresponding value is a number that specifies the filesystem number of the file system. The value corresponds to the value of `st_dev`, as returned by `stat(2)`.

Available in Mac OS X v10.0 and later.

18.1.65 NSFileSystemSize as string

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

Function: One of the keys to access the file attribute values contained in the dictionary object returned from the `attributesOfFileSystemForPath` function.

Notes: The key in a file system attribute dictionary whose value indicates the size of the file system.

The corresponding value is a number that specifies the size of the file system in bytes. The value is determined by `statfs()`.

Available in Mac OS X v10.0 and later.

18.1.66 NSFileType as string

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

Function: One of the keys to access file attribute values contained in dictionary used by `setAttributes`, `attributesOfItemAtPath`, `createDirectory`, and `createFile`.

Notes: The key in a file attribute dictionary whose value indicates the file's type.

The corresponding value is a string (see `NSFileType*` shared methods for possible values).

Available in Mac OS X v10.0 and later.

18.1.67 NSFileTypeBlockSpecial as string

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

Function: One of the possible `NSFileType` values.

Example:

```
// check first hard disc
dim e as NSErrorMBS
dim n as new NSFileManagerMBS
dim d as Dictionary = n.attributesOfItemAtPath("/dev/disk0", e)
MsgBox d.Value(n.NSFileType)
```

Notes: Block special file (e.g. hard disk)

18.1.68 NSFileTypeCharacterSpecial as string

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

Function: One of the possible NSFileType values.

Notes: Character special file

18.1.69 NSFileTypeDirectory as string

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

Function: One of the possible NSFileType values.

Example:

```
dim e as NSErrorMBS
dim n as new NSFileManagerMBS
dim d as Dictionary = n.attributesOfItemAtPath("/System", e)
MsgBox d.Value(n.NSFileType)
```

Notes: Directory

18.1.70 NSFileTypeRegular as string

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

Function: One of the possible NSFileType values.

Notes: Regular file

18.1.71 `NSFileTypeSocket` as string

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

Function: One of the possible `NSFileType` values.

Notes: Socket (a socket visible in file system as a file)

18.1.72 `NSFileTypeSymbolicLink` as string

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

Function: One of the possible `NSFileType` values.

Notes: Symbolic link

18.1.73 `NSFileTypeUnknown` as string

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

Function: One of the possible `NSFileType` values.

Notes: Unknown

18.1.74 `NSUbiquityIdentityDidChangeNotification` as string

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sent after the iCloud (`NSUbiquityIdentityDidChangeNotification`) identity has changed.

Notes: The system generates this notification when the user logs into or out of an iCloud account or enables or disables the syncing of documents and data. This notification is your cue to update caches and any interface elements displaying iCloud-related content. For example, hide all references to iCloud files when the user logs out of iCloud.

When your app receives this notification, get the new token from the `NSURLRelationship` property. The value of that token is `nil` if the user disabled iCloud or logged out. There is no `userInfo` dictionary.

18.1.75 `pathExtension(pathOrURL as string)` as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the path extension of a file URL or file path.

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

18.1.76 removeItem(file as folderitem, byref error as NSErrorMBS) as boolean

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Removes the file or directory at the specified path.

Notes: path: A path string indicating the file or directory to remove. If the path specifies a directory, the contents of that directory are recursively removed.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was removed successfully or if path was nil. Returns false if an error occurred. If the delegate aborts the operation for a file, this method returns true. However, if the delegate aborts the operation for a directory, this method returns false.

Available in OS X v10.5 and later.

See also:

- 18.1.77 removeItem(path as string, byref error as NSErrorMBS) as boolean 1001

18.1.77 removeItem(path as string, byref error as NSErrorMBS) as boolean

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Removes the file or directory at the specified path.

Notes: path: A path string indicating the file or directory to remove. If the path specifies a directory, the contents of that directory are recursively removed.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if the item was removed successfully or if path was nil. Returns false if an error occurred. If the delegate aborts the operation for a file, this method returns true. However, if the delegate aborts the operation for a directory, this method returns false.

Available in OS X v10.5 and later.

See also:

- 18.1.76 removeItem(file as folderitem, byref error as NSErrorMBS) as boolean 1001

18.1.78 replaceItem(originalItem as FolderItem, newItem as FolderItem, backupItemName as String, options as integer, byref resultingItem as FolderItem, byref error as NSErrorMBS) as Boolean

Plugin Version: 19.3, Platform: macOS, Targets: All.

Function: Replaces the contents of the item at the specified URL in a manner that ensures no data loss occurs.

Example:

```

Dim fileManager As New NSFileManagerMBS

Dim source As FolderItem = SpecialFolder.Desktop.Child("test.rtf")
Dim dest As FolderItem = SpecialFolder.Desktop.Child("dest.rtf")

Dim result As Boolean
Dim output As FolderItem
Dim error As NSErrorMBS

result = fileManager.replaceItemAtURL(dest, source, "", 0, output, error)

If result Then
  MsgBox output.NativePath
Else
  MsgBox error.LocalizedDescription
End If

```

Notes: originalItem: The item containing the content you want to replace.

newItem: The item containing the new content for originalItem. It is recommended that you put this item in a temporary directory as provided by the OS. If a temporary directory is not available, put this item in a uniquely named directory that is in the same directory as the original item.

backupItemName: If provided, the name used to create a backup of the original item.

The backup is automatically placed in the same directory as the original item. If an error occurs during the creation of the backup item, the operation fails. If there is already an item with the same name as the backup item, that item will be removed.

The backup item will be removed in the event of success unless the `NSFileManagerItemReplacementWithoutDeletingBackupItem` option is provided in options.

options: The options to use during the replacement. Typically, you pass `NSFileManagerItemReplacementUsingNewMetadataOnly` for this parameter, which uses only the metadata from the new item. You can also combine the options described in `NSFileManagerItemReplacementOptions` using the C-bitwise OR operator.

resultingItem: On input, a variable for a folderitem variable. When the item is replaced, this variable is set to the folderitem of the new item. If no new file system object is required, the folderitem in this parameter may be the same passed to the originalItem parameter. However, if a new file system object is required, the folderitem may be different. For example, replacing an RTF document with an RTFD document requires the creation of a new file.

error: On input, a variable for an error object. If an error occurs, this variable is set to an error object containing the error information.

Returns true if the replacement was successful or false if an error occurred.

By default, the creation date, permissions, Finder label and color, and Spotlight comments of the original item are preserved on the new item. You can configure which metadata is preserved using the options parameter.

This method works only when the `originalItem` and `newItem` parameters are located on the same volume. Attempting to call this method by passing `originalItem` and `newItem` parameters that have locations on different volumes results in an error. Instead, you can call the `URLForDirectory` method, passing `NSItemReplacementDirectory` as the search path directory, to get a temporary URL on the destination's volume that is suitable for use with this method.

If an error occurs and the original item is not in the original location or a temporary location, the resulting error object contains a user info dictionary with the key `”NSFileOriginalItemLocationKey”`. The value assigned to that key is an URL with the location of the item. The error code is one of the file-related errors described in `NSError Codes`.

18.1.79 `replaceItemAtURL(originalItemURL as string, newItemURL as String, backupItemName as String, options as integer, byref resultingURL as String, byref error as NSErrorMBS) as Boolean`

Plugin Version: 19.3, Platform: macOS, Targets: All.

Function: Replaces the contents of the item at the specified URL in a manner that ensures no data loss occurs.

Notes: `originalItemURL`: The item containing the content you want to replace.

`newItemURL`: The item containing the new content for `originalItemURL`. It is recommended that you put this item in a temporary directory as provided by the OS. If a temporary directory is not available, put this item in a uniquely named directory that is in the same directory as the original item.

`backupItemName`: If provided, the name used to create a backup of the original item.

The backup is automatically placed in the same directory as the original item. If an error occurs during the creation of the backup item, the operation fails. If there is already an item with the same name as the backup item, that item will be removed.

The backup item will be removed in the event of success unless the `NSFileManagerItemReplacementWithoutDeletingBackupItem` option is provided in `options`.

`options`: The options to use during the replacement. Typically, you pass `NSFileManagerItemReplacementUsingNewMetadataOnly` for this parameter, which uses only the metadata from the new item. You can also combine the options described in `NSFileManagerItemReplacementOptions` using the C-bitwise OR operator.

`resultingURL`: On input, a variable for an URL variable. When the item is replaced, this variable is set to the URL of the new item. If no new file system object is required, the URL object in this parameter may be the same passed to the `originalItemURL` parameter. However, if a new file system object is required, the URL object may be different. For example, replacing an RTF document with an RTFD document requires the creation of a new file.

`error`: On input, a variable for an error object. If an error occurs, this variable is set to an error object containing the error information.

Returns true if the replacement was successful or false if an error occurred.

By default, the creation date, permissions, Finder label and color, and Spotlight comments of the original item are preserved on the new item. You can configure which metadata is preserved using the `options` parameter.

This method works only when the `originalItemURL` and `newItemURL` parameters are located on the same

volume. Attempting to call this method by passing `originalItemURL` and `newItemURL` parameters that have locations on different volumes results in an error. Instead, you can call the `URLForDirectory` method, passing `NSItemReplacementDirectory` as the search path directory, to get a temporary URL on the destination's volume that is suitable for use with this method.

If an error occurs and the original item is not in the original location or a temporary location, the resulting error object contains a user info dictionary with the key `”NSFileOriginalItemLocationKey”`. The value assigned to that key is an URL with the location of the item. The error code is one of the file-related errors described in `NSError Codes`.

18.1.80 `setAttributes(attributesDic as dictionary, item as folderitem, byref error as NSErrorMBS) as boolean`

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

Function: Sets the attributes of the specified file or directory.

Notes: `attributes`: A dictionary containing as keys the attributes to set for path and as values the corresponding value for the attribute. You can set the following attributes: `NSFileBusy`, `NSFileCreationDate`, `NSFileExtensionHidden`, `NSFileGroupOwnerAccountID`, `NSFileGroupOwnerAccountName`, `NSFileHFSCreatorCode`, `NSFileHFSTypeCode`, `NSFileImmutable`, `NSFileModificationDate`, `NSFileOwnerAccountID`, `NSFileOwnerAccountName`, `NSFilePosixPermissions`. You can change single attributes or any combination of attributes; you need not specify keys for all attributes.

`item`: The folderitem of a file or directory.

`error`: If an error occurs, this is set to an actual error object containing the error information.

Returns true if all changes succeed. If any change fails, returns false, but it is undefined whether any changes actually occurred.

Discussion

As in the POSIX standard, the application either must own the file or directory or must be running as superuser for attribute changes to take effect. The method attempts to make all changes specified in attributes and ignores any rejection of an attempted modification. If the last component of the path is a symbolic link it is traversed.

The `NSFilePosixPermissions` value must be initialized with the code representing the POSIX file-permissions bit pattern. `NSFileHFSCreatorCode` and `NSFileHFSTypeCode` will only be heeded when path specifies a file.

Available in Mac OS X v10.5 and later.

See also:

- 18.1.81 `setAttributes(attributesDic as dictionary, path as string, byref error as NSErrorMBS) as boolean` 1005

18.1.81 setAttributes(attributesDic as dictionary, path as string, byref error as NSErrorMBS) as boolean

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

Function: Sets the attributes of the specified file or directory.

Notes: attributes: A dictionary containing as keys the attributes to set for path and as values the corresponding value for the attribute. You can set the following attributes: NSFileBusy, NSFileCreationDate, NSFileExtensionHidden, NSFileGroupOwnerAccountID, NSFileGroupOwnerAccountName, NSFileHFSCreatorCode, NSFileHFSTypeCode, NSFileImmutable, NSFileModificationDate, NSFileOwnerAccountID, NSFileOwnerAccountName, NSFilePosixPermissions. You can change single attributes or any combination of attributes; you need not specify keys for all attributes.

path: The path of a file or directory.

error: If an error occurs, this is set to an actual error object containing the error information.

Returns true if all changes succeed. If any change fails, returns false, but it is undefined whether any changes actually occurred.

Discussion

As in the POSIX standard, the application either must own the file or directory or must be running as superuser for attribute changes to take effect. The method attempts to make all changes specified in attributes and ignores any rejection of an attempted modification. If the last component of the path is a symbolic link it is traversed.

The NSFilePosixPermissions value must be initialized with the code representing the POSIX file-permissions bit pattern. NSFileHFSCreatorCode and NSFileHFSTypeCode will only be heeded when path specifies a file.

Available in Mac OS X v10.5 and later.

See also:

- 18.1.80 setAttributes(attributesDic as dictionary, item as folderitem, byref error as NSErrorMBS) as boolean 1004

18.1.82 setUbiquitous(flag as boolean, item as folderitem, destitem as folderitem, byref error as NSErrorMBS) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Sets whether the item at the specified URL should be stored in the cloud.

Notes: flag: Specify true to move the item to iCloud or false to remove it from iCloud (if it is there currently).

item: Specify the folderitem of the item (file or directory) that you want to store in iCloud.

destitem: Specify the location in iCloud at which to store the file or directory. This folderitem must be constructed from a folderitem returned by the URLForUbiquityContainerIdentifier method, which you use to retrieve the desired iCloud container directory. The folderitem you specify may contain additional subdirec-

ories so that you can organize your files hierarchically in iCloud. However, you are responsible for creating those intermediate subdirectories (using the `NSFileManagerMBS` or `folderitem` class) in your iCloud container directory.

error: If an error occurs, this pointer is set to an `NSError` object containing information about the error. You may specify `nil` for this parameter if you do not want the error information.

Returns `yes` if the item's status was updated successfully or `false` if an error occurred. If this method returns `false` and you specified a value for the error parameter, this method returns an error object.

Use this method to move a file from its current location to iCloud. For files located in an application's sandbox, this involves physically removing the file from the sandbox directory. (The system extends your application's sandbox privileges to give it access to files it moves to iCloud.) You can also use this method to move files out of iCloud and back into a local directory.

Your application must have an active file presenter object configured to monitor the specified file or directory before calling this method. When you specify `true` for the `flag` parameter, this method attempts to move the file or directory to the cloud and returns `true` if it is successful. This method also notifies your file presenter of the new location of the file so that your application can continue to operate on it.

Important: Do not call this method from your application's main thread. This method performs a coordinated write operation on the file you specify, and calling this method from the main thread can trigger a deadlock with the file presenter you have monitoring the file. Instead, use a dispatch queue (other than the main thread queue) to perform the method call on a secondary thread. You can always message your main thread after the call finishes to update the rest of your application's data structures.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.83 `setUbiquitous(flag as boolean, item as folderitem, destURL as string, byref error as NSErrorMBS) as boolean` 1006

18.1.83 `setUbiquitous(flag as boolean, item as folderitem, destURL as string, byref error as NSErrorMBS) as boolean`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Sets whether the item at the specified URL should be stored in the cloud.

Notes: `flag`: Specify `true` to move the item to iCloud or `false` to remove it from iCloud (if it is there currently).

`item`: Specify the `folderitem` of the item (file or directory) that you want to store in iCloud.

`destitem`: Specify the location in iCloud at which to store the file or directory. This `folderitem` must be constructed from a `folderitem` returned by the `URLForUbiquityContainerIdentifier` method, which you use to retrieve the desired iCloud container directory. The `folderitem` you specify may contain additional subdirectories so that you can organize your files hierarchically in iCloud. However, you are responsible for creating

those intermediate subdirectories (using the `NSFileManagerMBS` or `folderitem` class) in your iCloud container directory.

error: If an error occurs, this pointer is set to an `NSError` object containing information about the error. You may specify `nil` for this parameter if you do not want the error information.

Returns `yes` if the item's status was updated successfully or `false` if an error occurred. If this method returns `false` and you specified a value for the error parameter, this method returns an error object.

Use this method to move a file from its current location to iCloud. For files located in an application's sandbox, this involves physically removing the file from the sandbox directory. (The system extends your application's sandbox privileges to give it access to files it moves to iCloud.) You can also use this method to move files out of iCloud and back into a local directory.

Your application must have an active file presenter object configured to monitor the specified file or directory before calling this method. When you specify `true` for the flag parameter, this method attempts to move the file or directory to the cloud and returns `true` if it is successful. This method also notifies your file presenter of the new location of the file so that your application can continue to operate on it.

Important: Do not call this method from your application's main thread. This method performs a coordinated write operation on the file you specify, and calling this method from the main thread can trigger a deadlock with the file presenter you have monitoring the file. Instead, use a dispatch queue (other than the main thread queue) to perform the method call on a secondary thread. You can always message your main thread after the call finishes to update the rest of your application's data structures.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.82 `setUbiquitous(flag as boolean, item as folderitem, destitem as folderitem, byref error as NSErrorMBS) as boolean` 1005

18.1.84 `startDownloadingUbiquitousItem(item as folderitem, byref error as NSErrorMBS) as boolean`

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Starts downloading (if necessary) the specified item to the local system.

Notes: `item:` Specify the folderitem for the file or directory in the cloud that you want to download.

`error:` On input, a pointer to variable for an `NSError` object. If an error occurs, this pointer is set to an `NSError` object containing information about the error. `Y`

Returns `true` if the download started successfully or was not necessary, otherwise `false`. If `false` is returned and `error` is not `nil`, an `NSError` object describing the error is returned in that parameter.

If a cloud-based file or directory has not been downloaded yet, calling this method starts the download process. If the item exists locally, calling this method synchronizes the local copy with the version in the cloud.

For a given folderitem, you can determine if a file is downloaded by getting the value of the `IsDownloaded` method. You can also use related methods to determine the current progress in downloading the file.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.85 `startDownloadingUbiquitousItem(URL as string, byref error as NSErrorMBS)` as boolean 1008

18.1.85 `startDownloadingUbiquitousItem(URL as string, byref error as NSErrorMBS)` as boolean

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Starts downloading (if necessary) the specified item to the local system.

Notes: item: Specify the URL for the file or directory in the cloud that you want to download.

error: On input, a pointer to variable for an NSError object. If an error occurs, this pointer is set to an NSError object containing information about the error. Y

Returns true if the download started successfully or was not necessary, otherwise false. If false is returned and error is not nil, an NSError object describing the error is returned in that parameter.

If a cloud-based file or directory has not been downloaded yet, calling this method starts the download process. If the item exists locally, calling this method synchronizes the local copy with the version in the cloud.

For a given folderitem, you can determine if a file is downloaded by getting the value of the `IsDownloaded` method. You can also use related methods to determine the current progress in downloading the file.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.84 `startDownloadingUbiquitousItem(item as folderitem, byref error as NSErrorMBS)` as boolean 1007

18.1.86 `stringByAbbreviatingWithTildeInPath(path as string)` as string

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string that replaces the current home directory portion of the current path with a tilde (`~`) character.

Notes: A new string based on the current string object. If the new string specifies a file in the current home directory, the home directory portion of the path is replaced with a tilde (`~`) character. If the string does not specify a file in the current home directory, this method returns a new string object whose path is unchanged from the path in the current string.

Note that this method only works with file paths. It does not work for string representations of URLs.

For sandboxed apps in OS X, the current home directory is not the same as the user's home directory. For a sandboxed app, the home directory is the app's home directory. So if you specified a path of `/Users/<current_user>/file.txt` for a sandboxed app, the returned path would be unchanged from the original. However, if you specified the same path for an app not in a sandbox, this method would replace the `/Users/<current_user>` portion of the path with a tilde.

18.1.87 `stringByAppendingPathComponent(path as string, Component as string) as string`

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string made by appending to the receiver a given string.

Notes: The following table illustrates the effect of this method on a variety of different paths, assuming that aString is supplied as `~/scratch.tiff`:

Receiver's String Value	Resulting String
<code>~/tmp</code>	<code>~/tmp/scratch.tiff</code>
<code>~/tmp/</code>	<code>~/tmp/scratch.tiff</code>
<code>~/</code>	<code>~/scratch.tiff</code>
<code>~/</code> (an empty string)	<code>~/scratch.tiff</code>

Note that this method only works with file paths (not, for example, string representations of URLs).

18.1.88 `stringByAppendingPathExtension(path as string, Extension as string) as string`

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string made by appending to the receiver an extension separator followed by a given extension.

Notes: The following table illustrates the effect of this method on a variety of different paths, assuming

that ext is supplied as ".tiff":

Receiver, "String Value"	Resulting String
"/tmp/scratch.old."	"/tmp/scratch.old.tiff."
"/tmp/scratch.",	"/tmp/scratch.tiff."
"/tmp/,"	"/tmp.tiff."
"/scratch."	"/scratch.tiff."

Note that adding an extension to "/tmp/" causes the result to be "/tmp.tiff" instead of "/tmp.tiff.". This difference is because a file named ".tiff" is not considered to have an extension, so the string is appended to the last nonempty path component.

Note that this method only works with file paths (not, for example, string representations of URLs).

Special Considerations

Prior to OS X v10.9 this method did not allow you to append file extensions to filenames starting with the tilde character (textasciitilde).

18.1.89 stringByDeletingLastPathComponent(path as string) as string

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string made by deleting the last path component from the receiver, along with any final path separator.

Notes: A new string made by deleting the last path component from the receiver, along with any final path separator. If the receiver represents the root path it is returned unaltered.

The following table illustrates the effect of this method on a variety of different paths:

Receiver, "String Value"	Resulting String
"/tmp/scratch.tiff."	"/tmp."
"/tmp/lock/,"	"/tmp."
"/tmp/,"	"/."
"/tmp."	"/."
"/,"	"/."
"/scratch.tiff."	"/" (an empty string)

Note that this method only works with file paths (not, for example, string representations of URLs).

18.1.90 stringByDeletingPathExtension(path as string) as string

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string made by deleting the extension (if any, and only the last) from the receiver.

Notes: A new string made by deleting the extension (if any, and only the last) from the receiver. Strips any trailing path separator before checking for an extension. If the receiver represents the root path, it is returned unaltered.

The following table illustrates the effect of this method on a variety of different paths:

Receiver, Ås String Value	Resulting String
, Åú/tmp/scratch.tiff, Åù	, Åú/tmp/scratch, Åù
, Åú/tmp/, Åù	, Åú/tmp, Åù
, Åúscratch.bundle/, Åù	, Åúscratch, Åù
, Åúscratch..tiff, Åù	, Åúscratch., Åù
, Åú.tiff, Åù	, Åú.tiff, Åù
, Åú/, Åù	, Åú/, Åù

Note that attempting to delete an extension from ".tiff" causes the result to be "@.tiff" instead of an empty string. This difference is because a file named ".tiff" is not considered to have an extension, so nothing is deleted. Note also that this method only works with file paths (not, for example, string representations of URLs).

18.1.91 stringByExpandingTildeInPath(path as string) as string

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string made by expanding the initial component of the receiver to its full path value.

Notes: A new string made by expanding the initial component of the receiver, if it begins with , Åú textasciitilde , Åù or , Åú textasciitilde user, Åù, to its full path value. Returns a new string matching the receiver if the receiver, Ås initial component can, Åt be expanded.

Note that this method only works with file paths (not, for example, string representations of URLs).

18.1.92 stringByResolvingSymlinksInPath(path as string) as string

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string made from the receiver by resolving all symbolic links and standardizing

path.

Notes: A new string made by resolving all symbolic links, then removing extraneous path components. For absolute paths, all symbolic links are guaranteed to be removed. For relative paths, symbolic links that can,Äôt be resolved are left unresolved in the returned string.

Returns self if an error occurs.

Note that this method only works with file paths (not, for example, string representations of URLs).

18.1.93 `stringByStandardizingPath(path as string) as string`

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Returns a new string made by removing extraneous path components from the receiver.

Notes: A new string made by performing the following operations:

- Expanding an initial tilde expression using `stringByExpandingTildeInPath`.
- Removing an initial component of `,Äú/private/var/automount,Äù`, `,Äú/var/automount,Äù`, or `,Äú/private,Äù` from the path, if the result still indicates an existing file or directory (checked by consulting the file system).
- Reducing empty components and references to the current directory (that is, the sequences `,Äú//,Äù` and `,Äú/.,Äù`) to single path separators.
- Removing a trailing slash from the last component.
- For absolute paths only, resolving references to the parent directory (that is, the component `,Äú..,Äù`) to the real parent directory if possible using `stringByResolvingSymlinksInPath`. For relative paths, references to the parent directory are left in place.
- Returns self if an error occurs.

Note that the path returned by this method may still have symbolic link components in it. Note also that this method only works with file paths (not, for example, string representations of URLs).

18.1.94 `trashItem(file as folderItem, byref Resulting as folderItem, byref error as NSErrorMBS) as boolean`

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Moves an item to the trash.

Notes: Returns true if the item at url was successfully moved to the trash. Since the operation may require

renaming the item to avoid a file name collision, this method returns, by reference, the resulting folderitem that the item was moved to.

If this method returns false, the item was not moved to the trash; the error parameter then contains information about the error.

Available in OS X v10.8 and later.

See also:

- 18.1.95 trashItem(Path as String, byref Resulting as FolderItem, byref error as NSErrorMBS) as boolean 1013

18.1.95 trashItem(Path as String, byref Resulting as FolderItem, byref error as NSErrorMBS) as boolean

Plugin Version: 19.5, Platform: macOS, Targets: All.

Function: Moves an item to the trash.

Notes: Returns true if the item at url was successfully moved to the trash. Since the operation may require renaming the item to avoid a file name collision, this method returns, by reference, the resulting folderitem that the item was moved to.

If this method returns false, the item was not moved to the trash; the error parameter then contains information about the error.

Available in OS X v10.8 and later.

See also:

- 18.1.94 trashItem(file as folderItem, byref Resulting as folderItem, byref error as NSErrorMBS) as boolean 1012

18.1.96 URLByAppendingPathComponent(URL as string, pathComponent as string) as string

Plugin Version: 12.1, 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.

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 Mac OS X v10.6 and later.

See also:

- 18.1.97 `URLByAppendingPathComponent(URL as string, pathComponent as string, isDirectory as boolean) as string` 1014

18.1.97 `URLByAppendingPathComponent(URL as string, pathComponent as string, isDirectory as boolean) as string`

Plugin Version: 12.1, 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 Mac OS X v10.7 and later.

See also:

- 18.1.96 `URLByAppendingPathComponent(URL as string, pathComponent as string) as string` 1013

18.1.98 `URLByAppendingPathExtension(URL as string, pathExtension as string) as string`

Plugin Version: 12.1, 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 Mac OS X v10.6 and later.

