

air.net.

air.net.ServiceMonitor

Public Properties

available : Boolean
lastUpdated : Date

Public Methods

ServiceMonitor()
augmentPrototype(proto:Object):void

Protected Methods

checkStatus():void

pollInterval : Number
running : Boolean
start():void
stop():void
toString():String

Events

status

air.net.SocketMonitor

Public Properties

host : String
port : int

Public Methods

SocketMonitor(host:String, port:int) toString():String

Protected Methods

checkStatus():void

air.net.URLMonitor

Public Properties

acceptableStatuses : Array
urlRequest : URLRequest

Public Methods

URLMonitor(urlRequest:URLRequest, acceptableStatuses:Array = null)
toString():String

Protected Methods

checkStatus():void

flash.data.SQLDatabase

Public Properties

caseSensitiveLike : Boolean
columnNameStyle : String
connected : Boolean
inTransaction : Boolean
totalChanges : uint
version : Number

Public Methods

SQLDatabase()
addEventListener(type:String, listener:Function, useCapture:Boolean = false, priority:int = 0, useWeakReference:Boolean = false):void
analyze(resourceName:String = null):void
attach(name:String, file:File = null, autoCreate:Boolean = true):void
begin(option:String = null):void
cancel():void
clean():void
close():void
commit():void
deanalyze():void
detach(name:String):void
getFile():File
open(file:File = null, autoCreate:Boolean = true, autoClean:Boolean = false, pageSize:int = -1):void
removeEventListener(type:String, listener:Function, useCapture:Boolean = false):void
rollback():void

Events

status

Public Constants

DEFAULT_COLUMN_NAMES : String = "defaultColumnNames"
LOCK_DEFERRED : String = "deferred"
LOCK_EXCLUSIVE : String = "exclusive"
LOCK_IMMEDIATE : String = "immediate"
LONG_COLUMN_NAMES : String = "longColumnNames"
SHORT_COLUMN_NAMES : String = "shortColumnNames"

flash.data.SQLStatement

Public Properties

executing : Boolean
itemClass : Class
parameters : Object
prepared : Boolean
sqlDatabase : SQLDatabase
text : String

Public Methods

SQLStatement()
clearParameters():void
execute(prefetch:int = -1, responder:Responder = null):void
next(prefetch:int = -1, responder:Responder = null):void
prepare(responder:Responder = null):void
reset(responder:Responder = null):void

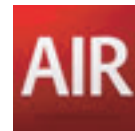
Events

result
status

flash.data.

AIR Cheatsheet

actionscripcheatsheet.com



adobe.com/go/air

flash.desktop.

flash.desktop.ClipboardManager

Public Properties

data : TransferableData

Public Methods

accessClipboard(closure:Function):void

flash.desktop.DragActions

Public Constants

COPY : String = "copy"
LINK : String = "link"
MOVE : String = "move"
NONE : String = "none"

flash.desktop.DragManager

Public Properties

dragInitiator : DisplayObject
dropAction : String

Public Methods

acceptDragDrop(target:DisplayObject):void
doDrag(initiator:DisplayObject, transferable:TransferableData, dragImage:BitmapData = null, offset:Point = null, actionsAllowed:DragOptions = null):void
isDragging():Boolean

Events

nativeDragComplete
nativeDragDrop
nativeDragEnter
nativeDragExit
nativeDragOver
nativeDragStart

flash.desktop.DragOptions

Public Properties

allowCopy : Boolean = true
allowLink : Boolean = true
allowMove : Boolean = true

Public Methods

toString():String

flash.desktop.Icon

Public Properties

bitmaps : Array

flash.desktop.TransferableData

Public Properties

formats : Array

Public Methods

TransferableData()
addData(data:Object, format:String, serializable:Boolean = true):void
addHandler(handler:Function, format:String, serializable:Boolean = true):void
dataForFormat(format:String, transferMode:String):Object
hasFormat(format:String):Boolean
removeData(format:String):void

flash.desktop.TransferableFormats

Public Constants

BITMAP_FORMAT : String = "bitmap"
FILE_LIST_FORMAT : String = "file list"
TEXT_FORMAT : String = "text"
URL_FORMAT : String = "url"

flash.desktop.TransferableTransferMode

Public Constants

CLONE_ONLY : String = "cloneOnly"
CLONE_PREFERRED : String = "clonePreferred"
ORIGINAL_ONLY : String = "originalOnly"
ORIGINAL_PREFERRED : String = "originalPreferred"