18.1.99 URLByDeletingLastPathComponent(URL as string) as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new URL made by deleting the last path component from the original URL.

Notes: If the original URL represents the root path, the returned URL is identical. Otherwise, if the original URL has only one path component, the new URL is the empty string.

Available in Mac OS X v10.6 and later.

18.1.100 URLByDeletingPathExtension(URL as string) as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new URL made by deleting the path extension, if any, from the original URL.

Notes: If the original URL represents the root path, the returned URL is identical. If the URL has multiple path extensions, only the last one is removed.

Available in Mac OS X v10.6 and later.

18.1.101 URLByResolvingSymlinksInPath(URL as string) as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new URL that points to the same resource as the original URL and includes no symbolic links.

Notes: If the original URL has no symbolic links, the returned URL is identical to the original URL.

This method only works on URLs with the file: path scheme. This method will return an identical URL for all other URLs.

Available in Mac OS X v10.6 and later.

18.1.102 URLByStandardizingPath(URL as string) as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new URL that points to the same resource as the original URL and is an absolute path.

Notes: This method only works on URLs with the file: path scheme. This method will return an identical URL for all other URLs.

Available in Mac OS X v10.6 and later.

18.1.103 URLForPublishingUbiquitousItem(item as folderitem, byref expirationDate as date, byref error as NSErrorMBS) as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a URL that can be emailed to users to allow them to download a copy of a cloud-based item.

Example:

```
dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Desktop.Child("test.rtf")

dim error as NSErrorMBS
dim exdate as date
dim url as string = m.URLForPublishingUbiquitousItem(f, exdate, error)

// always fails as desktop folder is not in cloud
if error<>nil then
MsgBox error.localizedDescription
end if
```

Notes: url: Specify the URL of the item in the cloud that you want to share. The URL must be prefixed with the base URL returned from the URLForUbiquityContainerIdentifier method that corresponds to the item's location.

expirationDate: On output, this parameter contains the date after which the item is no longer available at the returned URL.

error: If an error occurs, this pointer is set to an NSError object containing information about the error.

Returns an URL with which users can download a copy of the item at url. Returns nil if the URL could not be created for any reason.

Discussion

This method creates a snapshot of the specified file and places that copy in a temporary iCloud location where it can be accessed by other users using the returned URL. The snapshot reflects the contents of the file at the time the URL was generated and is not updated when subsequent changes are made to the original file in the user's iCloud storage. The snapshot file remains available at the specified URL until the date specified in the outDate parameter, after which it is automatically deleted.

Your application must have access to the network for this call to succeed.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.104 URLForPublishingUbiquitousItem(URL as string, byref expirationDate as date, byref error as NSErrorMBS) as string 1018

18.1.104 `URLForPublishingUbiquitousItem(URL as string, byref expirationDate as date, byref error as NSErrorMBS) as string`

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

Function: Returns a URL that can be emailed to users to allow them to download a copy of a cloud-based item.

Notes: `url`: Specify the URL of the item in the cloud that you want to share. The URL must be prefixed with the base URL returned from the `URLForUbiquityContainerIdentifier` method that corresponds to the item's location.

`expirationDate`: On output, this parameter contains the date after which the item is no longer available at the returned URL.

`error`: If an error occurs, this pointer is set to an `NSError` object containing information about the error.

Returns an URL with which users can download a copy of the item at `url`. Returns `nil` if the URL could not be created for any reason.

Discussion

This method creates a snapshot of the specified file and places that copy in a temporary iCloud location where it can be accessed by other users using the returned URL. The snapshot reflects the contents of the file at the time the URL was generated and is not updated when subsequent changes are made to the original file in the user's iCloud storage. The snapshot file remains available at the specified URL until the date specified in the `outDate` parameter, after which it is automatically deleted.

Your application must have access to the network for this call to succeed.

Available in Mac OS X v10.7 and later.

See also:

- 18.1.103 `URLForPublishingUbiquitousItem(item as folderitem, byref expirationDate as date, byref error as NSErrorMBS) as string` 1017

18.1.105 `URLForUbiquityContainerIdentifier(containerIdentifier as string) as string`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the iCloud directory associated with the specified container ID.

Notes: `containerID`: Specify the container ID of the cloud-based storage container. The string you specify must not contain wildcards and must be of the form `<TEAMID>.<CONTAINER>`, where `<TEAMID>` is your development team ID and `<CONTAINER>` describes the bundle identifier of the container you want to access. The container identifiers for your application must be declared in the `com.apple.developer.ubiquity-container-identifiers` entitlement.

If you specify `""`, this method returns the first container listed in the `com.apple.developer.ubiquity-container-identifiers` entitlement.

Returns a folderitem pointing to the specified container directory or nil if the container could not be located or if iCloud storage is unavailable for the current user or device.

You can use the folderitem returned by this method to build paths to files and directories in the user's iCloud storage. Each application that syncs documents to the cloud must have at least one associated container directory in which to put those files. This container directory can be unique to the application or shared by multiple applications. You use this method to retrieve the folderitem for that container directory.

In addition to writing to its own container directory, an application can write to any container directory for which it has the appropriate permission. Each additional container directory should be listed as an additional value in the com.apple.developer.ubiquity-container-identifiers entitlement.

Note: The development team ID that precedes each container ID string is the unique identifier associated with your development team. You can find this string in the Member Center of the Apple Developer website (<http://developer.apple.com/membercenter>). From the Member Center home page, select the Your Account tab and then select Organization Profile from the column on the left of that tab. Your team's identifier is in the Company/Organization ID field.

The first time you call this method for a given container directory, iOS extends your application sandbox to include that container directory. Thus, it is important that you call this method at least once before trying to search for files in iCloud. And if your application accesses multiple container directories, you should call the method once for each directory.

Available in Mac OS X v10.7 and later.

FileForUbiquityContainerIdentifier returns folderitem while URLForUbiquityContainerIdentifier returns URL string.

18.1.106 Properties

18.1.107 currentDirectory as folderitem

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns the path of the program's current directory.

Example:

```
dim m as new NSFileManagerMBS
dim f as FolderItem = SpecialFolder.Applications
if m.changeCurrentDirectory(f) then
MsgBox m.currentDirectory.NativePath
end if
```

Notes: Returns the path of the program's current directory. If the program's current working directory

isn't accessible, returns nil.

The string returned by this method is initialized to the current working directory; you can change the working directory by invoking `changeCurrentDirectoryPath`.

Relative pathnames refer implicitly to the current directory. For example, if the current directory is `/tmp`, and the relative pathname `reports/info.txt` is specified, the resulting full pathname is `/tmp/reports/info.txt`. (Read only property)

18.1.108 `currentDirectoryPath` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns the path of the program's current directory as string.

Example:

```
dim m as new NSFileManagerMBS
MsgBox m.currentDirectoryPath
```

Notes: Returns the path of the program's current directory. If the program's current working directory isn't accessible, returns "".

The string returned by this method is initialized to the current working directory; you can change the working directory by invoking `changeCurrentDirectoryPath`.

Relative pathnames refer implicitly to the current directory. For example, if the current directory is `/tmp`, and the relative pathname `reports/info.txt` is specified, the resulting full pathname is `/tmp/reports/info.txt`. (Read only property)

18.1.109 `defaultManager` as `NSFileManagerMBS`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The default manager.

Notes: This is just one global property, so we only create instance on first call and then return the same instance again.

(Read only property)

18.1.110 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

18.1.111 homeDirectoryForCurrentUser as FolderItem

Plugin Version: 18.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the home directory for the current user.

Notes: (Read only property)

18.1.112 temporaryDirectory as FolderItem

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the temporary directory for the current user.

Notes: (Read only property)

18.1.113 Constants

Constants

Constant	Value	Description
<code>NSDirectoryEnumerationSkipsHiddenFiles</code>	4	One of the option constants for enumerating the contents of the <code>directoryEnumerator</code> method. Do not enumerate hidden files. Available in Mac OS X v10.6 and later.
<code>NSDirectoryEnumerationSkipsPackageDescendants</code>	2	One of the option constants for enumerating the contents of the <code>directoryEnumerator</code> method. Do not descend into packages. Available in Mac OS X v10.6 and later.
<code>NSDirectoryEnumerationSkipsSubdirectoryDescendants</code>	1	One of the option constants for enumerating the contents of the <code>directoryEnumerator</code> method. Perform a shallow enumeration; do not descend into subdirectories. Available in Mac OS X v10.6 and later.
<code>NSFileManagerItemReplacementUsingNewMetadataOnly</code>	1	One of the options constants to specify the replacement behavior of the <code>replaceItemAtURLWithOptions:error:options:completionHandler:</code> method. Causes <code>NSFileManagerItemReplacementWithoutDeletingBackupItem</code> to not delete the backup item. Available in Mac OS X v10.6 and later.
<code>NSFileManagerItemReplacementWithoutDeletingBackupItem</code>	2	One of the options constants to specify the replacement behavior of the <code>replaceItemAtURLWithOptions:error:options:completionHandler:</code> method. Causes <code>NSFileManagerItemReplacementWithoutDeletingBackupItem</code> to not delete the backup item in place after a successful replacement. Available in Mac OS X v10.6 and later.
<code>NSVolumeEnumerationProduceFileReferenceURLs</code>	4	One of the option constants for enumerating mounted volumes. The enumeration produces file reference URLs rather than volume names. Available in Mac OS X v10.6 and later.
<code>NSVolumeEnumerationSkipHiddenVolumes</code>	2	One of the option constants for enumerating mounted volumes. The enumeration skips hidden volumes. Available in Mac OS X v10.6 and later.

Unmount options

Constant	Value	Description
<code>NSFileManagerUnmountAllPartitionsAndEjectDisk</code>	1	Unmount all partitions of this disk and eject it.
<code>NSFileManagerUnmountWithoutUI</code>	2	Unmount without user interface.

Relationships

Constant	Value	Description
NSURLRelationshipContains	0	The directory contains the specified item.
NSURLRelationshipOther	2	The directory does not contain the item and is not the same as the item.
NSURLRelationshipSame	1	The directory and the item are the same. This relationship occurs when the value of the NSURLFileResourceIdentifierKey is the same for the directory and item.

Chapter 19

Instant Message

19.1 class IMServiceMBS

19.1.1 class IMServiceMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** An IMService object represents a service available to a user through iChat, such as AIM, Jabber, and Bonjour.

Notes: Each IMService object represents one service available through iChat. Class methods such as allServices and serviceWithName will return these objects. Each object acts as the liaison to its single service, allowing you to access the individual user's global status, the user's list of acquaintances, and other information which can be integrated into your application.

You may want to subclass the InstantMessageMBS class to get events for changes.

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.

Blog Entries

- [Cleanup Xojo Plugins](#)
- [MBS Xojo Plugins, version 20.5pr8](#)

19.1.2 Methods

19.1.3 `imageFileForStatus(status as Integer)` as folderitem

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the folderitem of the image corresponding to the `IMPersonStatus` specified by `status`.

Notes: Convenience function which does the same as `InstantMessageMBS.imageFileForStatus`.

19.1.4 `imageNameForStatus(status as Integer)` as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the image name for the given status.

Example:

```
MsgBox IMServiceMBS.imageNameForStatus(IMServiceMBS.IMPersonStatusIdle)
```

Notes: Mac OS X 10.5 only.

On Mac OS X the `NSImage` class can be used to access system images by name and there you can use this name.

19.1.5 `imageURLForStatus(status as Integer)` as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the URL of the image corresponding to the `IMPersonStatus` specified by `status`.

Notes: Convenience function which does the same as `InstantMessageMBS.imageURLForStatus`.

19.1.6 `IMCapabilityAudioConference` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

19.1.7 `IMCapabilityDirectIM` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

19.1.8 IMCapabilityFileSharing as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

19.1.9 IMCapabilityFileTransfer as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

19.1.10 IMCapabilityText as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

19.1.11 IMCapabilityVideoConference as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

19.1.12 IMPersonAVBusyKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's busy status. The value is a number set to 0 if the person's audio/video capabilities are available, or 1 if they are busy.

19.1.13 IMPersonCapabilitiesKey as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's iChat capabilities. The value is an array of capability properties. Check for `IMCapability*` strings in this array.

19.1.14 `IMPersonEmailKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's email address. The value is a string containing the person's email address. This is a key used directly by Bonjour; however, if a person has an Address Book entry associated with a relevant AIM account, this key reflects the first email address of that person.

19.1.15 `IMPersonFirstNameKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's first name. The value is a string containing the person's first name. This is a key used directly by Bonjour; however, if a person has an Address Book entry associated with a relevant AIM account, this key reflects the first name of that person.

19.1.16 `IMPersonIdleSinceKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's idle status. The value is a date containing the time, in seconds, since the last user activity. Available if the person's status is idle.

19.1.17 `IMPersonLastNameKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's last name. The value is a string containing the person's last name. This is a key used directly by Bonjour; however, if a person has an Address Book entry associated with a relevant AIM account, this key reflects the last name of that person.

19.1.18 `IMPersonPictureDataKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's image. The value is a string containing the image for the person's icon.

19.1.19 `IMPersonScreenNameKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's screen name. The value is a string containing the service-specific identifier for a person. For example, "User123" or "steve@mac.com" for AIM, and "John Doe" for Bonjour.

19.1.20 `IMPersonServiceNameKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's service name. The value is a string containing the name of the service this person belongs to.

19.1.21 `IMPersonStatusKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's online status. The value is a number representing the current online status of the person, if known.

19.1.22 `IMPersonStatusMessageKey` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: One of the keys for the info dictionaries.

Notes: Used to obtain a person's status message. The value is a string containing the person's current status message.

19.1.23 infoForAllScreenNames as dictionary()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns information about all people and all accounts currently logged in to the service.

Notes: If a person is logged in on multiple accounts (determined by the user's Address Book), this method will return the information for all of the logged-in accounts.

Returns an Array of the dictionaries. Use IMPerson*Key strings for the keys in that dictionary.

19.1.24 infoForPreferredScreenNames as dictionary()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns information about all people and their primary accounts currently logged in to the service.

Notes: If a person is logged in on multiple accounts (determined by the user's Address Book), this method will only return the information for the preferred account. The preferred account is determined by iChat, using a combination of capabilities (video chat capability, audio chat capability, and so on), status (available, idle, away), and other user attributes.

Returns an Array of the dictionaries for all people and is guaranteed to provide only one array entry for any logged-in person. Use IMPerson*Key strings for the keys in that dictionary.

19.1.25 infoForScreenName(name as string) as dictionary

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns information about the person specified by his/her screenName.

Notes: screenName: A string containing the screen name identifier of a person.

Returns a dictionary on success and nil on failure. Use IMPerson*Key strings for the keys in that dictionary.

19.1.26 LocalizedName as String

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the user-visible localized name of the service.

Example:

```
dim services() as IMServiceMBS = InstantMessageMBS.allServices
```

```
for each service as IMServiceMBS in Services
MsgBox service.LocalizedName
next
```

Notes: Returns a String. Will contain the localized service name, such as "AOL Instant Messenger", "Jabber", or "Bonjour", for example. This string will be localized if required.

19.1.27 LocalizedShortName as String

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns a shorter version, if available, of the user-visible localized name of the service.

Example:

```
dim services() as IMServiceMBS = InstantMessageMBS.allServices

for each service as IMServiceMBS in Services
MsgBox service.LocalizedShortName
next
```

Notes: Returns a "" on failure. Will return a localized string if required.

19.1.28 Name as String

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the fixed canonical name of the service.

Example:

```
dim services() as IMServiceMBS = InstantMessageMBS.allServices

for each service as IMServiceMBS in Services
MsgBox service.Name
next
```

Notes: Returns a "" on failure. This string is not localized.

19.1.29 peopleWithScreenName(screenName as string) as ABPersonMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns the Address Book person objects that correspond to the person with the screenName matched by screenName.

Notes: screenName: An string containing the screen name identifier of a person or persons.

Returns an array of ABPersonMBS objects that match the screen name matched by screenName. Can return an empty array or an array with one or more items.

Returns empty array on failure.

19.1.30 screenNamesForPerson(person as ABPersonMBS) as string()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns a list of valid screen names for any given person.

Example:

```
// show my screennames on all services
```

```
dim AddressBook as new ABAddressBookMBS
dim owner as ABPersonMBS = AddressBook.owner
dim services() as IMServiceMBS = InstantMessageMBS.allServices

for each service as IMServiceMBS in Services
MsgBox service.LocalizedName+":"+EndOfLine+join(service.screenNamesForPerson(owner), EndOfLine)
next
```

Notes: person: An Address Book ABPerson object.

Returns an Array of Strings that are valid screen names for the person specified by person. See Address Book documentation for more information on ABPerson and accessing the user's address book. Can return an empty array or an array with one or more items.

Returns an empty array on failure.

19.1.31 Status as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the login status of the service.

Notes: Returns the appropriate IMServiceStatus number.

19.1.32 Properties

19.1.33 Handle as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Handle to the internal used IMService reference.

Notes: (Read and Write property)

19.1.34 Constants

Constants

Constant	Value	Description
IMPersonStatusAvailable	4	A status constant.
IMPersonStatusAway	3	A status constant.
IMPersonStatusIdle	2	A status constant.
IMPersonStatusNoStatus	5	The status constant for persons where the status is unknown. Mac OS X 10.5 only.
IMPersonStatusOffline	1	A status constant.
IMPersonStatusUnknown	0	A status constant.
IMServiceStatusDisconnected	1	A status constant.
IMServiceStatusLoggedIn	4	A status constant.
IMServiceStatusLoggedOut	0	A status constant.
IMServiceStatusLoggingIn	3	A status constant.
IMServiceStatusLoggingOut	2	A status constant.

19.2 class InstantMessageMBS

19.2.1 class InstantMessageMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: The class to handle Instant Message support for Mac OS X 10.4 and newer.

Notes: This class is based on the Instant Message framework from Apple.

it can be used to get the status of iChat.

Blog Entries

- [Addressbook classes updated](#)

19.2.2 Methods

19.2.3 allServices as IMServiceMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns the list of services currently available to the user, regardless of their status.

Notes: Returns nil on failure.

Returns a array with IMServiceMBS objects corresponding to the current available services (AIM, Bonjour, Jabber, and so on.)

19.2.4 Available as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Whether the InstantMessaging framework is available or not.

Notes: Should be true on Mac OS X 10.4.

19.2.5 imageFileForStatus(status as Integer) as folderitem

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the folderitem of the image corresponding to the IMPersonStatus specified by status.

Notes: This image reflects the status of the user, and is usually reflected by a colored bubble or triangle.

Returns nil on failure.

This is a convenience function which will call imageFileForStatus and return the folderitem matching the

URL in case it points to a disc file.

For Mac OS X 10.4 all images are stored as TIFF files on hard disc. Use the TiffPictureMBS class to load them (OpenAsPicture will not work because of the masks).

values:

```
IMPersonStatusUnknown    = 0
IMPersonStatusOffline    = 1
IMPersonStatusIdle       = 2
IMPersonStatusAway       = 3
IMPersonStatusAvailable  = 4
```

19.2.6 imageNameForStatus(status as Integer) as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the image name for the given status.

Notes: Mac OS X 10.5 only.

On Mac OS X the NSImage class can be used to access system images by name and there you can use this name.

19.2.7 imageURLForStatus(status as Integer) as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns the URL of the image corresponding to the IMPersonStatus specified by status.

Notes: This image reflects the status of the user, and is usually reflected by a colored bubble or triangle.

Returns nil on failure.

values:

```
IMPersonStatusUnknown    = 0
IMPersonStatusOffline    = 1
IMPersonStatusIdle       = 2
IMPersonStatusAway       = 3
IMPersonStatusAvailable  = 4
```

19.2.8 myIdleTime as Double

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Class method to return the idle time of the active user.

Notes: Returns in seconds the time since the currently active user went idle.

19.2.9 myStatus as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Class method to return the status of the currently active user.

Notes: This status is global across all services.

values:

```

IMPersonStatusUnknown    = 0
IMPersonStatusOffline    = 1
IMPersonStatusIdle       = 2
IMPersonStatusAway       = 3
IMPersonStatusAvailable  = 4

```

19.2.10 notificationCenter as NSNotificationCenterMBS

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

Function: Returns the custom notification center for the service.

Notes: A custom notification center that manages IMService notifications.

Available in Mac OS X v10.4 and later.

19.2.11 serviceWithName(name as string) as IMServiceMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: Returns only the service specified by a given name.

Notes: name: A String containing a service name as returned by a previous call to name.

Hard-coding the service names internally is not recommended.

Returns an IMService object corresponding to the available service specified by name.

Returns nil on any failure.

19.2.12 Events

19.2.13 MyStatusChanged

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: This event is called when the local user changes online status.

Notes: The client should call myStatus function to get the new status.

19.2.14 PersonInfoChanged(info as dictionary)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Event sent when a screenName changes some aspect of their published information.

Notes: The user information dictionary will always contain an ScreenName and may contain any of the following values: StatusMessage, IdleSince, FirstName, LastName, IMPersonEmailKey, PictureData, AVAvailable and AVBusy, Capabilities values.

19.2.15 PersonStatusChanged(info as dictionary)

Plugin Version: 11.2, Platform: macOS, Targets: .

Function: Event sent when a different user (screenName) logs in, logs off, goes away, and so on.

Notes: The info object knows the ScreenName and the Status of the person.

19.2.16 ServiceStatusChanged

Plugin Version: 7.1, Platform: macOS, Targets: .

Function: Event sent when the user logs in, logs off, goes away, and so on.

Notes: Call MyStatus to get the new state.

19.2.17 StatusImagesChangedAppearance

Plugin Version: 7.1, Platform: macOS, Targets: .

Function: Event sent when the user changes their preferred images for displaying status.

Notes: Clients that display status information graphically (using the green/yellow/red dots) should call imageURLForStatus to get the new image.

Chapter 20

Linguistic

20.1 class NSLinguisticTaggerMBS

20.1.1 class NSLinguisticTaggerMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Analyze natural language to tag part of speech and lexical class, identify proper names, perform lemmatization, and determine the language and script (orthography) of text.

Example:

```
dim TagScheme as string = NSLinguisticTaggerMBS.NSLinguisticTagSchemeLanguage
dim TagSchemes() as string = array(TagScheme)
dim t as new NSLinguisticTaggerMBS(TagSchemes)
```

```
t.Text = "Hallo Leute"
```

```
dim tokenRange as NSRangeMBS
dim sentenceRange as NSRangeMBS
dim tag as string = t.tagAtIndex(0, TagScheme, tokenRange, sentenceRange)
```

```
MsgBox "Language: "+tag // should be "de" for German
```

Notes: The NSLinguisticTaggerMBS class provides a uniform interface to a variety of natural language processing functionality with support for many different languages and scripts. You can use NSLinguisticTaggerMBS to segment natural language text into paragraphs, sentences, or words, and tag information about those tokens, such as part of speech, lexical class, lemma, script, and language.

When you create a linguistic tagger, you specify what kind of information you're interested in by passing one or more

NSLinguisticTagScheme values. Set the string property to the natural language text you want to analyze,

and the linguistic tagger processes it according to the specified tag schemes. You can then enumerate over the tags in a specified range, using the methods described in Enumerating Linguistic Tags, to get the information requested for a given scheme and unit.

Blog Entries

- [MBS Xojo Plugins, version 21.1pr6](#)
- [MBS Xojo Plugins, version 17.3pr3](#)

20.1.2 Methods

20.1.3 `availableTagSchemesForLanguage(Language as String) as String()`

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the tag schemes available for a particular language on the current device.

Notes: language: A BCP-47 tag identifying the language. For example, "en" for English or "zh-Hans" for Chinese written using the Simplified Chinese script.

Returns the available tag schemes. For possible values, see `NSLinguisticTagScheme`.

This is a convenience method for calling the `availableTagSchemesForUnit`, passing `NSLinguisticTaggerUnit-Word` as the linguistic unit.

20.1.4 `availableTagSchemesForUnit(unit as Integer, Language as String) as String()`

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the tag schemes available for a particular unit and language on the current device.

Notes: unit: The linguistic unit. For possible values, see `NSLinguisticTaggerUnit`.

language: A BCP-47 tag identifying the language. For example, "en" for English or "zh-Hans" for Chinese written using the Simplified Chinese script.

Returns the supported tag schemes. For possible values, see `NSLinguisticTagScheme`.

Available in macOS 10.13 and newer.

20.1.5 `Constructor(tagSchemes() as String, options as integer = 0)`

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Creates a linguistic tagger instance using the specified tag schemes and options.

Notes: tagSchemes: An array of tag schemes. See NSLinguisticTagScheme for the possible values.
options: Reserved for future use. Specify 0 for this parameter.

20.1.6 dominantLanguageForString(text as String) as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the dominant language for the specified string.

Notes: string: The string for which the dominant language is determined.

Returns the BCP-47 tag identifying the dominant language of the string, or the tag "und" if a specific language cannot be determined.

This is a convenience method for creating a new linguistic tagger, setting the string property, and getting the dominantLanguage property. If you analyze the same string more than once, you should create a linguistic tagger object instead of calling this method.

Available in macOS 10.13 and newer.

20.1.7 NSLinguisticTagAdjective as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is an adjective.

20.1.8 NSLinguisticTagAdverb as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is an adverb.

20.1.9 NSLinguisticTagClassifier as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a classifier.

20.1.10 NSLinguisticTagCloseParenthesis as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a close parenthesis.

20.1.11 NSLinguisticTagCloseQuote as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a close quote.

20.1.12 NSLinguisticTagConjunction as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a conjunction.

20.1.13 NSLinguisticTagDash as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a dash.

20.1.14 NSLinguisticTagDeterminer as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a determiner.

20.1.15 NSLinguisticTagIdiom as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is an idiom.

20.1.16 NSLinguisticTagInterjection as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is an interjection.

20.1.17 NSLinguisticTagNoun as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: The token is a noun.

20.1.18 NSLinguisticTagNumber as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a number.

20.1.19 NSLinguisticTagOpenParenthesis as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is an open parenthesis.

20.1.20 NSLinguisticTagOpenQuote as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is an open quote.

20.1.21 `NSLinguisticTagOrganizationName` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the Name Types.

Notes: This token is an organization name.

20.1.22 `NSLinguisticTagOther` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the token types.

Notes: The token indicates a non-linguistic item, such as a symbol.

20.1.23 `NSLinguisticTagOtherPunctuation` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is punctuation other than a kind described by other lexical classes (sentence terminator, open or close quote, open or close parenthesis, word joiner, and dash).

20.1.24 `NSLinguisticTagOtherWhitespace` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is whitespace other than a kind described by other lexical classes (paragraph break).

20.1.25 `NSLinguisticTagOtherWord` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a word other than a kind described by other lexical classes (noun, verb, adjective, adverb, pronoun, determiner, particle, preposition, number, conjunction, interjection, classifier, and idiom).

20.1.26 NSLinguisticTagParagraphBreak as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a paragraph break.

20.1.27 NSLinguisticTagParticle as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a particle.

20.1.28 NSLinguisticTagPersonalName as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the Name Types.

Notes: This token is a personal name.

20.1.29 NSLinguisticTagPlaceName as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the Name Types.

Notes: This token is a place name.

20.1.30 NSLinguisticTagPreposition as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a preposition.

20.1.31 NSLinguisticTagPronoun as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a pronoun.

20.1.32 `NSLinguisticTagPunctuation` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the token types.

Notes: The token indicates punctuation.

20.1.33 `NSLinguisticTagSchemeLanguage` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Supplies a BCP-47 language identifier for a token.

Notes: For example, the language identifier for English is "en" and the identifier for Chinese written using the Simplified Chinese script is "zh-Hans". The identifier "und" is used if a specific language cannot be determined.

The tagger generally attempts to determine the language of text at the level of an entire sentence, paragraph, or document, rather than word by word.

20.1.34 `NSLinguisticTagSchemeLemma` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Supplies a stem form of a word token, if known.

20.1.35 `NSLinguisticTagSchemeLexicalClass` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Classifies tokens according to class: part of speech for words, type of punctuation, or whitespace.

20.1.36 `NSLinguisticTagSchemeNameType` as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Classifies tokens according to whether they are part of a named entity.

20.1.37 NSLinguisticTagSchemeNameTypeOrLexicalClass as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Classifies tokens corresponding to names according to NSLinguisticTagSchemeNameType and tokens all other tokens according to NSLinguisticTagSchemeLexicalClass.

20.1.38 NSLinguisticTagSchemeScript as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Supplies an ISO 15924 script identifier for a token.

Notes: For example, the identifier for Latin script is "Latn" and the identifier for Simplified Chinese script is "Hans". The identifier "Zyyy" is used if a specific script cannot be determined.

20.1.39 NSLinguisticTagSchemeTokenType as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Classifies tokens according to their broad type: word, punctuation, or whitespace.

Notes: To classify tokens by a more specific type, for example, distinguishing words between nouns and verbs, use the NSLinguisticTagSchemeLexicalClass scheme.

20.1.40 NSLinguisticTagSentenceTerminator as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a sentence terminator.

20.1.41 NSLinguisticTagVerb as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a verb.

20.1.42 NSLinguisticTagWhitespace as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the token types.

Notes: The token indicates white space of any sort.

20.1.43 NSLinguisticTagWord as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the token types.

Notes: The token indicates a word.

20.1.44 NSLinguisticTagWordJoiner as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the lexical classes.

Notes: This token is a word joiner.

20.1.45 orthographyAtIndex(charIndex as integer, byref effectiveRange as NSRangeMBS) as NSOrthographyMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the orthography at the index and also returns the effective range.

Example:

```
dim TagScheme as string = NSLinguisticTaggerMBS.NSLinguisticTagSchemeLanguage
dim TagSchemes() as string = array(TagScheme)
dim t as new NSLinguisticTaggerMBS(TagSchemes)
```

```
t.Text = "Hallo Leute"
```

```
dim tokenRange as NSRangeMBS
dim sentenceRange as NSRangeMBS
dim tag as string = t.tagAtIndex(0, TagScheme, tokenRange, sentenceRange)
```

```
dim effectiveRange as NSRangeMBS
dim o as NSOrthographyMBS = t.orthographyAtIndex(0, effectiveRange)
```

```
MsgBox o.dominantLanguage+" "+o.dominantScript
```

Notes: charIndex: The character index to begin examination.
effectiveRange: An NSRangeMBS that, upon completion, contains the range of the orthography containing charIndex.

Returns the orthography for the location.

20.1.46 sentenceRangeForRange(range as NSRangeMBS) as NSRangeMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the range of a sentence containing the specified range.

Notes: charRange: The character range.

Returns the range of the sentence.

This is a convenience method for calling tokenRangeAtIndex, passing the NSLinguisticTaggerUnitSentence unit and the first position of the provided range.

20.1.47 setOrthography(orthography as NSOrthographyMBS, range as NSRangeMBS)

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Sets the orthography for the specified range.

Notes: orthography: The orthography.

charRange: The range.

If the orthography of the linguistic tagger is not set, it will determine it automatically from the contents of the text. You should call this method only if you know the orthography of the text by some other means.

20.1.48 tagAtIndex(charIndex as Integer, Scheme as String, byref tokenRange as NSRangeMBS, byref sentenceRange as NSRangeMBS) as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns a tag for a single scheme at the specified character position.

Notes: charIndex: The position of the initial character.

tagScheme: The tag scheme. See NSLinguisticTagScheme for the possible values.

tokenRange: The token range for output.

sentenceRange: The range of the sentence for output.

Returns the tag for the requested tag scheme, or nil. If a tag is returned, this function returns by reference the range of the token to tokenRange, and the range of the enclosing sentence to sentenceRange, if applicable.

This is a convenience method for calling tagAtIndex and passing NSLinguisticTaggerUnitWord as the linguistic unit.

Available in macOS 10.7 and newer.

See also:

- 20.1.49 tagAtIndex(charIndex as Integer, unit as Integer, Scheme as String, byref tokenRange as NSRangeMBS) as String 1050

20.1.49 tagAtIndex(charIndex as Integer, unit as Integer, Scheme as String, byref tokenRange as NSRangeMBS) as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns a tag for a single scheme for a given linguistic unit at the specified character position.

Notes: charIndex: The position of the initial character.

unit: The linguistic unit. See NSLinguisticTaggerUnit for possible values.

tagScheme: The tag scheme. See NSLinguisticTagScheme for possible values.

tokenRange: The token range for output.

Returns the tag for the requested tag scheme and linguistic unit, or nil. If a tag is returned, this function returns by reference the range of the token to tokenRange.

Available in macOS 10.13 and newer.

See also:

- 20.1.48 tagAtIndex(charIndex as Integer, Scheme as String, byref tokenRange as NSRangeMBS, byref sentenceRange as NSRangeMBS) as String 1049

20.1.50 tagForString(text as string, charIndex as Integer, unit as Integer, Scheme as String, orthography as NSOrthographyMBS, byref tokenRange as NSRangeMBS) as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns a tag for a single scheme for a given linguistic unit at the specified character position in a string.

Notes: charIndex: The position of the initial character.

unit: The linguistic unit. See `NSLinguisticTaggerUnit` for possible values.
 tagScheme: The tag scheme. See `NSLinguisticTagScheme` for possible values.
 tokenRange: The token range for output.

Returns the tag for the requested tag scheme and linguistic unit, or nil. If a tag is returned, this function returns by reference the range of the token to tokenRange.

This is a convenience method for initializing a linguistic tagger, setting the string property, and calling the `tagForString` method. If you analyze the same string more than once, you should create a linguistic tagger object instead of calling this method.

20.1.51 tagSchemes as String()

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the tag schemes configured for this linguistic tagger.

Notes: For possible values, see `NSLinguisticTagScheme`.

20.1.52 TagsForString(text as string, range as NSRangeMBS, unit as Integer, Scheme as String, options as Integer, orthography as NSOrthographyMBS) as NSLinguisticValueMBS()

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns an array of linguistic tags and token ranges.

Notes: range: The range from which to return tags.

unit: The linguistic unit. See `NSLinguisticTaggerUnit` for possible values.

tagScheme: The tag scheme. See `NSLinguisticTagScheme` for possible values.

options: The linguistic tagger options to use. See `NSLinguisticTaggerOptions` for possible values.

An array of the values in the requested range (tag and tokenRange set).

When the returned array contains an entry that doesn't have a corresponding tag scheme, that entry is an empty string ("").

This is a convenience method for initializing a linguistic tagger, setting the string property, and calling the `tagsInRange` method. If you analyze the same string more than once, you should create a linguistic tagger object instead of calling this method.

Available in macOS 10.13 and newer.

20.1.53 tagsInRange(range as NSRangeMBS, Scheme as String, options as Integer) as NSLinguisticValueMBS()

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns an array of linguistic tags and token ranges.

Notes: range: The range to analyze.

tagScheme: The tag scheme. For possible values, see NSLinguisticTagScheme.

options: The linguistic tagger options to use. See NSLinguisticTaggerOptions for possible values.

Available in macOS 10.7 and newer.

See also:

- 20.1.54 tagsInRange(range as NSRangeMBS, unit as Integer, Scheme as String, options as Integer) as NSLinguisticValueMBS() 1052

20.1.54 tagsInRange(range as NSRangeMBS, unit as Integer, Scheme as String, options as Integer) as NSLinguisticValueMBS()

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns an array of linguistic tags and token ranges.

Notes: range: The range to analyze.

unit: The linguistic unit. For possible values, see NSLinguisticTaggerUnit.

tagScheme: The tag scheme. For possible values, see NSLinguisticTagScheme.

options: The linguistic tagger options to use. See NSLinguisticTaggerOptions for possible values.

Available in macOS 10.13 and newer.

Returns all tokens intersecting a given range, supplying tags and ranges. The tagger segments the string into sentences and tokens as necessary, and return those ranges along with a tag for any scheme in its array of tag schemes. For example, if the tag scheme is NSLinguisticTagSchemeLexicalClass, the tags specify the part of speech (for word tokens) or the type of whitespace or punctuation (for whitespace or punctuation tokens). If the tag scheme is NSLinguisticTagSchemeLemma, the tags specify the stem form of the word (if known) for each word token.

See also:

- 20.1.53 tagsInRange(range as NSRangeMBS, Scheme as String, options as Integer) as NSLinguisticValueMBS() 1052

20.1.55 tokenRangeAtIndex(charIndex as Integer, Unit as Integer) as NSRangeMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the range of the linguistic unit containing the specified character index.

Notes: charIndex: The character index to begin examination.

unit: The linguistic unit. For possible values, see NSLinguisticTaggerUnit.

Returns the range of the substring for the linguistic unit.

Available in macOS 10.13 and newer.

20.1.56 Properties

20.1.57 dominantLanguage as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the dominant language of the string set for the linguistic tagger.

Notes: The BCP-47 tag identifying the dominant language of the string, or the tag "und" if a specific language cannot be determined.

Available in macOS 10.13 and newer.

(Read only property)

20.1.58 Handle as Integer

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

20.1.59 Text as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The string being analyzed by the linguistic tagger.

Notes: (Read and Write property)

20.1.60 Constants

Options

Constant	Value	Description
<code>NSLinguisticTaggerJoinNames</code>	16	Typically, multiple-word names will be returned as multiple tokens, following the standard tokenization practice of the tagger. If this option is set, multiple-word names will be joined together and returned as a single token.
<code>NSLinguisticTaggerOmitOther</code>	8	Omit tokens of type <code>NSLinguisticTagOther</code> (non-linguistic items, such as symbols).
<code>NSLinguisticTaggerOmitPunctuation</code>	2	Omit tokens of type <code>NSLinguisticTagPunctuation</code> (all punctuation).
<code>NSLinguisticTaggerOmitWhitespace</code>	4	Omit tokens of type <code>NSLinguisticTagWhitespace</code> (whitespace of all sorts).
<code>NSLinguisticTaggerOmitWords</code>	1	Omit tokens of type <code>NSLinguisticTagWord</code> (items considered to be words).

Units

Constant	Value	Description
<code>NSLinguisticTaggerUnitDocument</code>	3	The document in its entirety.
<code>NSLinguisticTaggerUnitParagraph</code>	2	An individual paragraph.
<code>NSLinguisticTaggerUnitSentence</code>	1	An individual sentence.
<code>NSLinguisticTaggerUnitWord</code>	0	An individual word.

20.2 class NSLinguisticValueMBS

20.2.1 class NSLinguisticValueMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The class for linguistic tokens.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo Plugins, version 21.1pr6](#)
- [MBS Xojo Plugins, version 17.3pr3](#)

20.2.2 Methods

20.2.3 Constructor

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The private constructor.

20.2.4 Properties

20.2.5 sentenceRange as NSRangeMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The sentence range.

Notes: Only filled by tagAtIndex when used without unit.
(Read and Write property)

20.2.6 Tag as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The linguistic tag.

Notes: (Read and Write property)

20.2.7 Text as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The text for the token.

Notes: (Read and Write property)

20.2.8 tokenRange as NSRangeMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The token range.

Notes: (Read and Write property)

20.3 class NSOrthographyMBS

20.3.1 class NSOrthographyMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: A description of the linguistic content of a piece of text typically used for spelling and grammar checking.

Notes: An NSOrthography instance describes:

- Which scripts the text contains.
- A dominant language and possibly other languages for each of these scripts.
- A dominant script and language for the text as a whole.

Scripts are uniformly described by standard four-letter tags (Latn, Grek, Cyrl, etc.) with the supertags Jpan and Kore typically used for Japanese and Korean text, Hans and Hant for Chinese text; the tag Zyyy is used if a specific script cannot be identified. See Internationalization and Localization Guide for more information on internationalization.

Languages are uniformly described by BCP-47 tags , preferably in canonical form; the tag und is used if a specific language cannot be determined.

Blog Entries

- [MBS Xojo Plugins, version 21.1pr6](#)
- [MBS Xojo Plugins, version 17.3pr3](#)

20.3.2 Methods

20.3.3 allLanguages as String()

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns an array containing all the languages appearing in the values of the language map.

Notes: Available in macOS 10.6 and newer.

20.3.4 allScripts as String()

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns an array containing all the scripts appearing as keys in the language map.

Notes: Available in macOS 10.6 and newer.

20.3.5 Constructor(script as string, map as dictionary)

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Creates and returns an orthography instance with the specified dominant script and language map.

Notes: script: The dominant script.

map: A dictionary containing the language map.

Returns an initialized orthography object for the specified script and language map.

20.3.6 copy as NSOrthographyMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Create a copy of the object.

20.3.7 defaultOrthographyForLanguage(language as string) as NSOrthographyMBS

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Queries default orthography for a language.

Notes: Available in macOS 10.13 and newer.

20.3.8 dominantLanguageForScript(script as string) as String

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the dominant language for the specified script.

Notes: Available in macOS 10.6 and newer.

20.3.9 languagesForScript(script as string) as String()

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the list of languages for the specified script.

Notes: Available in macOS 10.6 and newer.

20.3.10 `orthographyWithDominantScript(script as string, map as dictionary)` as `NSOrthographyMBS`

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Creates and returns an orthography instance with the specified dominant script and language map.

Notes: `script`: The dominant script.

`map`: A dictionary containing the language map.

Returns an initialized orthography object for the specified script and language map.

20.3.11 Properties

20.3.12 `dominantLanguage` as `String`

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: Returns the first language in the list of languages for the dominant script.

Notes: Available in macOS 10.6 and newer.

(Read only property)

20.3.13 `dominantScript` as `String`

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The dominant script for the text.

Notes: The dominant script should be a script tag, such as `Latn`, `Cyrl`, etc.

Available in macOS 10.6 and newer.

(Read only property)

20.3.14 `Handle` as `Integer`

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

20.3.15 languageMap as Dictionary

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: A dictionary that map script tags to arrays of language tags.

Notes: The dictionary,Ãs keys are script tags (such as Latn, Cyrl, and so forth) and whose values are arrays of language tags (such as en, fr, de, etc.)

(Read only property)

Chapter 21

Navigation

21.1 class NSOpenPanelMBS

21.1.1 class NSOpenPanelMBS

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The class for the Cocoa Open Panel.

Notes: Subclass of the NSSavePanelMBS class.

Blog Entries

- [News from the MBS Xojo Plugins in Version 23.0](#)
- [MBS Xojo Plugins, version 22.6pr3](#)
- [Customized File Open and Save Dialogs for Windows](#)
- [MBS Xojo Plugins, version 18.1pr4](#)
- [New for Mac OS X 10.10 in MBS Xojo Plugins](#)
- [MBS Xojo / Real Studio Plugins, version 14.3pr2](#)

21.1.2 Methods

21.1.3 beginForDirectory(path as folderitem, name as string, filetype() as string)

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Presents a modeless Open panel.

Notes: path:

Directory whose files the panel displays. When nil, the directory is the same directory used in the previous

invocation of the panel; this is probably the best choice for most situations.

name:

Specifies a particular file in `absoluteDirectoryPath` that is selected when the Open panel is presented to the user. When `nil`, no file is initially selected.

fileTypes:

Array of file extensions and/or HFS file types. Specifies the files the panel allows the user to select. An empty array makes all files in `path` selectable by the user.

This method will later called the `savePanelDidEnd` event.

21.1.4 `beginSheetForDirectory(path as folderitem, name as string, filetypes() as string, targetWindow as DesktopWindow)`

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

Function: Presents a sheet Open panel on a given window.

Notes: `path:`

Directory whose files the panel displays. When `nil`, the directory is the same directory used in the previous invocation of the panel; this is probably the best choice for most situations.

name:

Specifies a particular file in `path` that is selected when the Open panel is presented to the user. When `""`, no file is initially selected.

filetypes:

Array of file extensions and/or HFS file types. Specifies the files the panel allows the user to select. An empty array makes all files in `absoluteDirectoryPath` selectable by the user.

targetWindow:

Window to open the sheet on.

This method will later called the `savePanelDidEnd` event.

Seems like on Mac OS X 10.4 the usage of sheets will raise `NSEExceptions` for missing methods in the `NSCarbonWindowFrame` class.

See also:

- 21.1.5 `beginSheetForDirectory(path as folderitem, name as string, filetypes() as string, targetWindow as window)`

21.1.5 beginSheetForDirectory(path as folderitem, name as string, filetypes() as string, targetWindow as window)

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Presents a sheet Open panel on a given window.

Notes: path:

Directory whose files the panel displays. When nil, the directory is the same directory used in the previous invocation of the panel; this is probably the best choice for most situations.

name:

Specifies a particular file in path that is selected when the Open panel is presented to the user. When "", no file is initially selected.

filetypes:

Array of file extensions and/or HFS file types. Specifies the files the panel allows the user to select. An empty array makes all files in absoluteDirectoryPath selectable by the user.

targetWindow:

Window to open the sheet on.

This method will later called the savePanelDidEnd event.

Seems like on Mac OS X 10.4 the usage of sheets will raise NSExceptions for missing methods in the NSCarbonWindowFrame class.

See also:

- 21.1.4 beginSheetForDirectory(path as folderitem, name as string, filetypes() as string, targetWindow as DesktopWindow) 1062

21.1.6 Constructor

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The constructor which initializes the panel with default values.

21.1.7 Files as FolderItem()

Plugin Version: 18.1, Platform: macOS, Targets: Desktop only.

Function: Queries all folderitems.

See also:

- 21.1.8 Files(index as UInt32) as folderitem

1064

21.1.8 Files(index as UInt32) as folderitem

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The array of files.

Notes: Index is zero based.

See also:

- 21.1.7 Files as FolderItem()

1063

21.1.9 runModalForDirectory(path as folderitem, name as string, filetypes as string) as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A convenience version of the runModalForDirectory function which passes one filetype instead of an array of file types.

Notes: See NSCancelButton and NSOkButton constants for return values.

See also:

- 21.1.10 runModalForDirectory(path as folderitem, name as string, filetypes() as string) as Integer 1064

21.1.10 runModalForDirectory(path as folderitem, name as string, filetypes() as string) as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Runs the panel as a modal dialog with the given filetypes.

Notes: path is the directory to use as a start point. Pass nil to use the last directory which has been used.

filetypes:

The filetypes you want to allow.

See NSCancelButton and NSOkButton constants for return values.

See also:

- 21.1.9 runModalForDirectory(path as folderitem, name as string, filetypes as string) as Integer 1064

21.1.11 runModalForTypes(filetypes as string) as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A convenience version of the runModalForTypes function which passes one filetype instead of an array of file types.

Example:

```
Dim dlg As New NSOpenPanelMBS
Dim iResult as Integer = dlg.runModalForTypes("jpg")
```

Notes: See NSCancelButton and NSOkButton constants for return values.
See also:

- 21.1.12 runModalForTypes(filetypes() as string) as Integer 1065

21.1.12 runModalForTypes(filetypes() as string) as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Runs the panel as a modal dialog with the given filetypes.

Example:

```
Dim dlg As New NSOpenPanelMBS
dim types() as string = array("jpg", "tif")
Dim iResult as Integer = dlg.runModalForTypes(types)
```

Notes: See NSCancelButton and NSOkButton constants for return values.
See also:

- 21.1.11 runModalForTypes(filetypes as string) as Integer 1064

21.1.13 URL(index as UInt32) as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The array of file URLs.

Notes: Index is zero based.

21.1.14 URLs as String()

Plugin Version: 18.1, Platform: macOS, Targets: Desktop only.

Function: Queries all URLs.

21.1.15 Properties

21.1.16 `AccessoryViewDisclosed` as Boolean

Plugin Version: 23.0, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the panel's accessory view is visible.

Notes: The value of this property is true when the accessory view is visible, and false when it isn't. Setting the value of this property programmatically changes the visibility of the accessory panel. If no accessory panel is present, setting this property does nothing.

(Read and Write property)

21.1.17 `allowsMultipleSelection` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the panel's browser allows the user to open multiple files (and directories) at a time.

Notes: (Read and Write property)

21.1.18 `canChooseDirectories` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the user can select directories in the pane's browser.

Notes: When a directory is selected, the OK button is enabled only if flag is true.

(Read and Write property)

21.1.19 `canChooseFiles` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the panel allows the user to choose files to open.

Notes: (Read and Write property)

21.1.20 `canDownloadUbiquitousContents` as Boolean

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: Controls how the receiver responds to ubiquitous documents that aren't yet fully downloaded locally.

Notes: If true, then the receiver will disallow opening non-local ubiquitous files. Also, if the user attempts

to select a non-local file, the receiver will trigger or reprioritize downloading for that file so that it can be opened as soon as possible. If false, then the receiver will allow the user to select and open non-local files, giving your application responsibility for downloading and reporting progress. The default value is true, except for applications linked against the 10.9 SDK or earlier that have adopted iCloud by specifying a ubiquitous container identifier entitlement.

To provide the ideal user experience, you should set this property to NO and download the file's contents (with NSFileCoordinator) and show downloading progress (with NSProgress or NSMetadataQuery) in the context of your application, instead of relying on the open panel to do it.

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

21.1.21 canResolveUbiquitousConflicts as Boolean

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: Controls how the receiver responds to ubiquitous documents with conflicting versions.

Notes: If true, then when the user attempts to open one or more documents with conflicts, the receiver will first display conflict resolution UI, requiring the user to resolve those conflicts before the documents can be opened. If false, then the receiver does nothing in response to conflicts, allowing your application to handle them. The default value is true, except for applications linked against the 10.9 SDK or earlier that have adopted iCloud by specifying a ubiquitous container identifier entitlement.

To provide the ideal user experience, you should set this property to NO and do conflict detection and resolution (using NSURLUbiquitousItemHasUnresolvedConflictsKey and NSFileVersion) in the context of your application, instead of relying on the open panel to do it.

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

21.1.22 FilesCount as UInt32

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The number of selected files.

Notes: (Read only property)

21.1.23 resolvesAliases as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver resolves aliases.

Notes: If true, the effect is that dropping an alias on the panel or asking for filenames returns the resolved aliases. The default is true.

(Read and Write property)

21.2 class NSSavePanelMBS

21.2.1 class NSSavePanelMBS

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The class for a Cocoa Save Panel.

Example:

```
dim n as new NSSavePanelMBS

dim e as integer = n.runModal

if e = n.NSCancelButton then
// cancelled
Break
elseif e = n.NSOKButton then
// got file
dim file as FolderItem = n.file
Break
end if
```

Notes: Subclass of the NSPanelMBS class.

Blog Entries

- [News from the MBS Xojo Plugins in Version 23.0](#)
- [MBS Xojo Plugins, version 22.6pr3](#)
- [MBS Xojo Plugins, version 20.1pr5](#)
- [Customized File Open and Save Dialogs for Windows](#)
- [MBS Xojo Plugins, version 19.4pr1](#)
- [MBS Xojo / Real Studio Plugins, version 14.3pr2](#)
- [MBS Real Studio Plugins, version 12.5pr8](#)
- [Sandboxing](#)
- [MBS Plugins 11.1 Release notes](#)
- [MonkeyBread Software Releases the MBS Plugins 8.2](#)

Xojo Developer Magazine

- [18.3, page 52: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes](#)

21.2.2 Methods

21.2.3 `allowedFileTypes` as `string()`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns an array of the allowed file types.

Notes: If the user specifies a file whose type is in the array of allowed types, the user is not presented with another dialog (see `allowsOtherFileTypes` for details about this dialog) when trying to save. Examples of common file types are "rtf", "tiff", and "ps". File type strings encoding HFS file types are not valid values for this attribute. A nil return value, which is the default, indicates that the user can save to any ASCII file.

Available in Mac OS X v10.3 and later.

21.2.4 `beginSheetForDirectory(path as folderitem, name as string, targetWindow as DesktopWindow)`

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

Function: Presents a Save panel as a sheet with the directory specified by path and optionally, the file specified by name selected.

Notes: If targetWindow is nil, the panel will be a modal dialog.

This method will later called the `savePanelDidEnd` event.

Seems like on Mac OS X 10.4 the usage of sheets will raise `NSEExceptions` for missing methods in the `NSCarbonWindowFrame` class.

See also:

- 21.2.5 `beginSheetForDirectory(path as folderitem, name as string, targetWindow as window)` 1070

21.2.5 `beginSheetForDirectory(path as folderitem, name as string, targetWindow as window)`

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Presents a Save panel as a sheet with the directory specified by path and optionally, the file specified by name selected.

Notes: If targetWindow is nil, the panel will be a modal dialog.

This method will later called the `savePanelDidEnd` event.

Seems like on Mac OS X 10.4 the usage of sheets will raise `NSEExceptions` for missing methods in the `NSCarbonWindowFrame` class.