flash.display.NativeMenu

Public Properties

numItems : uint
parent : NativeMenu

Public Methods

addItem(item:NativeMenuItem):NativeMenuItem
addItemAt(item:NativeMenuItem, index:uint):NativeMenuItem
containsItem(item:NativeMenuItem):Boolean
getItemAt(index:uint):NativeMenuItem
getItemByName(name:String):NativeMenuItem
getItemIndex(item:NativeMenuItem):int
removeItem(item:NativeMenuItem):NativeMenuItem
removeItemAt(index:int):NativeMenuItem
setItemIndex(item:NativeMenuItem, index:uint):void

Events

DISPLAY
DISPLAYING
SELECT

flash.display.NativeMenuItem

Public Properties

data : Object	label : String
enabled : Boolean	labelFont : Font
icon : Bitmap	menu : NativeMenu
isSeparator : Boolean	mnemonicIndex : uint
keyEquivalent : String	name : String
keyEquivalentModifiers : Array	submenu : NativeMenu

Public Methods

NativeMenuItem(label:String = "", isSeparator:Boolean = false)

Events

DISPLAY
DISPLAYING
SELECT

flash.display.NativeWindow

Public Properties

alwaysInFront : Boolean	minSize : Point
bounds : Rectangle	stage : Stage
closed : Boolean	title : String
displayState : String	visible : Boolean
height : Number	width : Number
initOptions : NativeWindowInitOptions	x : Number
maxSize : Point	y : Number

Public Methods

NativeWindow(visible:Boolean, initOptions:NativeWindowInitOptions)
close():void
globalToScreen(globalPoint:Point):Point
maximize():void
minimize():void
restore():void
startMove():void
startResize(edgeOrCorner:String):void

Events

activate	displayStateChanging
close	move
closing	moving
deactivate	resize
displayStateChange	resizing

flash.display.NativeWindowDisplayState

Public Constants

MAXIMIZED : String = "maximized"
MINIMIZED : String = "minimized"
NORMAL : String = "normal"

flash.display.NativeWindowInitOptions

Public Properties

appearsInWindowMenu : Boolean	owner : NativeWindow
hasMenu : Boolean	resizable : Boolean
maximizable : Boolean	systemChrome : String
minimizable : Boolean	transparent : Boolean
	type : String

Public Methods

NativeWindowInitOptions()

flash.display.NativeWindowResize

Public Constants

BOTTOM : String = "B"
BOTTOM_LEFT : String = "BL"
BOTTOM_RIGHT : String = "BR"
LEFT : String = "L"
NONE : String = ""
RIGHT : String = "R"
TOP : String = "T"
TOP_LEFT : String = "TL"
TOP_RIGHT : String = "TR"

flash.display.NativeWindowSystemChrome

Public Constants

ALTERNATE : String = "alternate"
NONE : String = "none"
STANDARD : String = "standard"
UTILITY : String = "utility"

flash.display.NativeWindowType

Public Constants

LIGHTWEIGHT : String = "lightweight"
NORMAL : String = "normal"
UTILITY : String = "utility"

AIR Application Descriptor File

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
Ensure the appID attribute is fully qualified to prevent
two AIR apps with the same name from overlapping
-->
<application
    xmlns="http://ns.adobe.com/air/application/1.0.M4"
    appId="com.actionscripcheatsheet.aircheatsheet"
    version="1.0 Beta">

    <!--
    Application name, shown by OS to user on menus etc.
    -->
    <name>AIR Cheatsheet</name>

    <!-- Application title shown during installation -->
    <title>AIR Cheatsheet Installation</title>

    <description>
        Quick reference for Adobe AIR. ActionScript
        3.0 packages and classes for Adobe AIR Beta 1.
    </description>

    <copyright>
        © Sean Moore, SeanTheFlashGuy.com
    </copyright>

    <rootContent
        systemChrome="standard"
        transparent="false"
        visible="true">
        [SWF reference is generated]
    </rootContent>

</application>
```

flash.events.FileListEvent*Public Properties*

files : Array

Public Methods

FileListEvent(type:String,
bubbles:Boolean = false,
cancelable:Boolean = false,
files:Array = null)

Public Constants

DIRECTORY_LISTING : String = "directoryListing"
SELECT_MULTIPLE : String = "selectMultiple"

flash.events.HTMLUncaughtJavaScriptExceptionEvent*Public Properties*

exceptionValue : *
stackTrace : Array

Public Methods

HTMLUncaughtJavaScriptExceptionEvent(exceptionValue:*)
clone():Event

Public Constants

UNCAUGHT_JAVASCRIPT_EXCEPTION : * = uncaughtJavaScriptException

flash.events.HTTPStatusEvent*Public Properties*

responseURL : String
responseHeaders : Array status : int

Public Methods

HTTPStatusEvent(type:String, bubbles:Boolean = false,
cancelable:Boolean = false, status:int = 0)

clone():Event
toString():String

Public Constants

HTTP_RESPONSE_STATUS : String = "httpResponseStatus"
HTTP_STATUS : String = "httpStatus"

flash.events.InvokeEvent*Public Properties*

arguments : Array currentDirectory : File

Public Methods

InvokeEvent(type:String,
bubbles:Boolean = false,
cancelable:Boolean = false,
dir:File = null,
argv:Array = null)

clone():Event

Public Constants

INVOKE : String = "invoke"

flash.events.NativeDragEvent*Public Properties*

actionsAllowed : DragOptions dropAction : String
transferable : TransferableData

Public Methods

NativeDragEvent(type:String,
bubbles:Boolean = false,
cancelable:Boolean = true,
localX:Number = 0,
localY:Number = 0,
transferable:TransferableData = null,
actionsAllowed:DragOptions = null,
dropAction:String = null,
ctrlKey:Boolean = false,
altKey:Boolean = false,
shiftKey:Boolean = false)

clone():Event
toString():String

Public Constants

NATIVE_DRAG_COMPLETE : String = "nativeDragComplete"
NATIVE_DRAG_DROP : String = "nativeDragDrop"
NATIVE_DRAG_ENTER : String = "nativeDragEnter"
NATIVE_DRAG_EXIT : String = "nativeDragExit"
NATIVE_DRAG_OVER : String = "nativeDragOver"
NATIVE_DRAG_START : String = "nativeDragStart"

flash.events.NativeWindowBoundsEvent*Public Properties*

afterBounds : Rectangle
beforeBounds : Rectangle

Public Methods

NativeWindowBoundsEvent(type:String,
bubbles:Boolean = false, cancelable:Boolean = false,
beforeBounds:Rectangle = null,
afterBounds:Rectangle = null)

clone():Event
toString():String

Public Constants

MOVE : String = "move" RESIZE : String = "resize"
MOVING : String = "moving" RESIZING : String = "resizing"

flash.events.NativeWindowDisplayStateEvent*Public Properties*

afterDisplayState : String beforeDisplayState : String

Public Methods

NativeWindowDisplayStateEvent(type:String, bubbles:Boolean = true,
cancelable:Boolean = false,
beforeDisplayState:String = "",
afterDisplayState:String = "")

clone():Event
toString():String

Public Constants

DISPLAY_STATE_CHANGE : String = "displayStateChange"
DISPLAY_STATE_CHANGING : String = "displayStateChanging"

flash.events.NativeWindowErrorEvent*Public Methods*

NativeWindowErrorEvent(type:String, bubbles:Boolean = false,
cancelable:Boolean = false,
text:String = "", id:int = 0)

clone():Event
toString():String

Public Constants

INVALID_BOUNDS_ERROR : String = "invalidBoundsError"

flash.events.OutputProgressEvent*Public Properties*

bytesPending : Number bytesTotal : Number

Public Methods

OutputProgressEvent(type:String, bubbles:Boolean = false,
cancelable:Boolean = false,
bytesPending:Number = 0,
bytesTotal:Number = 0)

clone():Event
toString():String

Public Constants

OUTPUT_PROGRESS : String = "outputProgress"

flash.events.SQLResultEvent*Public Properties*

complete : Boolean data : Array

Public Methods

SQLResultEvent(type:String, bubbles:Boolean = false,
cancelable:Boolean = false, code:String = "",
level:String = "", details:String = "",
data:Array = null, complete:Boolean = true,
changeCount:uint = 0, totalTime:uint = 0)

Public Constants

RESULT : String = "result"

* flash.events.SQLStatusEvent

Public Properties

details : String

Public Methods

SQLStatusEvent(type:String,