See also:

- 21.2.4 `beginSheetForDirectory(path as folderitem, name as string, targetWindow as DesktopWindow)`
1070

21.2.6 Cancel

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: You can cancel the dialog using this method.

21.2.7 Constructor

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The constructor which initializes the panel with default values.

21.2.8 File as folderitem

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The file currently shown in the panel.

Notes: May be nil.

21.2.9 FileTypeForHFSType(hfstype as string) as string

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Returns a string encoding a file type code.

Example:

```
dim n as new NSSavePanelMBS
```

```
MsgBox n.FileTypeForHFSType("TEXT") // shows 'TEXT'
```

Notes: When using Mac Type codes, you need to use this function to convert them in a string the file manager understands.

21.2.10 HideNSNavNodePopUpButton

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Hides the navigation popup menu.

Notes: This is a function using undocumented features from the Apple NSSavePanel class, so there is not guarantee that it will work in future versions.

21.2.11 Ok

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: You can click ok in the dialog using this method.

21.2.12 runModal as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Displays the panel and begins its event loop with the current working (or last selected) directory as the default starting point.

Example:

```
dim n as new NSSavePanelMBS

dim e as integer = n.runModal

if e = n.NSCancelButton then
// cancelled
Break
elseif e = n.NSOKButton then
// got file
dim file as FolderItem = n.file
Break
end if
```

Notes: Returns NSOKButton or NSCancelButton or -1 on any error.

21.2.13 runModalForDirectory(path as folderitem, name as string) as Integer

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Initializes the panel to the directory specified by path and, optionally, the file specified by filename, then displays it and begins its modal event loop; path and filename can be empty strings.

Notes: If path is nil, the previous directory the Save panel was in is used. See NSCancelButton and NSOkButton constants for return values.

21.2.14 setAllowedFileTypes(filetype as string)

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A convenience method for setAllowedFileTypes with only one file type. See also:

- 21.2.15 setAllowedFileTypes(filetypes() as string) 1073

21.2.15 setAllowedFileTypes(filetypes() as string)

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Specifies the allowed file types.

Notes: A file type is an extension to be appended to any selected files that don't already have that extension; "nib" and "rtf" are examples. The items in types should not include the period that begins the extension. File type strings encoding HFS file types are not valid values. Pass an empty array, to allow any file type, which is the default.

See also:

- 21.2.14 setAllowedFileTypes(filetype as string) 1073

21.2.16 setTagNames(tagNames() as string)

Plugin Version: 23.0, Platform: macOS, Targets: Desktop only.

Function: Sets the tag names that you want to include on a saved file.

Example:

```
dim n as new NSSavePanelMBS
n.showsTagField = true
n.setTagNames array("test")
```

Notes: When the value of showsTagField is true, use this property to provide an array of strings that represent the initial tag names to display in the panel. If you set the property to nil or an empty array, the panel displays no initial tag names.

21.2.17 tagNames as string()

Plugin Version: 23.0, Platform: macOS, Targets: Desktop only.

Function: The tag names that you want to include on a saved file.

Notes: When the value of `showsTagField` is true, use this property to provide an array of strings that represent the initial tag names to display in the panel. If you set the property to nil or an empty array, the panel displays no initial tag names.

21.2.18 `validateVisibleColumns`

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Validates and possibly reloads the browser columns visible in the receiver by invoking the delegate method `shouldShowFilename`.

Notes: You might use this method if you want the browser to only allow selection of files with certain extensions based on the selection made in an accessory-view pop-up list. When the user changes the selection, you would invoke this method to revalidate the visible columns.

21.2.19 Properties

21.2.20 `accessoryView` as `NSViewMBS`

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The custom accessory view for the current application.

Notes: You can place any Cocoa control on that panel.
(Read and Write property)

21.2.21 `allowsOtherFileTypes` as `boolean`

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver allows the user to save files with an extension that's not in the list of allowed types.

Notes: If the user tries to save a filename with a recognized extension that's not in the list of allowed types they are presented with a dialog. If this property is true, then the dialog presents the option of using the extension the user specified.

The default setting is false.
(Read and Write property)

21.2.22 canCreateDirectories as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver allows the user to create directories.

Notes: (Read and Write property)

21.2.23 canSelectHiddenExtension as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the receiver allows the user to hide or show extensions.

Notes: (Read and Write property)

21.2.24 Directory as folderitem

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The path of the directory currently shown in the panel.

Notes: May be nil on any error.

(Read and Write property)

21.2.25 directoryURL as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The directory shown in the panel as a URL.

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

(Read and Write property)

21.2.26 isExpanded as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the panel is expanded.

Notes: (Read only property)

21.2.27 isExtensionHidden as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: Whether the extension-hiding checkbox is visible and checked.

Notes: True to show the checkbox and false to hide.

(Read and Write property)

21.2.28 Message as string

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The message displayed in the panel.

Notes: The default message text is an empty string.

(Read and Write property)

21.2.29 NameFieldLabel as string

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The string displayed in front of the filename text field.

Notes: By default the label is "Save As:".

(Read and Write property)

21.2.30 nameFieldStringValue as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The user-editable filename currently shown in the name field.

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

(Read and Write property)

21.2.31 Prompt as string

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The prompt of the default button.

Notes: This prompt appears on all SavePanel objects (or all OpenPanel objects if the panel of this message is an NSOpenPanel instance) in your application. By default the text in the default button is "Open" for an Open panel and "Save" for a Save panel.

It is intended that short words or phrases, such as "Open", "Save", "Set", or "Choose", be used on the button. The button is not resized to accommodate long prompts.

Since this method previously affected a title field, any colon at the end of prompt is removed.
(Read and Write property)

21.2.32 requiredFileType as string

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The required file type (if any).

Notes: A file specified in the Save panel is saved with the designated filename and this file type as an extension. Examples of common file types are "rtf", "tiff", and "ps". File type strings encoding HFS file types are not valid values for this attribute. An "" return value indicates that the user can save to any ASCII file.

This method is equivalent to calling allowedFileTypes and returning the first element of the list of allowed types, or "" if there are none.

(Read and Write property)

21.2.33 showsHiddenFiles as boolean

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Whether to show or hide the invisible files.

Notes: This is a function using undocumented features from the Apple NSSavePanel class, so there is not guarantee that it will work in future versions.

(Read and Write property)

21.2.34 showsTagField as Boolean

Plugin Version: 23.0, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the panel displays the Tags field.

Example:

```
dim n as new NSSavePanelMBS
n.showsTagField = true
n.setTagNames array("test")
```

Notes: When the value of this property is true, the panel displays the Tags field; if NO, the panel doesn't display the Tags field. The default value is true. (Note that the Tags field is appropriate only in a Save panel.)

If you set this property to true, you are responsible for setting tag names on the resulting file after saving is complete. If you don't set this property, macOS will automatically show the tag field and attempt to

apply the tags to the file. To set tags on files, use the `NSURLTagNamesKey`.
(Read and Write property)

21.2.35 Title as string

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: The title of the panel.

Notes: (Read and Write property)

21.2.36 `treatsFilePackagesAsDirectories` as boolean

Plugin Version: 7.8, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that indicates whether the panel displays file packages to the user as directories.

Notes: (Read and Write property)

21.2.37 Events

21.2.38 `compareFilename(name1 as string, name2 as string, caseSensitive as boolean)` as Integer

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: Controls the ordering of files presented by the `NSSavePanel`.

Notes: If this event has no code, the default behavior is used.

The `caseSensitive` argument, if true, indicates that the ordering is to be case-sensitive.

Don't reorder filenames in the Save panel without good reason, because it may confuse the user to have files in one Save panel or Open panel ordered differently than those in other such panels or in the Finder. The default behavior of Save and Open panels is to order files as they appear in the Finder. Note also that by implementing this method you will reduce the operating performance of the panel.

Constants:

`const NSOrderedAscending=-1 // The left operand is smaller than the right operand.`

`const NSOrderedSame=0 // The two operands are equal.`

`const NSOrderedDescending=1 // The left operand is greater than the right operand.`

21.2.39 directoryDidChange(path as string, folder as folderitem)

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: Sent when the user has changed the selected directory in the panel.

Notes: If this event has no code, the default behavior is used.

21.2.40 isValidFilename(path as string, item as folderitem) as boolean

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: Gives the delegate the opportunity to validate selected items.

Notes: If this event has no code, the default behavior is used.

The NSSavePanel object sender sends this event just before the end of a modal session for each filename displayed or selected (including filenames in multiple selections). The event determines whether it wants the file identified by filename; it returns true if the filename is valid, or false if the save panel should stay in its modal loop and wait for the user to type in or select a different filename or names. If the event refuses a filename in a multiple selection, none of the filenames in the selection is accepted.

21.2.41 panelSelectionDidChange

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: The event called whenever the selection changed in the dialog.

Notes: If this event has no code, the default behavior is used.

21.2.42 savePanelDidEnd(ReturnCode as Integer)

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: The save panel finished in sheet mode.

Notes: Returncode is NSOKButton or NSCancelButton.

21.2.43 shouldShowFilename(path as string, item as folderitem) as boolean

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: Gives the delegate the opportunity to filter items that it doesn't want the user to choose.

Notes: If this event has no code, the default behavior is used.

The `NSSavePanel` sends this event for each file or directory (filename) it is about to load in the browser. The delegate returns true if filename should be selectable, and false if the save panel should disable the file or directory.

21.2.44 `userEnteredFilename(filename as string, confirmed as boolean) as string`

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: Sent when the user confirms a filename choice by hitting OK or Return in the `NSSavePanel`.

Notes: You can either leave the filename alone, return a new filename, or return "" to cancel the save (and leave the Save panel as is). This method is sent before any required extension is appended to the filename and before the Save panel asks the user whether to replace an existing file.

Note that in the future, this method may be called multiple times in the sessions as the user types. In those cases, `okFlag` will be false until the user confirms the choice, in which case `okFlag` will become true. If the delegate does extensive validation or puts up alerts, it should do so only when `okFlag` is true.

If this event has no code, the default behavior is used.

21.2.45 `willExpand(expanding as boolean)`

Plugin Version: 7.8, Platform: macOS, Targets: .

Function: Sent when the `NSSavePanel` is about to expand or collapse because the user clicked the disclosure triangle that displays or hides the file browser.

Notes: If this event has no code, the default behavior is used.

21.2.46 Constants

Constants

Constant	Value	Description
<code>NSCancelButton</code>	0	One of the result codes you may need with this class.
<code>NSOKButton</code>	1	One of the result codes you may need with this class.

Chapter 22

Process

22.1 class Application

22.1.1 class Application

Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Extends the Application class inside Xojo.

22.1.2 Methods

22.1.3 NSApplicationMBS as NSApplicationMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns the shared NSApplication object.

Notes: This method gives easy access to the NSApplicationMBS class.

The plugin makes sure that there is only one application object by returning the same object each time. Works for Console, Desktop and Web projects. Version 19.4 declares this method for ConsoleApplication and Application class.

22.2 class ConsoleApplication

22.2.1 class ConsoleApplication

Plugin Version: 19.4, Platforms: macOS, Linux, Windows, Targets: Console only.

Function: Extends the ConsoleApplication class inside Xojo.

22.2.2 Methods

22.2.3 NSApplicationMBS as NSApplicationMBS

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

Function: Returns the shared NSApplication object.

Notes: This method gives easy access to the NSApplicationMBS class.

The plugin makes sure that there is only one application object by returning the same object each time. Works for Console, Desktop and Web projects. Version 19.4 declares this method for ConsoleApplication and Application class.

22.3 class DesktopApplication

22.3.1 class DesktopApplication

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

Function: Extends the Application class inside Xojo.

22.3.2 Methods

22.3.3 NSApplicationMBS as NSApplicationMBS

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

Function: Returns the shared NSApplication object.

Notes: This method gives easy access to the NSApplicationMBS class.

The plugin makes sure that there is only one application object by returning the same object each time. Works for Console, Desktop and Web projects.

22.4 class NSRunningApplicationMBS

22.4.1 class NSRunningApplicationMBS

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: NSRunningApplication is a class to manipulate and provide information for a single instance of an application.

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox n.localizedName
```

Notes: Only user applications are tracked; this does not provide information about every process on the system.

Some properties of an application are fixed, such as the bundle identifier. Other properties may vary over time, such as whether the app is hidden. Properties that vary can be observed with key-value observing, in which case the description comment for the method notes this capability.

Properties that vary over time are inherently race-prone. For example, a hidden app may unhide itself at any time. To ameliorate this, properties persist until the next turn of the main run loop in a common mode. For example, if you repeatedly poll an unhidden app for its hidden property without allowing the run loop to run, it will continue to return false, even if the app hides, until the next turn of the run loop.

NSRunningApplication is thread safe, in that its properties are returned atomically. However, it is still subject to the main run loop policy described above. If you access an instance of NSRunningApplication from a background thread, be aware that its time-varying properties may change from under you as the main run loop runs (or not).

An NSRunningApplication instance remains valid after the application exits. However, most properties lose their significance, and some properties may not be available on a terminated application.

Requires Mac OS X 10.6.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 14.3pr5](#)
- [MBS Real Studio Plugins, version 11.3pr9](#)

22.4.2 Methods

22.4.3 activateWithOptions(options as Integer) as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Attempts to activate the application using the specified options.

Notes: options: The options to use when activating the application. See "NSApplicationActivationOptions"

for the possible values.

Returns true if the application was activated successfully, otherwise false.

This method will return false if the application has quit, or is not a type of application than can be activated.

Available in Mac OS X v10.6 and later.

22.4.4 Constructor

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox n.localizedName
```

Notes: Initializes the object with the current application.

Available in Mac OS X v10.6 and later.

22.4.5 currentApplication as NSRunningApplicationMBS

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSRunningApplication representing this application.

Example:

```
dim n as NSRunningApplicationMBS = NSRunningApplicationMBS.currentApplication
```

```
MsgBox n.localizedName
```

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

22.4.6 forceTerminate as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Attempts to force the receiver to quit.

Notes: Returns true if the application successfully terminated, otherwise false.

This method will return false if the application is no longer running when the forceTerminate message is sent to the receiver.

This method may return before the receiver exits; you should observe the terminated property to determine when the application terminates.

Available in Mac OS X v10.6 and later.

22.4.7 hide as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Attempts to hide or the application.

Example:

```
dim n as NSRunningApplicationMBS = NSRunningApplicationMBS.currentApplication
```

```
MsgBox str(n.hide) // hide me
```

Notes: The property of this value will be false if the application has already quit, or if of a type that is unable to be hidden.

Available in Mac OS X v10.6 and later.

22.4.8 runningApplications as NSRunningApplicationMBS()

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of NSRunningApplication representing the running applications.

Example:

```
dim n(-1) as NSRunningApplicationMBS = NSRunningApplicationMBS.runningApplications
```

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

```
for each r as NSRunningApplicationMBS in n
```

```
s.Append r.localizedName
```

```
next
```

```
MsgBox Join(s,EndOfLine)
```

Notes: Returns an array of NSRunningApplication instances.

The order of the array is unspecified, but it is stable, meaning that the relative order of particular applications will not change across multiple calls to `runningApplications`. See NSRunningApplication Class Reference for more information on NSRunningApplication.

Similar to the NSRunningApplication classes's properties, this property will only change when the main run loop is run in a common mode. Instead of polling, use key-value observing to be notified of changes to this array property.

This property is thread safe, in that it may be called from background threads and the result is returned atomically.

This list is not updated in a tight loop. For receiving updates, you need to have the runloop run (e.g. check regularly in a timer).

22.4.9 `runningApplicationsWithBundleIdentifier(bundleID as string) as NSRunningApplicationMBS()`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of currently running applications with the specified bundle identifier.

Example:

```
dim n(-1) as NSRunningApplicationMBS = NSRunningApplicationMBS.runningApplicationsWithBundleIdentifier("com.apple.iTunes")
```

```
if UBound(n)>=0 then
  MsgBox n(0).localizedname
else
  MsgBox "iTunes is not running?"
end if
```

Notes: An array of NSRunningApplications, or an empty array if no applications match the bundle identifier.

Available in Mac OS X v10.6 and later.

22.4.10 `runningApplicationWithProcessIdentifier(pid as Integer) as NSRunningApplicationMBS`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the running application with the given process identifier, or nil if no application has that pid.

Example:

```
dim n as NSRunningApplicationMBS
dim pid as Integer

while n=nil
  pid=pid+1
  n=NSRunningApplicationMBS.runningApplicationWithProcessIdentifier(pid)
wend

MsgBox n.localizedName+" has PID "+str(pid)
```

Notes: pid: The process identifier.

Returns an instance of NSRunningApplication for the specified pid, or nil if the application has no process identifier.

Applications that do not have PIDs cannot be returned from this method.
Available in Mac OS X v10.6 and later.

22.4.11 terminate as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Attempts to quit the receiver normally.

Notes: Returns true if the application successfully terminated, otherwise false.

This method will return false if the application is no longer running when the terminate message is sent to the receiver.

This method may return before the receiver exits; you should observe the terminated property to determine when the application terminates.

Available in Mac OS X v10.6 and later.

22.4.12 unhide as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Attempts to unhide or the application.

Notes: Returns true if the application was successfully shown, otherwise false.

The property of this value will be false if the application has already quit, or if of a type that is unable to be hidden.

Available in Mac OS X v10.6 and later.

22.4.13 Properties

22.4.14 activationPolicy as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the activation policy of the application. (read-only)

Notes: The value returned by this property is usually fixed, but it may change through a call to activateWithOptions.

Available in Mac OS X v10.6 and later.

(Read only property)

22.4.15 active as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the application is currently frontmost. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox str(n.active)
```

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

(Read only property)

22.4.16 bundleIdentifier as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the CFBundleIdentifier of the application. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

MsgBox n.bundleIdentifier

Notes: The value of this property will be nil if the application does not have an Info.plist.
(Read only property)

22.4.17 bundleURL as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the URL to the application's bundle. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox n.bundleURL
```

Notes: Available in Mac OS X v10.6 and later.
(Read only property)

22.4.18 executableArchitecture as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the URL to the application's executable. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox str(n.executableArchitecture) // shows 7
```

Notes: Available in Mac OS X v10.6 and later.
(Read only property)

22.4.19 executableURL as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the URL to the application's executable. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox n.executableURL
```

Notes: Available in Mac OS X v10.6 and later.
(Read only property)

22.4.20 finishedLaunching as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the receiver's process has finished launching, (read-only)

Notes: The value of this property corresponds to the running application having received an `NSApplicationDidFinishLaunchingNotification` notification internally. Some applications do not post this notification (applications that do not rely on `NSApplication`) and so are never reported as finished launching.

Available in Mac OS X v10.6 and later.

(Read only property)

22.4.21 Handle as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal reference to the `NSRunningApplication` object.

Notes: (Read and Write property)

22.4.22 hidden as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the application is currently hidden. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox str(n.hidden)
```

Notes: Available in Mac OS X v10.6 and later.
(Read only property)

22.4.23 icon as NSImageMBS

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the icon for the receiver's application. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
dim i as NSImageMBS = n.icon
```

```
// without this call the size is 32x32 Pixel
i.setSize 512,512
```

```
Backdrop = i.CopyPictureWithMask
```

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

(Read only property)

22.4.24 launchDate as date

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the date when the application was launched. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox n.launchDate.ShortTime
```

Notes: This property is not available for all applications. Specifically, it is not available for applications that were launched not launched by LaunchServices.

Available in Mac OS X v10.6 and later.

(Read only property)

22.4.25 launchDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the date when the application was launched. (read-only)

Notes: This property is not available for all applications. Specifically, it is not available for applications that were launched not launched by LaunchServices.

Available in Mac OS X v10.6 and later.
(Read only property)

22.4.26 localizedName as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the localized name of the application. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

```
MsgBox n.localizedName
```

Notes: The value of this property is dependent on the current localization of the application and is suitable for presentation to the user.

Available in Mac OS X v10.6 and later.
(Read only property)

22.4.27 ownsMenuBar as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns whether the application owns the current menu bar.

Example:

```
// put in a timer...
```

```
dim r as NSRunningApplicationMBS = NSRunningApplicationMBS.currentApplication  
window1.Title = str(r.ownsMenuBar)
```

Notes: Available in Mac OS X v10.7 and later.
(Read only property)

22.4.28 processIdentifier as Integer

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the process identifier (pid) of the application. (read-only)

Example:

```
dim n as new NSRunningApplicationMBS
```

MsgBox str(n.processIdentifier)

Notes: Not all applications have a pid. Applications without a return a value of -1.

Do not rely on this for comparing processes, instead compare NSRunningApplication instances using isEqual:

Available in Mac OS X v10.6 and later.

(Read only property)

22.4.29 terminated as boolean

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates that the receiver's application has terminated. (read-only)

Notes: The value of terminated is true if the receiver's application has terminated, otherwise false.

This property is observable using key-value observing.

Available in Mac OS X v10.6 and later.

(Read only property)

22.4.30 Constants

Constants

Constant	Value	Description
NSApplicationActivateAllWindows	1	<p>One of the flag constants for activateWithOptions. By default, activation brings only the main and key windows forward. To specify NSApplicationActivateAllWindows, all of the application's windows are brought forward.</p> <p>Available in Mac OS X v10.6 and later.</p>
NSApplicationActivateIgnoringOtherApps	2	<p>One of the flag constants for activateWithOptions. By default, activation deactivates the calling app (assuming it was active) and then the new app is activated only if there is no currently active app. This prevents the new app from stealing focus from the user, if the user wants to activate and the user has switched to a different app in the interim. If you specify NSApplicationActivateIgnoringOtherApps, the application is activated regardless of the currently active app, potentially stealing focus from the user. You should rarely pass this flag because stealing key focus from the user is a very bad user experience.</p> <p>Available in Mac OS X v10.6 and later.</p>
NSApplicationActivationPolicyAccessory	1	<p>One of the constants used for following activation policies. The application does not appear in the Dock and does not have a menu bar, but it may be activated programmatically or by clicking on one of its windows. This corresponds to value of the LSUIElement key in the application's Info.plist being 1.</p> <p>Available in Mac OS X v10.6 and later.</p>
NSApplicationActivationPolicyProhibited	2	<p>One of the constants used for following activation policies. The application does not appear in the Dock and may not create windows or be activated. This corresponds to the value of the LSBackgroundOnly key in the application's Info.plist being 1. This is also the default for unexecutables that do not have Info.plists.</p> <p>Available in Mac OS X v10.6 and later.</p>
NSApplicationActivationPolicyRegular	0	<p>One of the constants used for following activation policies. The application is an ordinary app that appears in the Dock and has a standard user interface. This is the default for bundled apps, unless overridden in the application's Info.plist.</p> <p>Available in Mac OS X v10.6 and later.</p>

Chapter 23

Speech

23.1 class NSSpeechRecognizerMBS

23.1.1 class NSSpeechRecognizerMBS

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The class to handle the cocoa speech recognition.

Notes: Available in Mac OS X v10.3 and later.

See newer SFSpeechRecognizerMBS class.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 13.2pr7](#)

23.1.2 Methods

23.1.3 commands as string()

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the current list of commands.

23.1.4 Destructor

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

Function: The destructor.

23.1.5 SetCommands(commands()) as string

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the list of commands for which the receiver should listen to commands.

Notes: If the receiver is already listening, the current command list is updated and listening continues. commands should be an array of strings. The commands must be in U.S. English.

23.1.6 StartListening

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Tells the speech recognition engine to begin listening for commands.

Notes: When a command is recognized the message didRecognizeCommand is called.

23.1.7 StopListening

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Tells the speech recognition engine to suspend listening for commands.

23.1.8 Properties

23.1.9 BlocksOtherRecognizers as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the receiver should block all other recognizers (that is, other applications attempting to understand spoken commands) when listening.

Notes: (Read and Write property)

23.1.10 DisplayedCommandsTitle as string

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the title of the commands section or "" if there is no title.

Notes: Commands are displayed in the Speech Commands window indented under a section with this title. (Read and Write property)

23.1.11 ListensInForegroundOnly as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the receiver should only enable its commands when the receiver's application is the frontmost one.

Notes: (Read and Write property)

23.1.12 Events

23.1.13 DidRecognizeCommand(command as string)

Plugin Version: 6.4, Platform: macOS, Targets: .

Function: Invoked when the recognition engine has recognized the application command command.

Notes: command is one of the strings from the array passed to setCommands. The delegate typically evaluates which command was recognized and performs the related action.

23.2 class NSSpeechSynthesizerMBS

23.2.1 class NSSpeechSynthesizerMBS

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The class to handle the cocoa speech synthesizing.

Notes: Available in Mac OS X v10.3 and later.

More details on Apple's website:

https://developer.apple.com/library/mac/documentation/Cocoa/Reference/ApplicationKit/Classes/NSSpeechSynthesizer_Class/

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 15.0pr9](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr7](#)
- [MBS Real Studio Plugins, version 13.1pr11](#)

23.2.2 Methods

23.2.3 addSpeechDictionary(speechDictionary as dictionary)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Registers the given speech dictionary with the receiver.

Notes: speechDictionary: Speech dictionary to add to the receiver's dictionaries.

See the discussion of UseSpeechDictionary in Speech Synthesis Manager Reference for more information. Available in OS X v10.5 and later.

23.2.4 attributesForVoice(voice as String) as NSVoiceMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns information about a voice or nil.

23.2.5 availableVoice(index as Integer) as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns name of a voice.

Notes: The available voices can be listed using this function.
Index is from 0 to count-1.

23.2.6 availableVoices as String()

Plugin Version: 10.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the array with the identifiers for the available voices.

Example:

```
MsgBox Join(NSSpeechSynthesizerMBS.availableVoices)
```

Notes: Same as availableVoice() and availableVoicesCount, but this function returns an array which is very useful for for-each-loops.

23.2.7 availableVoicesCount as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Number of voices available.

23.2.8 Constructor

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the class with the default voice.

Notes: This constructor is needed for the events to fire.
The given voice must be valid!

See also:

- 23.2.9 Constructor(voice as string)

23.2.9 Constructor(voice as string)

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the class with a voice.

Notes: This constructor is needed for the events to fire.
See also:

- 23.2.8 Constructor

23.2.10 continueSpeaking

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Continues speaking after it has been paused.

Notes: Mac OS X 10.5 only.

23.2.11 defaultVoice as String

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the identifier of the default voice.

Example:

```
dim s as NSSpeechSynthesizerMBS
```

```
s=new NSSpeechSynthesizerMBS
```

```
MsgBox s.defaultVoice
```

23.2.12 Destructor

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

Function: The destructor.

23.2.13 isAnyApplicationSpeaking as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether any other application is currently speaking through the sound output device.

23.2.14 NSSpeechCharacterModeProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForProperty to get or set the characteristics of a synthesizer.

Example:

```
dim s as new NSSpeechSynthesizerMBS
dim e as NSErrorMBS
```

```
call s.setObjectForProperty(s.NSSpeechModeLiteral, s.NSSpeechCharacterModeProperty, e)
msgBox s.objectForProperty(s.NSSpeechCharacterModeProperty, e)
```

```
call s.setObjectForProperty(s.NSSpeechModeNormal, s.NSSpeechCharacterModeProperty, e)
msgBox s.objectForProperty(s.NSSpeechCharacterModeProperty, e)
```

Notes: Get or set the synthesizer’s current text-processing mode. A string that specifies whether the channel is currently in text input mode or phoneme input mode.

When the character-processing mode is `NSSpeechModeNormal`, input characters are spoken as you would expect to hear them. When the mode is `NSSpeechModeLiteral`, each character is spoken literally, so that the word “cat” is spoken “C–A–T”.

Available in OS X v10.5 and later.

23.2.15 NSSpeechCommandDelimiterProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with `setObjectForProperty` and `objectForProperty` to get or set the characteristics of a synthesizer.

Notes: Set the embedded speech command delimiter characters to be used for the synthesizer. A dictionary that contains the delimiter information. See “Command Delimiter Keys” for the keys you can use to specify values in this dictionary.

By default, the opening delimiter is “[” and the closing delimiter is “] ”. Your application might need to change these delimiters temporarily if those character sequences occur naturally in a text buffer that is to be spoken. Your application can also disable embedded command processing by passing empty delimiters (as empty strings). See “Speech Command Delimiter” for the keys you can use to specify values in this dictionary.

Available in OS X v10.5 and later.

23.2.16 NSSpeechCommandPrefix as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants speech-command delimiters keys used in `NSSpeechCommandDelimiterProperty`.

Notes: The command delimiter string that prefixes a command, by default, this is “[”.

Available in OS X v10.5 and later.

23.2.17 NSSpeechCommandSuffix as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants speech-command delimiters keys used in NSSpeechCommandDelimiterProperty.

Notes: The command delimiter string that suffixes a command, by default, this is]]. Available in OS X v10.5 and later.

23.2.18 NSSpeechCurrentVoiceProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForProperty to get or set the characteristics of a synthesizer.

Notes: Set the current voice on the synthesizer to the specified voice. A dictionary that contains the phoneme symbols and example words defined for the current synthesizer.

Your application might use this information to show the user what symbols to use when entering phonemic text directly. See "NSSpeechPhonemeSymbolsProperty Dictionary Keys" for the keys you can use to specify values in this dictionary.

Available in OS X v10.5 and later.

23.2.19 NSSpeechDictionaryAbbreviations as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify key-value pairs used to add vocabulary to the dictionary using addSpeechDictionary.

Notes: An array of dictionary objects containing the keys NSSpeechDictionaryEntrySpelling and NSSpeechDictionaryEntryPhonemes.

Available in OS X v10.5 and later.

23.2.20 NSSpeechDictionaryEntryPhonemes as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify key-value pairs used to add vocabulary to the dictionary using addSpeechDictionary.

Notes: The phonemic representation of an entry. A string.

Available in OS X v10.5 and later.

23.2.21 NSSpeechDictionaryEntrySpelling as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify key-value pairs used to add vocabulary to the dictionary using `addSpeechDictionary`.

Notes: The spelling of an entry. A string.
Available in OS X v10.5 and later.

23.2.22 NSSpeechDictionaryLocaleIdentifier as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify key-value pairs used to add vocabulary to the dictionary using `addSpeechDictionary`.

Notes: The canonical locale identifier string describing the dictionary's locale. A locale is generally composed of three pieces of ordered information: a language code, a region code, and a variant code. Refer to documentation about `NSLocale` or `Locales Programming Guide` for more information
Available in OS X v10.5 and later.

23.2.23 NSSpeechDictionaryModificationDate as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify key-value pairs used to add vocabulary to the dictionary using `addSpeechDictionary`.

Notes: A string representation of the dictionary's last modification date in the international format (YYYY-MM-DD HH:MM:SS \pm HHMM). If the same word appears across multiple dictionaries, the one from the dictionary with the most recent date will be used.
Available in OS X v10.5 and later.

23.2.24 NSSpeechDictionaryPronunciations as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify key-value pairs used to add vocabulary to the dictionary using `addSpeechDictionary`.

Notes: An array of dictionary objects containing the keys `NSSpeechDictionaryEntrySpelling` and `NSSpeechDictionaryEntryPhonemes`.

23.2.25 NSSpeechErrorCount as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the key constants identify errors that may occur during speech synthesis. They are used with NSSpeechErrorsProperty.

Notes: The number of errors that have occurred in processing the current text string, since the last call to objectForProperty with the NSSpeechErrorsProperty property. A Number

Using the NSSpeechErrorOldestCode keys and the NSSpeechErrorNewestCode keys, you can get information about the oldest and most recent errors that occurred since the last call to objectForProperty, but you cannot get information about any intervening errors.

Available in OS X v10.5 and later.

23.2.26 NSSpeechErrorNewestCharacterOffset as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the key constants identify errors that may occur during speech synthesis. They are used with NSSpeechErrorsProperty.

Notes: The position in the text string of the most recent error that occurred since the last call to objectForProperty with the NSSpeechErrorsProperty property. A Number.

Available in OS X v10.5 and later.

23.2.27 NSSpeechErrorNewestCode as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the key constants identify errors that may occur during speech synthesis. They are used with NSSpeechErrorsProperty.

Notes: The error code of the most recent error that occurred since the last call to objectForProperty with the NSSpeechErrorsProperty property. A number

Available in OS X v10.5 and later.

23.2.28 NSSpeechErrorOldestCharacterOffset as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the key constants identify errors that may occur during speech synthesis. They are used with NSSpeechErrorsProperty.

Notes: The position in the text string of the first error that occurred since the last call to objectForProperty with the NSSpeechErrorsProperty property. A number

Available in OS X v10.5 and later.

23.2.29 NSSpeechErrorOldestCode as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the key constants identify errors that may occur during speech synthesis. They are used with NSSpeechErrorsProperty.

Notes: The error code of the first error that occurred since the last call to objectForKeyProperty with the NSSpeechErrorsProperty property. A Number
Available in OS X v10.5 and later.

23.2.30 NSSpeechErrorsProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForKeyProperty to get or set the characteristics of a synthesizer.

Notes: Get speech-error information for the synthesizer. An Dictionary object that contains speech-error information. See "NSSpeechErrorProperty Dictionary Keys" for a description of the keys present in the dictionary.

This property lets you get information about various run-time errors that occur during speaking, such as the detection of badly formed embedded commands. Errors returned directly by the Speech Synthesis Manager are not reported here.

If your application implements the didEncounterErrorAtIndex event, the event can use this property to get error information.

Available in OS X v10.5 and later.

23.2.31 NSSpeechInputModeProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForKeyProperty to get or set the characteristics of a synthesizer.

Notes: Get or set the synthesizer's current text-processing mode. A string that specifies whether the channel is currently in text input mode or phoneme input mode. The supported values are listed in "Speaking Modes for NSSpeechInputModeProperty."

When in phoneme-processing mode, a text string is interpreted to be a series of characters representing various phonemes and prosodic controls. Some synthesizers might support additional input-processing modes and define constants for these modes.

Available in OS X v10.5 and later.

23.2.32 NSSpeechModeLiteral as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants defining the available text-processing and number-processing modes for a synthesizer. This key is used with `NSSpeechInputModeProperty` and `NSSpeechNumberModeProperty`

Notes: Indicates that each digit or character is spoken literally (so that 12 is spoken as "one, two", or the word "cat" is spoken as "C A T").

Available in OS X v10.5 and later.

23.2.33 `NSSpeechModeNormal` as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants defining the available text-processing and number-processing modes for a synthesizer. This key is used with `NSSpeechInputModeProperty` and `NSSpeechNumberModeProperty`

Notes: Indicates that the synthesizer assembles digits into numbers (so that 12 is spoken as "twelve") and text into words.

Available in OS X v10.5 and later.

23.2.34 `NSSpeechModePhoneme` as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify input modes are used with `NSSpeechInputModeProperty`.

Notes: Indicates that the synthesizer is in phoneme-processing mode. When in phoneme-processing mode, a text buffer is interpreted to be a series of characters representing various phonemes and prosodic controls.

Available in OS X v10.5 and later.

23.2.35 `NSSpeechModeText` as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify input modes are used with `NSSpeechInputModeProperty`.

Notes: Indicates that the synthesizer is in text-processing mode.

Available in OS X v10.5 and later.

23.2.36 `NSSpeechNumberModeProperty` as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with `setObjectForProperty` and `objectForProperty` to get or set the characteristics of a synthesizer.

Example:

`dim s as new NSSpeechSynthesizerMBS`

`dim e as NSErrorMBS`

`call s.setObjectForProperty(s.NSSpeechModeLiteral, s.NSSpeechNumberModeProperty, e)`
`msgBox s.objectForProperty(s.NSSpeechNumberModeProperty, e)`

`call s.setObjectForProperty(s.NSSpeechModeNormal, s.NSSpeechNumberModeProperty, e)`
`msgBox s.objectForProperty(s.NSSpeechNumberModeProperty, e)`

Notes: Get or set the synthesizer's current number-processing mode. A string that specifies whether the synthesizer is currently in normal or literal number-processing mode. The constants `NSSpeechModeNormal` and `NSSpeechModeLiteral` are the possible values of this string.

When the number-processing mode is `NSSpeechModeNormal`, the synthesizer assembles digits into numbers (so that "12" is spoken as "twelve"). When the mode is `NSSpeechModeLiteral`, each digit is spoken literally (so that "12" is spoken as "one, two").

Available in OS X v10.5 and later.

23.2.37 NSSpeechOutputToFileURLProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with `setObjectForProperty` and `objectForProperty` to get or set the characteristics of a synthesizer.

Notes: Set the speech output destination to a file or to the computer's speakers. A `NSURL` object. To write the speech output to a file, use the file's `NSURL`; to generate the sound through the computer's speakers, use `nil`.

Available in OS X v10.5 and later.

23.2.38 NSSpeechPhonemeInfoExample as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys used in the `NSSpeechPhonemeSymbolsProperty` dictionary.

Notes: An example word that illustrates the use of the phoneme.

Available in OS X v10.5 and later.

23.2.39 NSSpeechPhonemeInfoHiliteEnd as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys used in the `NSSpeechPhonemeSymbolsProperty` dictionary.

Notes: The character offset into the example word that identifies the location of the end of the phoneme. Available in OS X v10.5 and later.

23.2.40 NSSpeechPhonemeInfoHiliteStart as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys used in the NSSpeechPhonemeSymbolsProperty dictionary.

Notes: The character offset into the example word that identifies the location of the beginning of the phoneme.

Available in OS X v10.5 and later.

23.2.41 NSSpeechPhonemeInfoOpcode as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys used in the NSSpeechPhonemeSymbolsProperty dictionary.

Notes: The opcode as Number.

Available in OS X v10.5 and later.

23.2.42 NSSpeechPhonemeInfoSymbol as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the keys used in the NSSpeechPhonemeSymbolsProperty dictionary.

Notes: The symbol used to represent the phoneme.

The symbol does not necessarily have a phonetic connection to the phoneme, but might simply be an abstract textual representation of it.

Available in OS X v10.5 and later.

23.2.43 NSSpeechPhonemeSymbolsProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForProperty to get or set the characteristics of a synthesizer.

Notes: Get a list of phoneme symbols and example words defined for the synthesizer. A Dictionary object that contains the phoneme symbols and example words defined for the current synthesizer

Your application might use this information to show the user what symbols to use when entering phonemic text directly. See "NSSpeechPhonemeSymbolsProperty Dictionary Keys" for a description of the keys present in the dictionary.

Available in OS X v10.5 and later.

23.2.44 NSSpeechPitchBaseProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with `setObjectForProperty` and `objectForProperty` to get or set the characteristics of a synthesizer.

Example:

```
dim s as new NSSpeechSynthesizerMBS
dim e as NSErrorMBS
```

```
dim n as Integer = s.objectForProperty(s.NSSpeechPitchBaseProperty, e)
msgBox str(n)
```

Notes: Get or set a synthesizer's baseline speech pitch. An number that specifies the baseline speech pitch. Typical voice frequencies range from around 90 hertz for a low-pitched male voice to perhaps 300 hertz for a high-pitched child's voice. These frequencies correspond to approximate pitch values in the ranges of 30.000 to 40.000 and 55.000 to 65.000, respectively.

Note: The change in speech pitch may not be noticeable until the next sentence or paragraph is spoken.

Available in OS X v10.5 and later.

23.2.45 NSSpeechPitchModProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with `setObjectForProperty` and `objectForProperty` to get or set the characteristics of a synthesizer.

Example:

```
dim s as new NSSpeechSynthesizerMBS
dim e as NSErrorMBS
```

```
dim n as Double = s.objectForProperty(s.NSSpeechPitchModProperty, e)
msgBox str(n)
```

Notes: Get or set a synthesizer's pitch modulation. A number object that specifies the synthesizer's pitch modulation.

Pitch modulation is also expressed as a floating-point value in the range of 0.000 to 127.000. These values correspond to MIDI note values, where 60.000 is equal to middle C on a piano scale. The most useful speech pitches fall in the range of 40.000 to 55.000. A pitch modulation value of 0.000 corresponds to a monotone in

which all speech is generated at the frequency corresponding to the speech pitch. Given a speech pitch value of 46.000, a pitch modulation of 2.000 would mean that the widest possible range of pitches corresponding to the actual frequency of generated text would be 44.000 to 48.000.

Note: The change in pitch modulation may not be noticeable until the next sentence or paragraph is spoken. Available in OS X v10.5 and later.

23.2.46 NSSpeechRateProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForProperty to get or set the characteristics of a synthesizer.

Example:

```
dim s as new NSSpeechSynthesizerMBS
dim e as NSErrorMBS
dim value as Double = s.objectForProperty(s.NSSpeechRateProperty, e)
MsgBox str(value)
```

Notes: Get or set the synthesizer's baseline speech pitch. A number that specifies the synthesizer's baseline speech pitch.

Typical voice frequencies range from around 90 hertz for a low-pitched male voice to perhaps 300 hertz for a high-pitched child's voice. These frequencies correspond to approximate pitch values in the ranges of 30.000 to 40.000 and 55.000 to 65.000, respectively.

Available in OS X v10.5 and later.

23.2.47 NSSpeechRecentSyncProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForProperty to get or set the characteristics of a synthesizer.

Notes: Get the message code for the most recently encountered synchronization command. A number that specifies the most recently encountered synchronization command.

Available in OS X v10.5 and later.

23.2.48 NSSpeechResetProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty to get or set the characteristics of a synthesizer.

Notes: Set a synthesizer back to its default state. There is no value associated with this property; to reset the channel to its default state, set the key to nil.

You can use this function to, for example, set speech pitch and speech rate to default values.

Available in OS X v10.5 and later.

23.2.49 NSSpeechStatusNumberOfCharactersLeft as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify speech status keys used with NSSpeechStatusProperty.

Notes: The number of characters left in the input string of text.

When the value of this key is zero, you can destroy the input string.

Available in OS X v10.5 and later.

23.2.50 NSSpeechStatusOutputBusy as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify speech status keys used with NSSpeechStatusProperty.

Notes: Indicates whether the synthesizer is currently producing speech.

A synthesizer is considered to be producing speech even at some times when no audio data is being produced through the computer's speaker. This occurs, for example, when the synthesizer is processing input, but has not yet initiated speech or when speech output is paused.

Available in OS X v10.5 and later.

23.2.51 NSSpeechStatusOutputPaused as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify speech status keys used with NSSpeechStatusProperty.

Notes: Indicates whether speech output in the synthesizer has been paused by sending the message `pauseSpeakingAtBoundary`.

Available in OS X v10.5 and later.

23.2.52 NSSpeechStatusPhonemeCode as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants identify speech status keys used with NSSpeechStatusProperty.

Notes: Indicates that the synthesizer is in phoneme-processing mode. When in phoneme-processing mode, a text buffer is interpreted to be a series of characters representing various phonemes and prosodic controls. Available in OS X v10.5 and later.

23.2.53 NSSpeechStatusProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with setObjectForProperty and objectForProperty to get or set the characteristics of a synthesizer.

Example:

```
dim s as new NSSpeechSynthesizerMBS
dim e as NSErrorMBS
```

```
call s.startSpeakingString "Hello"
```

```
dim status as Dictionary = s.objectForProperty(s.NSSpeechStatusProperty, e)
dim CharactersLeft as Integer = status.Value(s.NSSpeechStatusNumberOfCharactersLeft)
```

```
MsgBox str(CharactersLeft)+" Characters left"
```

Notes: Get speech-status information for the synthesizer. A dictionary that contains speech-status information for the synthesizer. See "NSSpeechStatusProperty Dictionary Keys" for a description of the keys present in the dictionary.

Available in OS X v10.5 and later.

23.2.54 NSSpeechSynthesizerInfoIdentifier as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants are keys used in the NSSpeechSynthesizerInfoProperty dictionary.

Notes: The identifier of the speech synthesizer.

Available in OS X v10.5 and later.

23.2.55 NSSpeechSynthesizerInfoProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with `setObjectForProperty` and `objectForProperty` to get or set the characteristics of a synthesizer.

Notes: Get information about the speech synthesizer being used on the specified synthesizer. A dictionary object that contains information about the speech synthesizer being used on the specified synthesizer. See "Speech Synthesizer Property Keys" for a description of the keys present in the dictionary.

Available in OS X v10.5 and later.

23.2.56 NSSpeechSynthesizerInfoVersion as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants are keys used in the `NSSpeechSynthesizerInfoProperty` dictionary.

Notes: The version of the speech synthesizer.

Available in OS X v10.5 and later.

23.2.57 NSSpeechVolumeProperty as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants used with `setObjectForProperty` and `objectForProperty` to get or set the characteristics of a synthesizer.

Example:

```
dim s as new NSSpeechSynthesizerMBS
dim e as NSErrorMBS
dim value as Double = s.objectForProperty(s.NSSpeechVolumeProperty, e)
MsgBox str(value)
```

Notes: Get or set the speech volume for a synthesizer. A Number that specifies the synthesizer's speech volume.

Volumes are expressed in floating-point values ranging from 0.0 through 1.0. A value of 0.0 corresponds to silence, and a value of 1.0 corresponds to the maximum possible volume. Volume units lie on a scale that is linear with amplitude or voltage. A doubling of perceived loudness corresponds to a doubling of the volume. Available in OS X v10.5 and later.

You may prefer to simply use the volume property.

23.2.58 objectForProperty(PropertyName as string, byref error as NSErrorMBS) as Variant

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the value of a receiver's property.

Notes: PropertyName: Property to get.

error: On output, error that occurred while obtaining the value of speechProperty.

Returns the value of speechProperty.

Available in OS X v10.5 and later.

23.2.59 pauseSpeakingAtBoundary(boundary as Integer)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Pauses the speaking on the given boundary.

Example:

```
dim s as NSSpeechSynthesizerMBS // your synthesizer
s.pauseSpeakingAtBoundary s.NSSpeechSentenceBoundary // pause on end of sentence.
```

Notes: Mac OS X 10.5 only.

boundary can be NSSpeechImmediateBoundary, NSSpeechSentenceBoundary or NSSpeechWordBoundary.

23.2.60 phonemesFromText(text as string) as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the phonemes from a given text.

Example:

```
dim s as new NSSpeechSynthesizerMBS
```

```
MsgBox s.phonemesFromText("Hello")
```

```
// shows "_hEHl1OW."
```

Notes: Mac OS X 10.5 only.

23.2.61 setObjectForProperty(value as Variant, PropertyName as string, byref error as NSErrorMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Specifies the value of a receiver's property.

Notes: PropertyName: Property to set.

Error: On output, error that occurred while setting speechProperty.

Returns true when the speechProperty was set. False when there was an error, specified in error.
Available in OS X v10.5 and later.

23.2.62 SetVoice(voice as string) as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the receiver's current voice.

Notes: Returns true on success.

23.2.63 StartSpeakingString(text as string) as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Begins speaking synthesized text through the system's default sound output device.

Example:

```
dim s as new NSSpeechSynthesizerMBS
```

```
s.rate=300 // not slow
```

```
s.volume=0.5 // not loud
```

```
call s.startSpeakingString "Hello World"
```

Notes: Returns true when synthesis starts successfully, false otherwise.

If the receiver is currently speaking synthesized speech when startSpeakingString is called, that process is stopped before text is spoken.

When synthesis of text finishes normally or is stopped, the message didFinishSpeaking(true) is called.

See also:

- 23.2.64 StartSpeakingString(text as string, file as folderitem) as boolean

- 23.2.65 `startSpeakingString(Text as String, URL as string)` as boolean 1118

23.2.64 `StartSpeakingString(text as string, file as folderitem)` as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Begins synthesizing text into a sound (AIFF) file.

Notes: Returns true when synthesis starts successfully, false otherwise.

When synthesis of text finishes normally or is stopped, the message `didFinishSpeaking(True)` is called.

One example of how you might use this method is in an email program that automatically converts new messages into sound files that can be stored on an iPod for later listening.

See also:

- 23.2.63 `StartSpeakingString(text as string)` as boolean 1117
- 23.2.65 `startSpeakingString(Text as String, URL as string)` as boolean 1118

23.2.65 `startSpeakingString(Text as String, URL as string)` as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Begins synthesizing text into a sound (AIFF) file.

Notes: Returns true when synthesis starts successfully, false otherwise.

When synthesis of text finishes normally or is stopped, the message `didFinishSpeaking(True)` is called.

One example of how you might use this method is in an email program that automatically converts new messages into sound files that can be stored on an iPod for later listening.

See also:

- 23.2.63 `StartSpeakingString(text as string)` as boolean 1117
- 23.2.64 `StartSpeakingString(text as string, file as folderitem)` as boolean 1118

23.2.66 `StopSpeaking`

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Stops synthesis in progress.

Notes: If the receiver is currently generating speech, synthesis is halted, and the message `didFinishSpeaking(false)` is called.

23.2.67 stopSpeakingAtBoundary(boundary as Integer)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Stops speech on the next w

Notes: Mac OS X 10.5 only.

23.2.68 Properties

23.2.69 IsSpeaking as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the receiver is currently generating synthesized speech.

Notes: true when the receiver is generating synthesized speech, false otherwise.

(Read only property)

23.2.70 rate as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The current rate of the speech.

Example:

```
dim s as NSSpeechSynthesizerMBS
s=new NSSpeechSynthesizerMBS
MsgBox str(s.rate)
// shows e.g. "160"
```

Notes: Mac OS X 10.5 only.

The range of supported rates is not predefined by the Speech Synthesis framework; but the synthesizer may only respond to a limited range of speech rates. Average human speech occurs at a rate of 180 to 220 words per minute.

(Read and Write property)

23.2.71 UsesFeedbackWindow as boolean

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the receiver uses the speech feedback window.

Notes: (Read and Write property)

23.2.72 Voice as string

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The current voice.

Example:

```
dim s as new NSSpeechSynthesizerMBS
MsgBox s.voice
// shows e.g. "com.apple.speech.synthesis.voice.Zarvox"
```

Notes: (Read and Write property)

23.2.73 volume as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The volume of the speech.

Example:

```
dim s as new NSSpeechSynthesizerMBS
MsgBox str(s.volume)
// shows e.g. 1
```

Notes: No sound is zero, full sound one.
Mac OS X 10.5 only.

Volumes are expressed in floating-point units ranging from 0.0 through 1.0. A value of 0.0 corresponds to silence, and a value of 1.0 corresponds to the maximum possible volume. Volume units lie on a scale that is linear with amplitude or voltage. A doubling of perceived loudness corresponds to a doubling of the volume.

Setting a value outside this range is undefined.
(Read and Write property)

23.2.74 Events**23.2.75 didEncounterErrorAtIndex(characterIndex as Integer, text as string, message as string)**

Plugin Version: 7.7, Platform: macOS, Targets: .

Function: An event called when an error has been found in the text while speaking.

Notes: Mac OS X 10.5 only.

23.2.76 didEncounterSyncMessage(message as string)

Plugin Version: 7.7, Platform: macOS, Targets: .

Function: A sync message was found in the text.

Notes: See Apple Speech documentation about the special tags you need to place in the text to get this event.

Mac OS X 10.5 only.

23.2.77 didFinishSpeaking(finishedSpeaking as boolean)

Plugin Version: 6.4, Platform: macOS, Targets: .

Function: Called when speaking through the sound output device is done.

Notes: finishedSpeaking is true when finished normally and false when StopSpeaking was called.

23.2.78 willSpeakPhoneme(phonemeOpcode as Integer)

Plugin Version: 6.4, Platform: macOS, Targets: .

Function: Sent just before a synthesized phoneme is spoken through the sound output device.

Notes: phonemeOpcode: Phoneme that sender is about to speak into the sound output device.

One use of this method might be to animate a mouth on screen to match the generated speech.

Important: The delegate is not sent this message when the SpeechSynthesizer object is synthesizing speech to a file (startSpeakingString).

23.2.79 willSpeakWord(Position as Integer, Length as Integer, Text as String)

Plugin Version: 6.4, Platform: macOS, Targets: .

Function: Sent just before a synthesized word is spoken through the sound output device.

Notes: Position and Length: Word that sender is about to speak into the sound output device.

text: Text that is being synthesized by sender.

One use of this method might be to visually highlight the word being spoken.

Important: The delegate is not sent this message when the `SpeechSynthesizer` object is synthesizing speech to a file (`startSpeakingString`).

23.2.80 Constants

Constants

Constant	Value	Description
<code>NSSpeechImmediateBoundary</code>	0	One of the constants for the <code>pauseSpeakingAtBoundary</code> method. Mac OS X 10.5 only.
<code>NSSpeechSentenceBoundary</code>	2	One of the constants for the <code>pauseSpeakingAtBoundary</code> method. Mac OS X 10.5 only.
<code>NSSpeechWordBoundary</code>	1	One of the constants for the <code>pauseSpeakingAtBoundary</code> method. Mac OS X 10.5 only.

23.3 class NSVoiceMBS

23.3.1 class NSVoiceMBS

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A voice on Mac OS X with its attributes.

Notes: Available in Mac OS X v10.3 and later.

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 13.2pr7](#)

23.3.2 Methods

23.3.3 Age as Integer

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The perceived age (in years) of the voice.

23.3.4 Constructor

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

Function: The private constructor.

23.3.5 Demotext as String

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A demonstration string to speak.

23.3.6 Gender as String

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The perceived gender of the voice.

Notes: May be either GenderNeuter ("VoiceGenderNeuter"), GenderFemale ("VoiceGenderFemale"), or GenderMale ("VoiceGenderMale").

23.3.7 GenderFemale as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the possible values for the gender property.

23.3.8 GenderMale as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the possible values for the gender property.

23.3.9 GenderNeuter as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the possible values for the gender property.

23.3.10 Identifier as String

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A unique string identifying the voice. The identifiers of the system voices are listed below.

Notes: Identifiers of the Mac OS X system voices

com.apple.speech.synthesis.voice.Agnes
com.apple.speech.synthesis.voice.Albert
com.apple.speech.synthesis.voice.BadNews
com.apple.speech.synthesis.voice.Bahh
com.apple.speech.synthesis.voice.Bells
com.apple.speech.synthesis.voice.Boing
com.apple.speech.synthesis.voice.Bruce
com.apple.speech.synthesis.voice.Bubbles
com.apple.speech.synthesis.voice.Cellos
com.apple.speech.synthesis.voice.Deranged
com.apple.speech.synthesis.voice.Fred
com.apple.speech.synthesis.voice.GoodNews
com.apple.speech.synthesis.voice.Hysterical
com.apple.speech.synthesis.voice.Junior
com.apple.speech.synthesis.voice.Kathy
com.apple.speech.synthesis.voice.Organ
com.apple.speech.synthesis.voice.Princess
com.apple.speech.synthesis.voice.Ralph
com.apple.speech.synthesis.voice.Trinoids

com.apple.speech.synthesis.voice.Vicki
 com.apple.speech.synthesis.voice.Victoria
 com.apple.speech.synthesis.voice.Whisper
 com.apple.speech.synthesis.voice.Zarvox

23.3.11 Language as String

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The language of the voice (currently US English only).

Notes: Language has been replaced by LocaleIdentifier in Mac OS X 10.5.

23.3.12 LocaleIdentifier as String

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The locale identifier.

Example:

```
dim v as NSVoiceMBS
dim s as NSSpeechSynthesizerMBS
dim n as string

s=new NSSpeechSynthesizerMBS

n=s.voice
v=s.attributesForVoice(n)

MsgBox n+EndOfLine+v.LocaleIdentifier
// shows for example "en_US"
```

Notes: Mac OS X 10.5 only.

Language has been replaced by LocaleIdentifier in Mac OS X 10.5.

23.3.13 Name as String

Plugin Version: 6.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The name of the voice suitable for display.

23.3.14 NSVoiceAge as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: The perceived age (in years) of the voice. A string.

23.3.15 NSVoiceDemoText as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: A demonstration string to speak. A String.

23.3.16 NSVoiceGender as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: The perceived gender of the voice.

23.3.17 NSVoiceIdentifier as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: A unique string identifying the voice.

Identifiers of the OS X system voices:

com.apple.speech.synthesis.voice.Agnes
com.apple.speech.synthesis.voice.Albert
com.apple.speech.synthesis.voice.Alex
com.apple.speech.synthesis.voice.BadNews
com.apple.speech.synthesis.voice.Bahh
com.apple.speech.synthesis.voice.Bells
com.apple.speech.synthesis.voice.Boing
com.apple.speech.synthesis.voice.Bruce
com.apple.speech.synthesis.voice.Bubbles
com.apple.speech.synthesis.voice.Cellos
com.apple.speech.synthesis.voice.Deranged
com.apple.speech.synthesis.voice.Fred
com.apple.speech.synthesis.voice.GoodNews
com.apple.speech.synthesis.voice.Hysterical

com.apple.speech.synthesis.voice.Junior
com.apple.speech.synthesis.voice.Kathy
com.apple.speech.synthesis.voice.Organ
com.apple.speech.synthesis.voice.Princess
com.apple.speech.synthesis.voice.Ralph
com.apple.speech.synthesis.voice.Trinoids
com.apple.speech.synthesis.voice.Vicki
com.apple.speech.synthesis.voice.Victoria
com.apple.speech.synthesis.voice.Whisper
com.apple.speech.synthesis.voice.Zarvox

23.3.18 NSVoiceIndividuallySpokenCharacters as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: A list of unicode character id ranges that define the unicode characters that can be spoken in character-by-character mode by this voice. Each list entry is a dictionary containing two keys: "UnicodeCharBegin", an integer value containing the beginning unicode id of this range; and "UnicodeCharEnd", an integer value containing the ending unicode id of this range.

These ranges can be used by your application to determine if the voice can speak the name of an individual character when spoken in character-by-character mode.

Some voices may not provide this attribute.

Available in OS X v10.5 and later.

23.3.19 NSVoiceLanguage as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: The language of the voice (currently US English only). A string

Deprecated: Use NSVoiceLocaleIdentifier instead.

Deprecated in OS X v10.5.

23.3.20 NSVoiceLocaleIdentifier as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: The language of the voice. A string

The canonical locale identifier string describing the voice's locale. A locale is generally composed of three pieces of ordered information: a language code, a region code, and a variant code. Refer to documentation about the NSLocale class or Locales Programming Guide for more information.

Available in OS X v10.5 and later.

23.3.21 NSVoiceName as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: The name of the voice suitable for display. A String.

23.3.22 NSVoiceSupportedCharacters as String

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the properties names.

Notes: A list of unicode character id ranges that define the unicode characters supported by this voice. a dictionary containing two keys: "UnicodeCharBegin", an integer value containing the beginning unicode id of this range; and "UnicodeCharEnd", an integer value containing the ending unicode id of this range. The synthesizer will convert or ignore any characters not contained in the range of supported characters.

Some voices may not provide this attribute.

Available in OS X v10.5 and later.

23.3.23 Properties as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns dictionary with all the properties for this voice.

Notes: This dictionary may contain additional information if Apple adds more features in newer Mac OS X versions.

Chapter 24

Spell Checking

24.1 class NSSpellCheckerMBS

24.1.1 class NSSpellCheckerMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The spell checker class.

Notes: The NSSpellChecker object is used by a client (e.g. a document in an application) to spell-check a given String.

There is only one NSSpellChecker instance per application (since spell-checking is interactive and you only have one mouse and one keyboard).

The string being spell-checked need only be valid for the duration of the call to checkSpellingOfString or countWordsInString.

Requires Mac OS X 10.2.

Blog Entries

- [Spell Checking on Windows for Xojo](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 19.1](#)
- [MBS Xojo Plugins, version 19.1pr6](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 17.4](#)
- [MBS Xojo Plugins, version 17.4pr5](#)
- [MBS Real Studio Plugins, version 13.1pr1](#)
- [MBS Real Studio Plugins, version 11.3pr14](#)

- [MBS REALbasic plug-in 9.6](#)

Xojo Developer Magazine

- [5.4, page 50: The Case of the Top 20, The mayor is caught in an explosive situation, and our boys are on the case by Toby Rush](#)
- [17.3, page 11: News](#)
- [15.6, page 9: News](#)

24.1.2 Methods

24.1.3 availableLanguages as string()

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a list of the available languages.

Notes: Requires Mac OS X 10.5.

24.1.4 checkGrammarOfString(text as string, start as Integer, language as string, wrap as boolean) as NSRangeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initiates a grammatical analysis of a given string.

Notes: text: String to analyze.

start: Location within string at which to start the analysis.

language: Language use in string. When nil, the language selected in the Spelling panel is used.

wrap: true to specify that the analysis continue to the beginning of string when the end is reached. false to have the analysis stop at the end of string.

outDetails: Optional. On output, dictionaries describing grammar-analysis details within the flagged grammatical unit. See the NSSpellServer class for information about these dictionaries.

Returns Location of the first flagged grammatical unit.

Available in Mac OS X v10.5 and later.

See also:

- [24.1.5 checkGrammarOfString\(text as string, start as Integer, language as string, wrap as boolean, Details\(\) as dictionary\) as NSRangeMBS](#) 1131

24.1.5 checkGrammarOfString(text as string, start as Integer, language as string, wrap as boolean, Details() as dictionary) as NSRangeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initiates a grammatical analysis of a given string.

Notes: text: String to analyze.

start: Location within string at which to start the analysis.

language: Language use in string. When nil, the language selected in the Spelling panel is used.

wrap: true to specify that the analysis continue to the beginning of string when the end is reached. false to have the analysis stop at the end of string.

outDetails: Optional. On output, dictionaries describing grammar-analysis details within the flagged grammatical unit. See the NSSpellServer class for information about these dictionaries.

Returns Location of the first flagged grammatical unit.

Available in Mac OS X v10.5 and later.

See also:

- 24.1.4 checkGrammarOfString(text as string, start as Integer, language as string, wrap as boolean) as NSRangeMBS 1130

24.1.6 checkSpellingOfString(text as string, start as Integer) as NSRangeMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initiates a spell-check of a string.

Notes: Returns the range of the first misspelled word.

See also:

- 24.1.7 checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean) as NSRangeMBS 1131
- 24.1.8 checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean, byref WordCount as Integer) as NSRangeMBS 1132

24.1.7 checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean) as NSRangeMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initiates a spell-check of a string.

Notes: Returns the range of the first misspelled word.

See also:

- 24.1.6 checkSpellingOfString(text as string, start as Integer) as NSRangeMBS 1131

- 24.1.8 `checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean, byref WordCount as Integer) as NSRangeMBS` 1132

24.1.8 `checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean, byref WordCount as Integer) as NSRangeMBS`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initiates a spell-check of a string.

Notes: Returns the range of the first misspelled word (and optionally the wordCount by reference).

See also:

- 24.1.6 `checkSpellingOfString(text as string, start as Integer) as NSRangeMBS` 1131
- 24.1.7 `checkSpellingOfString(text as string, start as Integer, language as string, wrap as boolean) as NSRangeMBS` 1131

24.1.9 `checkString(text as string, range as NSRangeMBS = nil, checkingTypes as Int64 = -1, options as Dictionary = nil, byref orthography as NSOrthographyMBS, byref wordCount as Integer) as NSTextCheckingResultMBS()`

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

Function: Requests unified text checking for the given range of the given string.

Notes: text: The string to check.

range: The range of the string to check.

checkingTypes: The type of checking to be performed. The possible constants are listed in NSTextCheckingType and can be combined using the C bit-wise OR operator to perform multiple checks at the same time.

options: The options dictionary specifying the types of checking to perform. See Spell Checking Option Dictionary Keys for the possible keys and expected values.

orthography: Returns by-reference, the orthography of the range of the string. See NSOrthographyMBS for more information.

wordCount: Returns by-reference, the word count for the range of the string.

Returns an array of NSTextCheckingResultMBS objects describing particular items found during checking and their individual ranges, sorted by range origin, then range end, then result type.

24.1.10 `completionsForPartialWordRange(start as Integer, length as Integer, text as string, language as string="") as string()`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Searches possible completions for the given word.

Notes: Returns an array of strings, in the order in which they should be presented, representing complete words that the user might be trying to type when starting by typing the partial word at the given range in the given string.

Within the text, length characters are picked starting at at position (0 based) and matched against the dictionary defined by language.

Up to around 100 words are returned.

Requires Mac OS X 10.3.

Returns an empty string on any error.

24.1.11 correctionForWordRange(range as NSRangeMBS, text as string, language as string) as string

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

Function: Returns a single proposed correction if a word is mis-spelled.

Notes: range: The range of the word to be corrected.

text: The string containing the proposed correction.

language: The language.

Returns the proposed correct string.

24.1.12 countWordsInString(word as string, language as string="") as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Just counts the words without checking spelling.

Example:

```
dim text as string = "Hello World"
dim spell as NSSpellCheckerMBS // your spellchecker

msgBox str(spell.countWordsInString(text,"en"))
```

Notes: Returns the number of words in text. The language argument specifies the language used in the string. If language is the empty string, the current selection in the Spelling panel's pop-up menu is used.

Returns -1 if text is nil or this spellchecker function is not available.

Returns -1 if counting words isn't supported by the spell server selected.

24.1.13 `deletesAutospaceBetweenString(precedingString as string, followingString as string, language as String = "") as Boolean`

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

Function: Deletes auto space between strings.

Notes: In some cases the space automatically inserted after an accepted candidate should be deleted when the next text is typed (e.g. if it is a period or comma). This method allows clients to recognize these cases in a standardized way.

24.1.14 `dismissCorrectionIndicatorForView(view as NSViewMBS)`

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

Function: Dismisses the correction indicator for the specified view.

24.1.15 `forgetWord(word as string)`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Removes the given word from the user dictionary.

24.1.16 `guessesForWord(range as NSRangeMBS, word as string, language as string) as string()`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of possible substitutions for the specified string.

Notes: range: The range of the string to check.

word: The string to guess

language: The language of the string

Returns an array of strings containing possible replacement words.

Available in Mac OS X v10.6 and later.

See also:

- 24.1.17 `guessesForWord(word as string) as string()`

24.1.17 guessesForWord(word as string) as string()

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array with words matching the given word.

Example:

```
dim a() as string
dim s as new NSSpellCheckerMBS
a=s.guessesForWord("Hell")

msgbox str(ubound(a)+1)+" suggestions."
```

Notes: Returns nil on any error.

Returns an array of suggested spellings for the misspelled word word. If word contains all capital letters, or its first letter is capitalized, the suggested words are capitalized in the same way.

See also:

- 24.1.16 guessesForWord(range as NSRangeMBS, word as string, language as string) as string() 1134

24.1.18 hasLearnedWord(word as string) as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Checks whether a word has been learned.

Notes: Returns true if the word is known.

Requires Mac OS X 10.5.

24.1.19 ignoredWords as string()

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The list of ignored words.

Example:

```
dim spell as new NSSpellCheckerMBS
dim f as FolderItem
dim i,c as Integer
dim t as TextOutputStream
dim n(-1) as string

f=SpecialFolder.Preferences.Child("SpellCheck RB.pref")
t=f.CreateTextFile
```

```
if t<>nil and spell<>nil then
n=spell.ignoredWords
MsgBox Join(n,EndOfLine)
end if
```

Notes: Returns nil on any error.

24.1.20 ignoreWord(word as string)

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds the word to the ignore list so it will be ignored for spell checking in this NSSpellCheckerMBS object.

Notes: Requires Mac OS X 10.2.

24.1.21 isAutomaticCapitalizationEnabled as boolean

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

Function: Whether automatic capitalization is enabled by the user.

24.1.22 isAutomaticDashSubstitutionEnabled as boolean

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

Function: Whether automatic dash substitution is enabled by the user.

24.1.23 isAutomaticPeriodSubstitutionEnabled as boolean

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

Function: Whether automatic period substitution is enabled by the user.

24.1.24 isAutomaticQuoteSubstitutionEnabled as boolean

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

Function: Whether automatic quote substitution is enabled by the user.

24.1.25 isAutomaticSpellingCorrectionEnabled as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether automatic spelling correction is enabled.

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

24.1.26 isAutomaticTextCompletionEnabled as boolean

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

Function: Whether automatic text completion is enabled by the user.

24.1.27 isAutomaticTextReplacementEnabled as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether automatic spelling replacement is enabled.

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

24.1.28 languageForWordRange(range as NSRangeMBS, text as string, orthography as NSOrthographyMBS = nil) as string

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

Function: Clients who have an NSOrthography from NSTextCheckingTypeOrthography checking and wish to determine a specific language from it for a particular word.

24.1.29 languageMenuEntries as string()

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The entries from the language menu of the panel.

Notes: Copies the list of menu entries from the panel.

Currently you can use this names to show to the user and see what languages are available. It uses a private property which works for Mac OS X 10.4 and 10.5.

For the language function you need the short names:

Australian English	en_AU
British English	en_GB
Canadian English	en_CA
Deutsch	de
English	en
Español	es
Français	fr
Italiano	it
Multilingual	Multilingual
Nederlands	nl
Português	pt
Português do Brasil	pt_BR
Svenska	sv

24.1.30 learnWord(word as string)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Learns the given word.

Notes: Requires Mac OS X 10.5.

24.1.31 menuForResult(TextCheckingResult as NSTextCheckingResultMBS, checkedString as string, options as Dictionary = nil, atLocation as NSPointMBS, view as NSViewMBS) as NSMenuMBS

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

Function: Provides a menu containing contextual menu items suitable for certain kinds of detected results.

Notes: TextCheckingResult: The NSTextCheckingResult instance for the checked string.

checkedString: The string that has been checked.

options: The options dictionary allows clients to pass in information associated with the document. See Spell Checking Option Dictionary Keys for possible key-value pairs.

location: The location, in the view's coordinate system, to display the menu.

view: The view object over which to display the contextual menu.

Returns a menu suitable for displaying as a contextual menu, or adding to another contextual menu as a submenu.

24.1.32 NSSpellCheckerDidChangeAutomaticCapitalizationNotification as string

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

Function: The notification named used when automatic capitalization setting changed.

24.1.33 NSSpellCheckerDidChangeAutomaticDashSubstitutionNotification as string

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

Function: The notification named used when automatic dash substitution setting changed.

24.1.34 NSSpellCheckerDidChangeAutomaticPeriodSubstitutionNotification as string

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

Function: The notification named used when automatic period substitution setting changed.

24.1.35 NSSpellCheckerDidChangeAutomaticQuoteSubstitutionNotification as string

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

Function: The notification named used when automatic quote substitution setting changed.

24.1.36 NSSpellCheckerDidChangeAutomaticSpellingCorrectionNotification as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Use with NSNotificationObserverMBS class.

This notification is posted when the spell checker did change text using automatic spell checking correction.

The are posted to the application's default notification center.

Available in Mac OS X v10.7 and later.

24.1.37 NSSpellCheckerDidChangeAutomaticTextCompletionNotification as string

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

Function: The notification named used when automatic text completion setting changed.

24.1.38 NSSpellCheckerDidChangeAutomaticTextReplacementNotification as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the notification names.

Notes: Use with NSNotificationObserverMBS class.

This notification is posted when the spell checker changed text using automatic text replacement. The are posted application's to the default notification center.

Available in Mac OS X v10.7 and later.

24.1.39 NSTextCheckingDocumentAuthorKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: An string containing the name of an author to be associated with the document

Available in Mac OS X v10.6 and later.

24.1.40 NSTextCheckingDocumentTitleKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: A string containing the title to be associated with the document.

Available in Mac OS X v10.6 and later.

24.1.41 NSTextCheckingDocumentURLKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: An NSURL to be associated with the document.

Available in Mac OS X v10.6 and later.

24.1.42 NSTextCheckingOrthographyKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: An NSOrthography instance indicating an orthography to be used as a starting point for orthography checking, or as the orthography if orthography checking is not enabled.

Available in Mac OS X v10.6 and later.

24.1.43 NSTextCheckingQuotesKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: An NSArray containing four strings to be used with NSTextCheckingTypeQuote (opening double quote, closing double quote, opening single quote, and closing single quote in that order); if not specified, values will be taken from user's preferences.

Available in Mac OS X v10.6 and later.

24.1.44 NSTextCheckingReferenceDateKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: An NSDate to be associated with the document, used as a referent for relative dates; if not specified, the current date will be used.

Available in Mac OS X v10.6 and later.

24.1.45 NSTextCheckingReferenceTimeZoneKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: An NSTimeZone to be associated with the document, used as a reference for dates without time zones; if not specified, the current time zone will be used.

Available in Mac OS X v10.6 and later.

24.1.46 NSTextCheckingRegularExpressionsKey as string

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

Function: One of the constants for the options dictionary.

Notes: Available in Mac OS X 10.7 or newer.

Currently not directly supported with our plugins.

24.1.47 NSTextCheckingReplacementsKey as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for the options dictionary.

Notes: A dictionary containing replacements to be used with NSTextCheckingTypeReplacement; if not specified, values will be taken from user's preferences.

Available in Mac OS X v10.6 and later.

24.1.48 NSTextCheckingSelectedRangeKey as string

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

Function: One of the constants for the options dictionary.

Notes: NSValueMBS containing NSRangeMBS, should be the portion of the selected range intersecting the string being checked, or a zero-length range if there is an insertion point in or adjacent to the string being checked, or New NSRangeMBS(NSNotFound, 0) if the selection is entirely outside of the string being checked.

24.1.49 preventsAutocorrectionBeforeString(text as string, language as String = "") as Boolean

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

Function: In some cases the next typing should prevent a pending correction (if it is an @, for example).

Notes: This method allows clients to recognize these cases in a standardized way.

24.1.50 recordResponse(response as Integer, correction as string, word as string, language as String = "")

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

Function: Records the user response to the correction indicator being displayed.

Notes: response: The user's response. The possible values are shown in NSCorrectionResponse.

correction: The corrected word. This should match the original correction.

word: The original word. This should match the original correction.

language: The language being edited. This should match the original correction.

When a correction is automatically proposed, the user may respond in one of several ways. Clients may report this to the spell checker so that it can learn from the user's response and adjust future correction behavior accordingly.

Use of this method implies that the client stored the original word and original correction at least from the point at which the user accepts it until the user edits or reverts it.

24.1.51 requestCandidatesForSelectedRange(selectedRange as NSRangeMBS, stringToCheck as string, types as Int64 = -1, options as Dictionary = nil, tag as Variant = nil) as Integer

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

Function: Requests candidates in background.

24.1.52 requestCheckingOfString(stringToCheck as string, range as NSRangeMBS, types as Int64 = -1, options as Dictionary = nil, tag as Variant = nil) as Integer

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

Function: Requests that the string be checked in the background.

Notes: stringToCheck: The string to check.

range: The range of the string to check.

types: The type of checking to be performed. The possible constants are listed in NSTextCheckingType and can be combined using the C bit-wise OR operator to perform multiple checks at the same time.

options: The options dictionary specifying the types of checking to perform. See Spell Checking Option Dictionary Keys for the possible keys and expected values.

tag: An identifier unique within the application used to inform the spell checker which document that text is

associated, potentially for many purposes, not necessarily just for ignored words. A value of 0 can be passed in for text not associated with a particular document.

The return value is a monotonically increasing sequence number that can be used to keep track of requests in flight.

Calls requestCheckingOfStringCompleted event later on main thread.

24.1.53 setIgnoredWords(words() as string)

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the ignored word list.

Example:

```
dim spell as NSSpellCheckerMBS // your spellchecker

dim f as FolderItem
dim t as TextInputStream
dim words(-1),line as string

f=SpecialFolder.Preferences.Child("SpellCheck RB.pref")
t=f.OpenAsTextFile
if t<>nil and spell<>nil then

while not t.eof
line=t.ReadLine(encodings.UTF8)
if line<>"" then
words.Append line
end if
wend

spell.setIgnoredWords words

end if
```

24.1.54 setLanguage(language as string) as boolean

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Allows programmatic setting of the language to spell-check in.

Notes: Normally chosen by a pop-up-list in the spelling panel and defaulted to the user's preferred language, so call this with care.

Set to "" to use the language from the panel popup menu.

24.1.55 sharedSpellCheckerExists as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns whether the application's NSSpellChecker has already been created.

Notes: Returns true if the shared spell checker already exists, otherwise false.

24.1.56 spellingPanel as NSPanelMBS

Plugin Version: 7.1, Platform: macOS, Targets: Desktop only.

Function: The spelling panel used for spell checking.

24.1.57 unlearnWord(word as string)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Tells the spell checker to unlearn a given word.

Notes: Available in Mac OS X v10.5 and later.

Same as the older forgetWord.

24.1.58 updatePanels

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Updates the available panels to account for user changes.

Notes: This method should be called when a client changes some relevant setting, such as what kind of spelling, grammar checking, or substitutions it uses.

Available in Mac OS X v10.6 and later.

24.1.59 updateSpellingPanelWithGrammarString(lang as string, detail as dictionary)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Specifies a grammar-analysis detail to highlight in the Spelling panel.

Notes: `problemString`: Problematic grammatical unit identified by `checkGrammarOfString`.
`detail`: One of the grammar-analysis details provided by `checkGrammarOfString`.

Available in Mac OS X v10.5 and later.

24.1.60 `updateSpellingPanelWithMisspelledWord(word as string)`

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Updates the panel with the word.

Notes: The `checkSpellingOfString` methods return the range of the misspelled word found. It is up to the client to select that word in their document and to cause the spelling panel to update itself to reflect the found misspelling. Clients can call `updateSpellPanelWithMisspelledWord` to insure that the spell panel is up to date.

24.1.61 `userPreferredLanguages as string()`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides a subset of the available languages to be used for spell checking.

Notes: Returns an array containing the user's preferred languages for spell checking. The order is set in the system preferences.

If `automaticallyIdentifiesLanguages` is true, then text checking will automatically use this method as appropriate; otherwise, it will use the language set by `Language` property.

The older `checkSpellingOfString` and `checkGrammarOfString` methods will use the language set by `Language` property, if they are called with an empty language argument.

Available in Mac OS X v10.6 and later.

24.1.62 `userQuotesArrayForLanguage(lang as string) as string()`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the default values for quote replacement.

Example:

```
dim n as new NSSpellCheckerMBS
```

```
dim en(-1) as string = n.userQuotesArrayForLanguage("en")
```

```
MsgBox Join(en, " ")
```

Notes: An array of quote replacements used by the NSTextCheckingQuotesKey key-value pair.

Available in Mac OS X v10.6 and later.

24.1.63 userReplacementsDictionary as dictionary

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the dictionary used when replacing words.

Notes: The key-value pairs in this dictionary are used by the NSTextCheckingQuotesKey when replacing characters and words.

Available in Mac OS X v10.6 and later.

24.1.64 Properties

24.1.65 accessoryView as NSViewMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The Spelling panel's accessory view.

Notes: The accessory view can be any custom view you want to display with the spelling panel. The accessory view is displayed below the spelling checker and the panel automatically resizes to accommodate the accessory view.

This method posts a notification named NSWindowDidResizeNotification with the Spelling panel object to the default notification center.

The accessory view or nil if there is none.
(Read and Write property)

24.1.66 automaticallyIdentifiesLanguages as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the spell checker will automatically identify languages.

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

(Read and Write property)

24.1.67 Handle as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the used NSSpellChecker reference.

Notes: (Read and Write property)

24.1.68 language as string

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The current language used.

Example:

```
dim c as NSSpellCheckerMBS
```

```
c=new NSSpellCheckerMBS
```

```
c.Language="en"
```

```
MsgBox c.Language // shows en
```

```
c.Language="Dutch"
```

```
MsgBox c.Language // shows nl
```

```
c.Language="Multilingual"
```

```
MsgBox c.Language // shows Multilingual
```

Notes: (Read and Write property)

24.1.69 Length as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The last length of the position where an error was found.

Notes: Length is 0 if no location was found.

(Read and Write property)

24.1.70 Location as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The last location where an error was found.

Notes: (Read and Write property)

24.1.71 substitutionsPanel as NSPanelMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The substitutions panel.

Notes: (Read only property)

24.1.72 substitutionsPanelAccessoryViewController as NSViewControllerMBS

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

Function: Sets the substitutions panel,Äôs accessory view.

Notes: accessoryController: The accessory view controller or nil if there is none.

The accessory view controller can accommodate any custom view you want to display with the substitutions panel. The accessory view controller,Äôs view is displayed below the substitutions list and the panel automatically resizes to accommodate the accessory view.

This method posts a notification named NSWindowDidResizeNotification with the substitutions panel object to the default notification center.

(Read only property)

24.1.73 Tag as Integer

Plugin Version: 7.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The document tag for the current document.

Notes: Every NSSpellCheckerMBS gets a new tag. The tag identifies which ignore list is used. This value is set automatically.

(Read and Write property)

24.1.74 WordFieldValue as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The word textfield content string.

Notes: (Read and Write computed property)

24.1.75 Events

24.1.76 Correct

Plugin Version: 7.1, Platform: macOS, Targets: .

Function: An event sent whenever the Correct button is pressed.

24.1.77 FindNext

Plugin Version: 7.1, Platform: macOS, Targets: .

Function: An event sent whenever the Find Next button is pressed.

24.1.78 Ignore

Plugin Version: 7.1, Platform: macOS, Targets: .

Function: An event sent whenever the Ignore button is pressed.

24.1.79 requestCandidatesForSelectedRangeCompleted(sequenceNumber as Integer, candidates() as NSTextCheckingResultMBS, stringToCheck as String, selectedRange as NSRangeMBS, checkingTypes as Int64, options as Dictionary, tag as Variant)

Plugin Version: 19.1, Platform: macOS, Targets: .

Function: Called later by requestCandidatesForSelectedRange method.

24.1.80 `requestCheckingOfStringCompleted(sequenceNumber as Integer, results() as NSTextCheckingResultMBS, orthography as NSOrthographyMBS, wordCount as Integer, stringToCheck as String, Range as NSRangeMBS, checkingTypes as Int64, options as Dictionary, tag as Variant)`

Plugin Version: 19.1, Platform: macOS, Targets: .

Function: Called later by `requestCheckingOfString` method.

Notes: `sequenceNumber`: A monotonically increasing sequence number.

`results`: An array of `NSTextCheckingResultMBS` objects describing particular items found during checking and their individual ranges, sorted by range origin, then range end, then result type..

`orthography`: The orthography of the string.

`wordCount`: The number of words in the range of the string.

24.1.81 Constants

Constants

Constant	Value	Description
<code>NSGrammarCorrections</code>	<code>"NSGrammarCorrections"</code>	One of the key constants for the dictionaries in the detail view.
<code>NSGrammarRange</code>	<code>"NSGrammarRange"</code>	One of the key constants for the dictionaries in the detail view.
<code>NSGrammarUserDescription</code>	<code>"NSGrammarUserDescription"</code>	One of the key constants for the dictionaries in the detail view.

Correction indicator type Constants

Constant	Value	Description
<code>NSCorrectionIndicatorTypeDefault</code>	0	The default indicator that shows a proposed correction. Available in Mac OS X v10.7 and later.
<code>NSCorrectionIndicatorTypeGuesses</code>	2	Shows multiple alternatives from which the user may choose the appropriate spelling. Available in Mac OS X v10.7 and later.
<code>NSCorrectionIndicatorTypeReversion</code>	1	Provides the option to revert to the original form after a correction has been made. Available in Mac OS X v10.7 and later.

Response Constants

Constant	Value	Description
<code>NSCorrectionResponseAccepted</code>	1	The user accepted the correction. Available in Mac OS X v10.7 and later.
<code>NSCorrectionResponseEdited</code>	4	After the correction was accepted, the user edited the corrected word (to something other than its original form). Available in Mac OS X v10.7 and later.
<code>NSCorrectionResponseIgnored</code>	3	The user continued in such a way as to ignore the correction. Available in Mac OS X v10.7 and later.
<code>NSCorrectionResponseNone</code>	0	No response was received from the user. Available in Mac OS X v10.7 and later.
<code>NSCorrectionResponseRejected</code>	2	The user rejected the correction by dismissing the correction indicator. Available in Mac OS X v10.7 and later.
<code>NSCorrectionResponseReverted</code>	5	After the correction was accepted, the user reverted the correction back to the original word. Available in Mac OS X v10.7 and later.

Chapter 25

Statusitem

25.1 class NSStatusItemMBS

25.1.1 class NSStatusItemMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: A class to handle a NSStatusitem which is a tiny little item in the menubar.

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

- [IP in menubar 4.6](#)
- [MBS Xojo Plugins, version 17.5pr9](#)
- [MBS Xojo / Real Studio Plugins, version 16.2pr1](#)
- [IP in menubar 4.5](#)
- [NSImage and image orientation](#)
- [Reducing app size with #if](#)
- [MBS Real Studio Plugins, version 11.2pr9](#)
- [One reason to upgrade REALbasic](#)
- [MBS REALbasic plug-in 9.6](#)
- [MonkeyBread Software Releases the MBS Plugins 8.2](#)

Xojo Developer Magazine

- 5.6, pages 31 to 33: Third Party Plugins: Statusitems, Getting an icon in the top right of the Mac OS X menubar by Christian Schmitz

25.1.2 Methods

25.1.3 Available as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Whether Statusitems are available.

Notes: True on MachO platforms.

25.1.4 Close

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The destructor.

Notes: There is no need to call this method except you want to free all resources of this object now without waiting for Xojo to do it for you.

25.1.5 CreateMenu as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new menu.

Notes: Handle is not 0 after this call was successful.

If the NSGrayBackground option is set in the system defaults, Mac OS X 10.5 will raise an NSImageCache-Exception, so please install an exception handle to catch NSExcptionMBS so your application can handle that.

See also:

- 25.1.6 CreateMenu(length as single) as boolean

1154

25.1.6 CreateMenu(length as single) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Creates the StatusItem menu with the given width.

Example:

```
dim e as new NSStatusItemMBS
call e.CreateMenu(24) // best for a 16x16 picture
```

Notes: Constants:

```
NSVariableStatusItemLength  -1
NSSquareStatusItemLength    -2
```

Other values between 0 and 10000 are used for the length.
Bad values like 20000 will crash the application.

Handle is not 0 after this call was successful.

If the NSGrayBackground option is set in the system defaults, Mac OS X 10.5 will raise an NSImageCache-Exception, so please install an exception handle to catch NSExcptionMBS so your application can handle that.

See also:

- 25.1.5 CreateMenu as boolean

1154

25.1.7 CreateMenuMiddle(length as single) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use CreateMenu instead.

Function: Same as CreateMenu but tries to place Statusitem on the right side.

Notes: Uses private Apple API which may break on future Mac OS X versions. Returns false if Apple changes something in the future and the function we use will not be available. So if this function returns false, you can call CreateMenu to continue. Works on Mac OS X 10.4 to 10.6 and make the new statusitem being the rightmost one (left to Apples MenuItems).

Constants:

```
NSVariableStatusItemLength  -1
NSSquareStatusItemLength    -2
```

Other values between 0 and 10000 are used for the length.
Bad values like 20000 will crash the application.

Handle is not 0 after this call was successful.

If the `NSGrayBackground` option is set in the system defaults, Mac OS X 10.5 will raise an `NSImageCacheException`, so please install an exception handle to catch `NSExcptionMBS` so your application can handle that.

Deprecated.

25.1.8 CreateMenuRight(length as single) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use `CreateMenu` instead.

Function: Same as `CreateMenu` but tries to place `Statusitem` on the right side.

Notes: Uses private Apple API which may break on future Mac OS X versions. Returns false if Apple changes something in the future and the function we use will not be available. So if this function returns false, you can call `CreateMenu` to continue. Works on Mac OS X 10.4 to 10.6 and make the new `statusitem` being the rightmost one.

If you run this code after you used `CreateMenuRight`, the new menu will be right of all other `statusitems`:

```
dim sh as new Shell
sh.Execute "killall", "SystemUIServer"
```

Constants:

```
NSVariableStatusItemLength  -1
NSSquareStatusItemLength    -2
```

Other values between 0 and 10000 are used for the length.

Bad values like 20000 will crash the application.

Handle is not 0 after this call was successful.

If the `NSGrayBackground` option is set in the system defaults, Mac OS X 10.5 will raise an `NSImageCacheException`, so please install an exception handle to catch `NSExcptionMBS` so your application can handle that.

Deprecated.

25.1.9 DrawStatusBarBackground(x as Double, y as Double, width as Double, height as Double, highlight as boolean)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Draws the menu background pattern for a custom status-bar item in regular or highlight pattern.

Notes: x, y, width and height: A rectangle defining the area of a custom status-bar item.

highlight: true to draw the background pattern in the standard highlight pattern, false to not highlight the pattern.

You can use this method to help a custom status-bar item emulate the behavior of a standard item.

Available in Mac OS X v10.3 and later.

25.1.10 MenuIsVertical as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns true if the statusitem has a vertical orientation.

Example:

```
MsgBox str(NSStatusItemMBS.MenuIsVertical)
```

25.1.11 MenuThickness as Double

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: The thickness of the status bar.

Example:

```
MsgBox str(NSStatusItemMBS.MenuThickness)
```

Notes: The status bar returned by `systemStatusBar` has a thickness of 22 pixels, the thickness of the menu bar.

25.1.12 popUpStatusItemMenu(menu as NSMenuMBS)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Displays a menu under a custom status bar item.

Notes: You can use this method to cause a popup menu to appear under a custom status bar item when the user clicks the item. Note that view must exist (that is, it must not be nil).

25.1.13 `SendActionOn(mode as Integer)`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Sets the conditions on which action event is called.

Notes: mask is set with one or more of the following bit masks described in NSEvent Constants: `NSLeftMouseDownMask`, `NSLeftMouseUpMask`, `NSLeftMouseDraggedMask`, and `NSPeriodicMask`.

mode is set with one or more of the following bit masks:

```
const NSLeftMouseDownMask    = &h00002
const NSLeftMouseUpMask      = &H00004
const NSLeftMouseDraggedMask = &h00040
const NSPeriodicMask         = &h10000
```

25.1.14 Properties

25.1.15 `alternateImage as NSImageMBS`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: An alternate image to be displayed when a status bar item is highlighted.

Notes: (Read and Write property)

25.1.16 `attributedString as NSAttributedStringMBS`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The attributed string that is displayed at the status item's position in the status bar.

Notes: If an image is also set, the title appears to the right of the image.

(Read and Write property)

25.1.17 `Button as Variant`

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: The button that is displayed in the status bar.

Notes: Value is a NSStatusBarButtonMBS object. Returned as Variant to reduce plugin dependencies.

This is created automatically on the creation of the StatusItem. Behavior customization for the button, such as image, target/action, tooltip, can be set with this property.

Available on Mac OS X 10.10 and newer.

(Read only property)

25.1.18 Enabled as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Whether this status item is enabled.

Notes: (Read and Write property)

25.1.19 Handle as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The handle of the NSStatusItem object used internally.

Notes: (Read and Write property)

25.1.20 Height as single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The height of the status item.

Notes: Should be 22 pixels.

This is a function using undocumented features from the Apple NSStatusItem class, so there is not guarantee that it will work in future versions.

(Read only property)

25.1.21 HighlightMode as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver is highlighted when clicked.

Notes: If you use HighlightMode and Menu, you need to first assign the menu and later set HighlightMode to true.

(Read and Write property)

25.1.22 image as NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The image that is displayed at the statusitem's position in the status bar.

Notes: nil if an image has not been set.

(Read and Write property)

25.1.23 Left as single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The left position of the status item.

Notes: Valid only after item was drawn the first time.

This is a function using undocumented features from the Apple `NSStatusItem` class, so there is not guarantee that it will work in future versions.

(Read only property)

25.1.24 Length as single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The length of the status item in pixels.

Notes: Constants for special values:

```
NSVariableStatusItemLength  -1  
NSSquareStatusItemLength   -2
```

(Read and Write property)

25.1.25 Menu as NSMenuMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The menu attached to this statusitem.

Notes: Nil if no menu is attached.

(Read and Write property)

25.1.26 Title as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The title of the status item.

Notes: (Read and Write property)

25.1.27 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)

25.1.28 Top as single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The top position of the status item.

Notes: Always 0.

This is a function using undocumented features from the Apple `NSStatusItem` class, so there is not guarantee that it will work in future versions.

(Read only property)

25.1.29 View as NSViewMBS

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Sets the view to be used for this status menuitem.

Notes: Using a custom view you can draw whatever you like in the menu item.

You can set it to nil to remove the view.

(Read and Write property)

25.1.30 Width as single

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The width of the status item.

Notes: This is a function using undocumented features from the Apple `NSStatusItem` class, so there is not guarantee that it will work in future versions.

(Read only property)

25.1.31 Window as NSWindowMBS

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: The window used for this NSStatusItem.

Notes: This is a function using undocumented features from the Apple NSStatusItem class, so there is not guarantee that it will work in future versions.

Returns nil on 64 bit target.

(Read only property)

25.1.32 Events

25.1.33 Action

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: The event called when the user clicks on the statusitem.

Notes: Mouse position can be calculated based on System.MouseX/System.MouseY relative to Left/Top. Mouse status can be read using System.MouseDown.

This event is limited. You can't for example do everything like quit an application. For using quit, start a timer which will remove the menu 500ms later and than quit 500ms later.

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.

25.1.34 DoubleAction

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: The event called when the user double clicks on the statusitem.

Notes: If two Action Events happen very fast, this one is called the second time so you can e.g. react on double clicks.

Mouse position can be calculated based on System.MouseX/System.MouseY relative to Left/Top. Mouse status can be read using System.MouseDown.

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.

Chapter 26

List of Questions in the FAQ

- 27.0.1 Can anyone help me convert seconds to time in this format hh:mm:ss? 1175
- 27.0.2 Do you have plugins for Android? 1176
- 27.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1176
- 27.0.4 How to catch delete key? 1177
- 27.0.5 How to convert cmyk to rgb? 1178
- 27.0.6 How to delete a folder? 1179
- 27.0.7 How to detect if CPU is 64bit processor? 1180
- 27.0.8 How to query variant type string for a variant? 1181
- 27.0.9 How to refresh a htmlviewer on Windows? 1182
- 27.0.10 Is there an example for vector graphics in Xojo? 1183
- 27.0.11 Picture functions do not preserve resolution values? 1184
- 27.0.12 A toolbox call needs a rect - how do I give it one? 1184
- 27.0.13 API client not supported? 1184
- 27.0.14 Can I access Access Database with Java classes? 1185
- 27.0.15 Can I create PDF from Xojo Report using DynaPDF? 1186
- 27.0.16 Can I use AppleScripts in a web application? 1186
- 27.0.17 Can I use graphics class with DynaPDF? 1186
- 27.0.18 Can I use sockets on a web application? 1187
- 27.0.19 Can I use your ChartDirector plugin on a web application? 1187

- 27.0.20 Can I use your DynaPDF plugin on a web application? 1188
- 27.0.21 Can I use your plugin controls on a web application? 1189
- 27.0.22 Can you get an unique machine ID? 1189
- 27.0.23 ChartDirector: Alignment Specification 1189
- 27.0.24 ChartDirector: Color Specification 1190
- 27.0.25 ChartDirector: Font Specification 1193
- 27.0.26 ChartDirector: Mark Up Language 1197
- 27.0.27 ChartDirector: Parameter Substitution and Formatting 1201
- 27.0.28 ChartDirector: Shape Specification 1205
- 27.0.29 Copy styled text? 1206
- 27.0.30 Do you have code to validate a credit card number? 1207
- 27.0.31 Do you have plugins for X-Rite EyeOne, eXact or i1Pro? 1208
- 27.0.32 Does SQL Plugin handle stored procedures with multiple result sets? 1208
- 27.0.33 Does the plugin home home? 1208
- 27.0.34 folderitem.absolutePath is limited to 255 chars. How can I get longer ones? 1209
- 27.0.35 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? 1209
- 27.0.36 How about Plugin support for older OS X? 1210
- 27.0.37 How can I detect whether an Intel CPU is a 64bit CPU? 1211
- 27.0.38 How can I disable the close box of a window on Windows? 1212
- 27.0.39 How can I get all the environment variables from Windows? 1212
- 27.0.40 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? 1213
- 27.0.41 How can I get text from a PDF? 1213
- 27.0.42 How can I get text from a Word Document? 1213
- 27.0.43 How can I get the item string for a given file creator? 1214
- 27.0.44 How can I launch an app using it's creator code? 1215
- 27.0.45 How can I learn what shared libraries are required by a plugin on Linux? 1215
- 27.0.46 How can I validate an email address? 1217
- 27.0.47 How do I decode correctly an email subject? 1217

	1167
• 27.0.48 How do I enable/disable a single tab in a tabpanel?	1218
• 27.0.49 How do I find the root volume for a file?	1219
• 27.0.50 How do I get the current languages list?	1219
• 27.0.51 How do I get the Mac OS Version?	1220
• 27.0.52 How do I get the printer name?	1221
• 27.0.53 How do I make a metal window if RB does not allow me this?	1222
• 27.0.54 How do I make a smooth color transition?	1222
• 27.0.55 How do I read the applications in the dock app?	1223
• 27.0.56 How do I truncate a file?	1224
• 27.0.57 How do update a Finder's windows after changing some files?	1224
• 27.0.58 How to access a USB device directly?	1225
• 27.0.59 How to add icon to file on Mac?	1225
• 27.0.60 How to ask the Mac for the Name of the Machine?	1225
• 27.0.61 How to automatically enable retina in my apps?	1226
• 27.0.62 How to avoid leaks with Cocoa functions?	1226
• 27.0.63 How to avoid trouble connecting to oracle database with SQL Plugin?	1227
• 27.0.64 How to avoid ___NSAutoreleaseNoPool console messages in threads?	1227
• 27.0.65 How to bring app to front?	1228
• 27.0.66 How to bring my application to front?	1228
• 27.0.67 How to catch Control-C on Mac or Linux in a console app?	1229
• 27.0.68 How to change name of application menu?	1229
• 27.0.69 How to change the name in the menubar of my app on Mac OS X?	1230
• 27.0.70 How to check if a folder/directory has subfolders?	1230
• 27.0.71 How to check if Macbook runs on battery or AC power?	1231
• 27.0.72 How to check if Microsoft Outlook is installed?	1232
• 27.0.73 How to check on Mac OS which country or language is currently selected?	1232
• 27.0.74 How to code sign my app with plugins?	1233
• 27.0.75 How to collapse a window?	1233
• 27.0.76 How to compare two pictures?	1234

- 27.0.77 How to compile PHP library? 1236
- 27.0.78 How to convert a `BrowserType` to a `String` with `WebSession.Browser`? 1237
- 27.0.79 How to convert a `EngineType` to a `String` with `WebSession.Engine`? 1238
- 27.0.80 How to convert a `PlatformType` to a `String` with `WebSession.Platform`? 1238
- 27.0.81 How to convert a text to iso-8859-1 using the `TextEncoder`? 1239
- 27.0.82 How to convert `ChartTime` back to Xojo date? 1240
- 27.0.83 How to convert line endings in text files? 1240
- 27.0.84 How to convert picture to string and back? 1241
- 27.0.85 How to copy an array? 1242
- 27.0.86 How to copy an dictionary? 1242
- 27.0.87 How to copy parts of a movie to another one? 1242
- 27.0.88 How to create a birthday like calendar event? 1243
- 27.0.89 How to create a GUID? 1244
- 27.0.90 How to create a Mac picture clip file? 1244
- 27.0.91 How to create a PDF file in Xojo? 1245
- 27.0.92 How to create `EmailAttachment` for PDF Data in memory? 1245
- 27.0.93 How to create PDF for image files? 1246
- 27.0.94 How to CURL Options translate to Plugin Calls? 1247
- 27.0.95 How to delete file with ftp and curl plugin? 1248
- 27.0.96 How to detect display resolution changed? 1248
- 27.0.97 How to detect retina? 1249
- 27.0.98 How to disable force quit? 1249
- 27.0.99 How to disable the error dialogs from Internet Explorer on javascript errors? 1249
- 27.0.100 How to display a PDF file in Xojo? 1249
- 27.0.101 How to do a lottery in RB? 1250
- 27.0.102 How to do an asycron DNS lookup? 1251
- 27.0.103 How to draw a dashed pattern line? 1251
- 27.0.104 How to draw a nice antialiased line? 1252
- 27.0.105 How to dump java class interface? 1253

	1169
• 27.0.106 How to duplicate a picture with mask or alpha channel?	1254
• 27.0.107 How to enable assistive devices?	1255
• 27.0.108 How to encrypt a file with Blowfish?	1255
• 27.0.109 How to extract text from HTML?	1256
• 27.0.110 How to find empty folders in a folder?	1256
• 27.0.111 How to find iTunes on a Mac OS X machine fast?	1256
• 27.0.112 How to find network interface for a socket by it's name?	1257
• 27.0.113 How to find version of Microsoft Word?	1258
• 27.0.114 How to fix CURL error 60/53 on connecting to server?	1259
• 27.0.115 How to format double with n digits?	1259
• 27.0.116 How to get a time converted to user time zone in a web app?	1260
• 27.0.117 How to get an handle to the frontmost window on Windows?	1260
• 27.0.118 How to get CFAbsoluteTime from date?	1261
• 27.0.119 How to get client IP address on web app?	1261
• 27.0.120 How to get fonts to load in charts on Linux?	1261
• 27.0.121 How to get fonts to load in DynaPDF on Linux?	1262
• 27.0.122 How to get GMT time and back?	1263
• 27.0.123 How to get good crash reports?	1263
• 27.0.124 How to get list of all threads?	1264
• 27.0.125 How to get parameters from webpage URL in Xojo Web Edition?	1264
• 27.0.126 How to get the color for disabled textcolor?	1264
• 27.0.127 How to get the current free stack space?	1265
• 27.0.128 How to get the current timezone?	1266
• 27.0.129 How to get the current window title?	1267
• 27.0.130 How to get the cursor blink interval time?	1268
• 27.0.131 How to get the list of the current selected files in the Finder?	1269
• 27.0.132 How to get the Mac OS system version?	1270
• 27.0.133 How to get the Mac OS Version using System.Gestalt?	1270
• 27.0.134 How to get the screensize excluding the task bar?	1271

- 27.0.135 How to get the size of the frontmost window on Windows? 1271
- 27.0.136 How to get the source code of a HTMLViewer? 1272
- 27.0.137 How to get Xojo apps running Linux? 1272
- 27.0.138 How to handle really huge images with GraphicsMagick or ImageMagick? 1272
- 27.0.139 How to handle tab key for editable cells in listbox? 1273
- 27.0.140 How to hard link MapKit framework? 1274
- 27.0.141 How to have a PDF downloaded to the user in a web application? 1275
- 27.0.142 How to hide all applications except mine? 1275
- 27.0.143 How to hide script errors in HTMLViewer on Windows? 1276
- 27.0.144 How to hide the grid/background/border in ChartDirector? 1276
- 27.0.145 How to hide the mouse cursor on Mac? 1276
- 27.0.146 How to insert image to NSTextView or TextArea? 1276
- 27.0.147 How to jump to an anchor in a htmlviewer? 1277
- 27.0.148 How to keep a movieplayer unclickable? 1277
- 27.0.149 How to keep my web app from using 100% CPU time? 1278
- 27.0.150 How to kill a process by name? 1278
- 27.0.151 How to know how many CPUs are present? 1279
- 27.0.152 How to know the calling function? 1279
- 27.0.153 How to launch an app using it's creator code? 1280
- 27.0.154 How to launch disc utility? 1280
- 27.0.155 How to make a lot of changes to a REAL SQL Database faster? 1281
- 27.0.156 How to make a NSImage object for my retina enabled app? 1281
- 27.0.157 How to make a window borderless on Windows? 1281
- 27.0.158 How to make an alias using AppleEvents? 1282
- 27.0.159 How to make AppleScripts much faster? 1283
- 27.0.160 How to make double clicks on a canvas? 1283
- 27.0.161 How to make my Mac not sleeping? 1285
- 27.0.162 How to make my own registration code scheme? 1286
- 27.0.163 How to make small controls on Mac OS X? 1286

	1171
• 27.0.164 How to mark my Mac app as background only?	1287
• 27.0.165 How to move a file or folder to trash?	1287
• 27.0.166 How to move an application to the front using the creator code?	1288
• 27.0.167 How to move file with ftp and curl plugin?	1289
• 27.0.168 How to normalize string on Mac?	1289
• 27.0.169 How to obscure the mouse cursor on Mac?	1290
• 27.0.170 How to open icon file on Mac?	1290
• 27.0.171 How to open PDF in acrobat reader?	1290
• 27.0.172 How to open printer preferences on Mac?	1291
• 27.0.173 How to open special characters panel on Mac?	1292
• 27.0.174 How to optimize picture loading in Web Edition?	1292
• 27.0.175 How to parse XML?	1292
• 27.0.176 How to play audio in a web app?	1293
• 27.0.177 How to pretty print xml?	1294
• 27.0.178 How to print to PDF?	1294
• 27.0.179 How to query Spotlight's Last Open Date for a file?	1295
• 27.0.180 How to quit windows?	1296
• 27.0.181 How to read a CSV file correctly?	1296
• 27.0.182 How to read the command line on windows?	1297
• 27.0.183 How to render PDF pages with PDF Kit?	1297
• 27.0.184 How to restart a Mac?	1298
• 27.0.185 How to resume ftp upload with curl plugin?	1298
• 27.0.186 How to rotate a PDF page with CoreGraphics?	1299
• 27.0.187 How to rotate image with CoreImage?	1300
• 27.0.188 How to run a 32 bit application on a 64 bit Linux?	1301
• 27.0.189 How to save HTMLViewer to PDF with landscape orientation?	1301
• 27.0.190 How to save RTFD?	1301
• 27.0.191 How to save RTFD?	1302
• 27.0.192 How to scale a picture proportionally with mask?	1302

- 27.0.193 How to scale a picture proportionally? 1303
- 27.0.194 How to scale/resize a CIImageMBS? 1304
- 27.0.195 How to scale/resize a picture? 1305
- 27.0.196 How to search with regex and use unicode codepoints? 1305
- 27.0.197 How to see if a file is invisible for Mac OS X? 1306
- 27.0.198 How to set cache size for SQLite or REALSQLDatabase? 1307
- 27.0.199 How to set the modified dot in the window? 1307
- 27.0.200 How to show a PDF file to the user in a Web Application? 1307
- 27.0.201 How to show Keyboard Viewer programmatically? 1308
- 27.0.202 How to show the mouse cursor on Mac? 1309
- 27.0.203 How to shutdown a Mac? 1309
- 27.0.204 How to sleep a Mac? 1310
- 27.0.205 How to speed up rasterizer for displaying PDFs with DynaPDF? 1310
- 27.0.206 How to use PDFLib in my RB application? 1310
- 27.0.207 How to use quotes in a string? 1311
- 27.0.208 How to use Sybase in Web App? 1311
- 27.0.209 How to use the Application Support folder? 1311
- 27.0.210 How to use the IOPMCopyScheduledPowerEvents function in Xojo? 1312
- 27.0.211 How to validate a GUID? 1315
- 27.0.212 How to walk a folder hierarchie non recursively? 1315
- 27.0.213 I got this error: PropVal, QDPictMBS.Name (property value), Type mismatch error. Expected CGDataProviderMBS, but got Variant, Name:QDPictMBS 1316
- 27.0.214 I registered the MBS Plugins in my application, but later the registration dialog is shown. 1316
- 27.0.215 I want to accept Drag & Drop from iTunes 1317
- 27.0.216 I'm drawing into a listbox but don't see something. 1319
- 27.0.217 I'm searching for a method or so to move a window from position x.y to somewhere else on the screen. 1319
- 27.0.218 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? 1319
- 27.0.219 Is the fn key on a powerbook keyboard down? 1320

	1173
• 27.0.220 Is there a case sensitive Dictionary?	1320
• 27.0.221 Is there a way to use the MBS plugin to get only the visible item and folder count on a volume?	1321
• 27.0.222 Is there an easy way I can launch the Displays preferences panel?	1321
• 27.0.223 List of Windows Error codes?	1322
• 27.0.224 Midi latency on Windows problem?	1322
• 27.0.225 My Xojo Web App does not launch. Why?	1322
• 27.0.226 SQLiteDatabase not initialized error?	1323
• 27.0.227 Textconverter returns only the first x characters. Why?	1323
• 27.0.228 The type translation between CoreFoundation/Foundation and Xojo data types.	1324
• 27.0.229 Uploaded my web app with FTP, but it does not run on the server!	1326
• 27.0.230 What classes to use for hotkeys?	1326
• 27.0.231 What do I need for Linux to get picture functions working?	1326
• 27.0.232 What does the NAN code mean?	1327
• 27.0.233 What font is used as a 'small font' in typical Mac OS X apps?	1327
• 27.0.234 What is last plugin version to run on Mac OS X 10.4?	1328
• 27.0.235 What is last plugin version to run on PPC?	1328
• 27.0.236 What is last version of the plugins for macOS 32-bit?	1329
• 27.0.237 What is the difference between Timer and WebTimer?	1329
• 27.0.238 What is the list of Excel functions?	1329
• 27.0.239 What is the replacement for PluginMBS?	1330
• 27.0.240 What to do on Xojo reporting a conflict?	1330
• 27.0.241 What to do with a NSImageCacheException?	1331
• 27.0.242 What to do with MySQL Error 2014?	1331
• 27.0.243 What to do with SQL Plugin reporting Malformed string as error?	1331
• 27.0.244 Where is CGGetActiveDisplayListMBS?	1331
• 27.0.245 Where is CGGetDisplaysWithPointMBS?	1332
• 27.0.246 Where is CGGetDisplaysWithRectMBS?	1332
• 27.0.247 Where is CGGetOnlineDisplayListMBS?	1332
• 27.0.248 Where is GetObjectClassNameMBS?	1332

- 27.0.249 Where is NetworkAvailableMBS? 1332
- 27.0.250 Where is StringHeight function in DynaPDF? 1333
- 27.0.251 Where is XLSDocumentMBS class? 1333
- 27.0.252 Where to get information about file formats? 1333
- 27.0.253 Where to register creator code for my application? 1334
- 27.0.254 Which Mac OS X frameworks are 64bit only? 1334
- 27.0.255 Which plugins are 64bit only? 1335
- 27.0.256 Why application doesn't launch because of a missing ddraw.dll!? 1335
- 27.0.257 Why application doesn't launch because of a missing shlwapi.dll!? 1335
- 27.0.258 Why do I hear a beep on keydown? 1335
- 27.0.259 Why does folderitem.item return nil? 1335
- 27.0.260 Why doesn't showurl work? 1335
- 27.0.261 Why don't the picture functions not work on Linux? 1336
- 27.0.262 Why have I no values in my chart? 1336
- 27.0.263 Will application size increase with using plugins? 1336
- 27.0.264 XLS: Custom format string guidelines 1336
- 27.0.265 Xojo doesn't work with your plugins on Windows 98. 1337
- 27.0.266 Xojo or my RB application itself crashes on launch on Mac OS Classic. Why? 1338

Chapter 27

The FAQ

27.0.1 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)

27.0.2 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.

27.0.3 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:

- 27.0.4 How to catch delete key? 1177
- 27.0.5 How to convert cmyk to rgb? 1178
- 27.0.6 How to delete a folder? 1179
- 27.0.7 How to detect if CPU is 64bit processor? 1180
- 27.0.8 How to query variant type string for a variant? 1181
- 27.0.9 How to refresh a htmlviewer on Windows? 1182

27.0.4 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:

- 27.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1176

- 27.0.5 How to convert cmyk to rgb? 1178
- 27.0.6 How to delete a folder? 1179
- 27.0.7 How to detect if CPU is 64bit processor? 1180
- 27.0.8 How to query variant type string for a variant? 1181
- 27.0.9 How to refresh a htmlviewer on Windows? 1182

27.0.5 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:

- 27.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1176
- 27.0.4 How to catch delete key? 1177
- 27.0.6 How to delete a folder? 1179
- 27.0.7 How to detect if CPU is 64bit processor? 1180
- 27.0.8 How to query variant type string for a variant? 1181
- 27.0.9 How to refresh a htmlviewer on Windows? 1182

27.0.6 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:

- 27.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1176
- 27.0.4 How to catch delete key? 1177
- 27.0.5 How to convert cmyk to rgb? 1178
- 27.0.7 How to detect if CPU is 64bit processor? 1180
- 27.0.8 How to query variant type string for a variant? 1181
- 27.0.9 How to refresh a htmlviewer on Windows? 1182

27.0.7 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:

- 27.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1176
- 27.0.4 How to catch delete key? 1177
- 27.0.5 How to convert cmyk to rgb? 1178
- 27.0.6 How to delete a folder? 1179
- 27.0.8 How to query variant type string for a variant? 1181
- 27.0.9 How to refresh a htmlviewer on Windows? 1182

27.0.8 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:

- 27.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1176
- 27.0.4 How to catch delete key? 1177
- 27.0.5 How to convert cmyk to rgb? 1178
- 27.0.6 How to delete a folder? 1179
- 27.0.7 How to detect if CPU is 64bit processor? 1180
- 27.0.9 How to refresh a htmlviewer on Windows? 1182

27.0.9 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:

- 27.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1176
- 27.0.4 How to catch delete key? 1177
- 27.0.5 How to convert cmyk to rgb? 1178

- 27.0.6 How to delete a folder? 1183
 - 27.0.7 How to detect if CPU is 64bit processor? 1179
 - 27.0.8 How to query variant type string for a variant? 1180
- 1181

27.0.10 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
```

27.0.11 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.

27.0.12 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
```

27.0.13 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.

27.0.14 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>

27.0.15 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.

27.0.16 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.

27.0.17 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)

27.0.18 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.

27.0.19 Can I use your ChartDirector plugin on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, our ChartDirector plugin works just fine on the Xojo Web Edition.

Example:

```
// The data for the pie chart
dim data(-1) as Double=array(55.0, 18.0, 25.0, 22.0, 18.0, 30.0, 35.0)

// The labels for the pie chart, Words are choosen random to check font!
dim labels(-1) as string=array("Germany", "Italy", "France", "Spain", "UK", "Poland", "Russia")

// The colors to use for the sectors
dim colors(-1) as Integer

colors.Append &h66aaee
colors.Append &heebb22
colors.Append &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.

27.0.20 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.

27.0.21 Can I use your plugin controls on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: No.

27.0.22 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.

27.0.23 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.

27.0.24 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.

Metal Color
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.

Gradient Color
A gradient color is a color that changes progressively across a direction.

Gradient colors are created using `BaseChart.gradientColor`, `BaseChart.gradientColor2`, `DrawArea.gradientColor` and `DrawArea.gradientColor2`. The `gradientColor` method creates a 2-point gradient color that changes from color A to color B. The `gradientColor2` method creates a multi-point gradient colors that changes from color A to B to C

These methods return a 32-bit integer acting as a handle to the gradient color. The handle can be used in any `ChartDirector` API that expects a color as its input.

One common use of multi-point gradient colors is to define colors that have metallic look and feel. Please refer to `DrawArea.gradientColor2` for details.

Dash Line Colors
A dash line color is a color that switches on and off periodically. When used to draw a line, the line will appear as a dash line.

Dash line colors are created using `BaseChart.dashLineColor` and `DrawArea.dashLineColor`. They accept a line color and a dash pattern code as arguments, and return a 32-bit integer acting as a handle to the dash line color. The handle can be used in any `ChartDirector` API that expects a color as its input.

Zone Colors
A zone color is for XY charts only. It is a color that automatically changes upon reaching a data threshold value along the x-axis or y-axis. Zone colors are created using `Layer.xZoneColor`, `Layer.yZoneColor`, `XYChart.xZoneColor` or `XYChart.yZoneColor`.

Palette Colors
Palette colors are colors of the format `FFFFnnnn`, where the least significant 16 bits (`nnnn`) are the index to the palette. A palette is simply an array of colors. For a palette color, the actual color is obtained by

looking up the palette using the index. For example, the color FFFF0001 is the second color in the palette (first color is index 0).

The colors in the palette can be ARGB colors or "dynamic" colors (pattern, gradient and dash line colors).

The first eight palette colors have special significance. The first three palette colors are the background color, default line color, and default text color of the chart. The 4th to 7th palette colors are reserved for future use. The 8th color is a special dynamic color that is equal to the data color of the "current data set".

The 9th color (index = 8) onwards are used for automatic data colors. For example, in a pie chart, if the sector colors are not specified, ChartDirector will automatically use the 9th color for the first sector, the 10th color for the second sector, and so on. Similarly, for a multi-line chart, if the line colors are not specified, ChartDirector will use the 9th color for the first line, the 10th color for the second line, and so on.

The ChartDirector API defines several constants to facilitate using palette colors.

ConstantValueDescription

Palette	FFFF0000	The starting point of the palette. The first palette color is (Palette + 0). The nth palette color is (Palette + n - 1).
BackgroundColor	FFFF0000	The background color.
LineColor	FFFF0001	The default line color.
TextColor	FFFF0002	The default text color.
[Reserved]	FFFF0003 - FFFF0006	These palette positions are reserved. Future versions of ChartDirector may use these palette positions for colors that have special significance.
SameAsMainColor	FFFF0007	A dynamic color that is equal to the data color of the current data set. This color is useful for objects that are associated with data sets. For example, in a pie chart, if the sector label background color is SameAsMainColor, its color will be the same as the corresponding sector color.
DataColor	FFFF0008	The starting point for the automatic data color allocation.

When a chart is created, it has a default palette. You may modify the palette using BaseChart.setColor, BaseChart.setColors, or BaseChart.setColors2.

The advantages of using palette colors are that you can change the color schemes of the chart in one place. ChartDirector comes with several built-in palettes represented by the following predefined constants.

ConstantDescription

27.0.25 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.

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.

Font Size
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.

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.

Font Color
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 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.

27.0.26 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.

27.0.27 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.

27.0.28 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
```

27.0.29 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.

27.0.30 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)

27.0.31 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.

27.0.32 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.

27.0.33 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.

27.0.34 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
```

27.0.35 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.

27.0.36 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.

27.0.37 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.

27.0.38 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.

27.0.39 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.

27.0.40 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.

27.0.41 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.

27.0.42 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.

27.0.43 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.

27.0.44 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
```

27.0.45 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.

27.0.48 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.

27.0.49 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
```

27.0.50 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 CFObjctMBS
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.

27.0.51 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.

27.0.52 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.

27.0.53 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.

27.0.54 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.

27.0.55 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.

27.0.56 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.

27.0.57 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.

27.0.58 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.

27.0.59 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.

27.0.60 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).

27.0.61 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.

27.0.62 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.

27.0.63 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>
```

27.0.64 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

27.0.65 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.

27.0.66 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)

27.0.67 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.

27.0.68 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.

27.0.69 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/>.

27.0.70 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.

27.0.71 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.

27.0.72 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
```

27.0.73 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.

27.0.74 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.

27.0.75 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.

27.0.76 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.

27.0.77 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.

27.0.78 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
```

27.0.79 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
```

27.0.80 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

27.0.81 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.

27.0.82 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.

27.0.83 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.

27.0.84 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

27.0.85 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.

27.0.86 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.

27.0.87 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.

27.0.88 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.

27.0.89 How to create a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the UUIDMBS class for this.

27.0.90 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.

27.0.91 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.

27.0.92 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.

27.0.93 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.

27.0.94 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)

27.0.95 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.

27.0.96 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".

27.0.97 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)
```

27.0.98 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.

27.0.99 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.

27.0.100 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.

27.0.101 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

```

27.0.102 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.

27.0.103 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.

27.0.104 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.

27.0.105 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

27.0.106 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.

27.0.107 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.

27.0.108 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.

27.0.109 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.

27.0.110 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
```

27.0.111 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
```

27.0.112 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.

27.0.113 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".

27.0.114 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/>

27.0.115 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.

27.0.116 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
```

27.0.117 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
```

27.0.118 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.

27.0.119 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
```

27.0.120 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.

27.0.121 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.

27.0.122 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.

27.0.123 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>

27.0.124 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.

27.0.125 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.

27.0.126 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" ...

27.0.127 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.

27.0.128 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

```

27.0.129 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

```

27.0.130 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.

27.0.131 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

27.0.132 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

```

27.0.133 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.

27.0.134 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.

27.0.135 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!

27.0.136 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
```

27.0.137 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.

27.0.138 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.

27.0.139 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.

27.0.140 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.

27.0.141 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.

27.0.142 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
```

27.0.143 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).

27.0.144 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.

27.0.145 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.

27.0.146 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.

27.0.147 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

```

27.0.148 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
```

27.0.149 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.

27.0.150 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.

27.0.151 How to know how many CPUs are present?

Plugin Version: all, Platform: macOS.

Answer: Try this function:

Example:

```
Function GetCPUCount() as Integer
Declare Function MPPProcessors Lib "Carbon" () as Integer
```

```
Return MPPProcessors()
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.

27.0.152 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.

27.0.153 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
```

27.0.154 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.

27.0.155 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.SQLiteExecute "BEGIN TRANSACTION"
// Do some Stuff
db.SQLiteExecute "END TRANSACTION"
```

Notes: This can increase speed by some factors.

27.0.156 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.

27.0.157 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
```

27.0.158 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

27.0.159 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

```

27.0.160 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

27.0.161 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.

27.0.162 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?

27.0.163 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))

```

27.0.164 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.

27.0.165 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.

27.0.166 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)

27.0.167 How to move file with ftp and curl plugin?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can set post/pre quotes to have ftp commands executed before or after the download/upload.

Example:

```
dim d as CURLMBS // your curl object

// rename/move file
dim ws() As String
ws.Append "RNFR Temp.txt"
ws.append "RNTO MyFile.txt"

d.SetOptionPostQuote(ws)
```

Notes: Use SetOptionPostQuote, SetOptionPreQuote or SetOptionQuote.

The ftp commands you pass here are native ftp commands and not the commands you use with ftp applications. So rename is two commands. First RNFR to tell where to rename from and second RNTD with the new file name. To delete use DELE and the file path.

27.0.168 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.

27.0.169 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.

27.0.170 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
```

27.0.171 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.

27.0.172 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

```

27.0.173 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.

27.0.174 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.

27.0.175 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.

27.0.176 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)+"");")
```

27.0.177 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>

27.0.178 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.

27.0.179 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.

27.0.180 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.

27.0.181 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.

27.0.182 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.

27.0.183 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.

27.0.184 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

```

27.0.185 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.

27.0.186 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.

27.0.187 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
```

27.0.188 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.

27.0.189 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.

27.0.190 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.

27.0.191 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.

27.0.192 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.

27.0.193 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)

27.0.194 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.

27.0.195 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.

27.0.196 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

```

27.0.197 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

```

27.0.198 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.

27.0.199 How to set the modified dot in the window?

Plugin Version: all, Platform: macOS.

Answer: Try this declares:

Example:

```

window1.ModifiedMBS=true

```

27.0.200 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.

27.0.201 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
```

27.0.202 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.

27.0.203 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
```

27.0.204 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
```

27.0.205 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.

27.0.206 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.

27.0.207 How to use quotes in a string?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Just double them.

Example:

```
msgbox "This String contains ""quotes"""
```

27.0.208 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
```

27.0.209 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!

27.0.210 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.

27.0.211 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".

27.0.212 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.

27.0.213 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.

27.0.214 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.

27.0.215 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.

27.0.216 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.

27.0.217 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.

27.0.218 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. pdfib for some classes)

27.0.219 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.

27.0.220 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
```

27.0.221 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.

27.0.222 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
```

27.0.223 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>

27.0.224 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)

27.0.225 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?

27.0.226 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.

27.0.227 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.

27.0.228 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.

27.0.229 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.

27.0.230 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.

27.0.231 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.

27.0.232 What does the NAN code mean?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

27.0.233 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

```

27.0.234 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.

27.0.235 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.

27.0.236 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.

27.0.237 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.

27.0.238 What is the list of Excel functions?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Below a list of function names known by LibXL.

Notes: LibXL parses the functions and writes tokens to the excel file. So even if Excel can do more functions, we can only accept the ones known by LibXL.

ABS, ABSREF, ACOS, ACOSH, ACTIVE.CELL, ADD.BAR, ADD.COMMAND, ADD.MENU, ADD.TOOLBAR, ADDRESS, AND, APP.TITLE, AREAS, ARGUMENT, ASC, ASIN, ASINH, ATAN, ATAN2, ATANH, AVEDEV, AVERAGE, AVERAGEA, BAHTTEXT, BETADIST, BETAINV, BINOMDIST, BREAK, CALL, CALLER, CANCEL.KEY, CEILING, CELL, CHAR, CHECK.COMMAND, CHIDIST, CHIINV, CHITEST, CHOOSE, CLEAN, CODE, COLUMN, COLUMNS, COMBIN, CONCATENATE, CONFIDENCE, CORREL, COS, COSH, COUNT, COUNTA, COUNTBLANK, COUNTIF, COVAR, CREATE.OBJECT, CRITBINOM, CUSTOM.REPEAT, CUSTOM.UNDO, DATE, DATEDIF, DATESTRING, DATEVALUE, DAVERAGE, DAY, DAYS360, DB, DBCS, DCOUNT, DCOUNTA, DDB, DEGREES, DELETE.BAR, DELETE.COMMAND, DELETE.MENU, DELETE.TOOLBAR, DEREf, DEVSQ, DGET, DIALOG.BOX, DIRECTORY, DMAX, DMIN, DOCUMENTS, DOLLAR, DPRODUCT, DSTDEV, DSTDEVP, DSUM, DVAR, DVARP, ECHO, ELSE, ELSE.IF, ENABLE.COMMAND, ENABLE.TOOL, END.IF, ERROR, ERROR.TYPE, EVALUATE, EVEN, EXACT, EXEC, EXECUTE, EXP, EXPONDIST, FACT, FALSE, FCLOSE, FDIST, FILES, FIND, FINDB, FINV, FISHER, FISHERINV, FIXED, FLOOR, FOPEN, FOR, FOR.CELL, FORECAST,

FORMULA.CONVERT, FPOS, FREAD, FREADLN, FREQUENCY, FSIZE, FTEST, FV, FWRITE, FWRITELN, GAMMADIST, GAMMAINV, GAMMALN, GEOMEAN, GET.BAR, GET.CELL, GET.CHART.ITEM, GET.DEF, GET.DOCUMENT, GET.FORMULA, GET.LINK.INFO, GET.MOVIE, GET.NAME, GET.NOTE, GET.OBJECT, GET.PIVOT.FIELD, GET.PIVOT.ITEM, GET.PIVOT.TABLE, GET.TOOL, GET.TOOLBAR, GET.WINDOW, GET.WORKBOOK, GET.WORKSPACE, GETPIVOTDATA, GOTO, GROUP, GROWTH, HALT, HARMEAN, HELP, HLOOKUP, HOUR, HYPERLINK, HYPGEOMDIST, IF, INDEX, INDIRECT, INFO, INITIATE, INPUT, INT, INTERCEPT, IPMT, IRR, ISBLANK, ISERR, ISERROR, ISLOGICAL, ISNA, ISNONTEXT, ISNUMBER, ISPMT, ISREF, ISTEXT, ISTHAIDIGIT, KURT, LARGE, LAST.ERROR, LEFT, LEFTB, LEN, LENB, LINEST, LINKS, LN, LOG, LOG10, LOGEST, LOGINV, LOGNORMDIST, LOOKUP, LOWER, MATCH, MAX, MAXA, MDETERM, MEDIAN, MID, MIDB, MIN, MINA, MINUTE, MINVERSE, MIRR, MMULT, MOD, MODE, MONTH, MOVIE.COMMAND, N, NA, NAMES, NEGBINOMDIST, NEXT, NORMDIST, NORMINV, NORMSDIST, NORMSINV, NOT, NOTE, NOW, NPER, NPV, NUMBERSTRING, ODD, OFFSET, OPEN.DIALOG, OPTIONS.LISTS.GET, OR, PAUSE, PEARSON, PERCENTILE, PERCENTRANK, PERMUT, PHONETIC, PI, PIVOT.ADD.DATA, PMT, POISSON, POKE, POWER, PPMT, PRESS.TOOL, PROB, PRODUCT, PROPER, PV, QUARTILE, RADIANS, RAND, RANK, RATE, 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.

27.0.239 What is the replacement for PluginMBS?

Plugin Version: all, Platform: macOS.

Answer: Use the SoftDeclareMBS class to load libraries dynamically.

27.0.240 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.

27.0.241 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.

27.0.242 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.

27.0.243 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.

27.0.244 Where is CGGetActiveDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetActiveDisplayList.

27.0.245 Where is CGGetDisplaysWithPointMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithPoint.

27.0.246 Where is CGGetDisplaysWithRectMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithRect.

27.0.247 Where is CGGetOnlineDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetOnlineDisplayList.

27.0.248 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.

27.0.249 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.macs.w.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.

27.0.250 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.

27.0.251 Where is XLSDocumentMBS class?

Plugin Version: all, Platform: macOS.

Answer: This class has been removed in favor of XLBookMBS class.

Notes: These classes have been removed: XLCellMBS, XLSDocumentMBS, XLSFormatRecordMBS, XLSMergedCellsMBS, XLSRowMBS and XLSSheetMBS.

27.0.252 Where to get information about file formats?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Please visit this web page:

<http://www.wotsit.org>

27.0.253 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>

27.0.254 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.

27.0.255 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.

27.0.256 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.

27.0.257 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.

27.0.258 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.

27.0.259 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.

27.0.260 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.

27.0.261 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.

27.0.262 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.

27.0.263 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.

27.0.264 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

27.0.265 Xojo doesn't work with your plugins on Windows 98.

Plugin Version: all, Platform: Windows.

Answer: Please upgrade your Windows version.

27.0.266 Xojo or my RB application itself crashes on launch on Mac OS Classic.
Why?

Plugin Version: all.

Answer:

You may check if the application has enough memory to be loaded.

RB should have on Mac OS Classic more than 20 MB of RAM.

I 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]

textasciitilde ' for no thousand separator. The default is 'textasciitilde ', which can be modified using `BaseChart.setNumberFormat`.

[c]

The thousand separator. Should be a non-alphanumeric character (not 0-9, A-Z, a-z). Use '.

The decimal point character. The default is '.', which can be modified using `BaseChart.setNumberFormat`.

[d]

textasciitilde ' for no negative sign character. The default is '-', which can be modified using `BaseChart.setNumberFormat`.

The negative sign character. Use '-'

Parameter	Description
yyyy	The year in 4 digits (e.g. 2002)
yyy	The year showing only the least significant 3 digits (e.g. 002 for the year 2002)
yy	The year showing only the least significant 2 digits (e.g. 02 for the year 2002)
y	The year showing only the least significant 1 digits (e.g. 2 for the year 2002)
mmm	The month formatted as its name. The default is to use the first 3 characters of the english month name (Jan, Feb, Mar ...). The names can be configured using <code>BaseChart.setMonthNames</code> .
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 <code>BaseChart.setMonthNames</code> .
MM	The first 2 characters of the month name converted to upper case. The names can be configured using <code>BaseChart.setMonthNames</code> .
M	The first character of the month name converted to upper case. The names can be configured using <code>BaseChart.setMonthNames</code> .
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 <code>BaseChart.setWeekDayNames</code> .
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 <code>BaseChart.setAMPM</code> .

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,,