bubbles:Boolean = false,
cancelable:Boolean = false,
code:String = "",
level:String = "",
details:String = "")

Public Constants

STATUS : String = "status"

flash.events.SQLStatusEventCodes

Public Constants

ABORT_ERROR : String = "abortError"
ANALYZE : String = "analyze"
ATTACH : String = "attach"
AUTHORIZATION_ERROR : String = "authorizationError"
BEGIN : String = "begin"
BIND_RANGE_ERROR : String = "bindRangeError"
BUSY_ERROR : String = "busyError"
CANNOT_OPEN_ERROR : String = "cannotOpenError"
CLEAN : String = "clean"
CLOSE : String = "close"
COMMIT : String = "commit"
CONSTRAINT_ERROR : String = "constraintError"
CORRUPT_ERROR : String = "corruptError"
DEANALYZE : String = "deanalyze"
DETACH : String = "detach"
EXECUTE : String = "execute"
FORMAT_ERROR : String = "formatError"
FULL_ERROR : String = "fullError"
INTERNAL_ERROR : String = "internalError"
INTERRUPT_ERROR : String = "interruptError"
INVALID_DATABASE_ERROR : String = "invalidDatabaseError"
IO_ERROR : String = "ioError"
LARGE_FILE_SUPPORT_ERROR : String = "largeFileSupportError"
LOCKED_ERROR : String = "lockedError"
MISMATCH_ERROR : String = "mismatchError"
MISUSE_ERROR : String = "misuseError"
OPEN : String = "connect"
OUT_OF_MEMORY_ERROR : String = "outOfMemoryError"
PERMISSION_ERROR : String = "permissionError"
PREPARE : String = "prepare"
PROTOCOL_ERROR : String = "protocolError"
READONLY_ERROR : String = "readonlyError"
RESET : String = "reset"
ROLLBACK : String = "rollback"
SCHEMA_ERROR : String = "schemaError"
TOO_BIG_ERROR : String = "tooBigError"
UNKNOWN_ERROR : String = "unknownError"

flash.events.SQLStatusEventLevels

Public Constants

ERROR : String = "error"
STATUS : String = "status"

flash.events.SQLUpdateEvent

Public Properties

kind : String
rowID : Number
table : String

Public Methods

SQLUpdateEvent(type:String,
bubbles:Boolean = false,
cancelable:Boolean = false,
kind:String = "",
table:String = "",
rowID:Number = -1)

Public Constants

DELETE : String = "delete"
INSERT : String = "insert"
UPDATE : String = "update"

* flash.html.HTMLControl

Public Properties

height : Number
historyLength : uint
historyPosition : uint
htmlHeight : Number
htmlHost : HTMLHost
htmlWidth : Number
loaded : Boolean
location : String
paintsDefaultBackground : Boolean
pdfCapability : int
scrollH : Number
scrollV : Number
useApplicationDomain : ApplicationDomain
userAgent : String
width : Number
window : JavaScriptObject

Public Methods

HTMLControl()
cancelLoad():void
historyAt(position:uint):URLRequest
historyBack():void
historyForward():void
historyGo(steps:int):void
load(urlRequestToLoad:URLRequest):void
loadString(htmlContent:String):void
reload():void

Events

complete
domInitialize
htmlBoundsChange
htmlRender
locationChange
scroll
uncaughtJavaScriptException

flash.html.HTMLHost

Public Properties

htmlControl : HTMLControl
windowRect : Rectangle

Public Methods

HTMLHost(defaultBehaviors:Boolean = true)
createWindow(crtOpt:HTMLWindowCreateOptions):HTMLControl
updateLocation(locationURL:String):void
updateStatus(status:String):void
updateTitle(title:String):void
windowBlur():void
windowFocus():void

flash.html.HTMLWindowCreateOptions

Public Properties

fullscreen : Boolean
locationBarVisible : Boolean
menuBarVisible : Boolean
resizable : Boolean
scrollBarsVisible : Boolean
statusBarVisible : Boolean
toolBarVisible : Boolean

flash.html.JavaScriptFunction

None

flash.html.JavaScriptObject

None

*

flash.filesystem.File

Public Properties

applicationResourceDirectory : File
applicationStorageDirectory : File
desktopDirectory : File
documentsDirectory : File
exists : Boolean
icon : Icon
isDirectory : Boolean
isHidden : Boolean
lineEnding : String
nativePath : String
parent : File
separator : String
systemCharset : String
url : String
userDirectory : File

Public Methods

File(path:String = null)
browseForDirectory(title:String):void
browseForOpen(title:String,
typeFilter:Array = null):void
browseForOpenMultiple(title:String,
typeFilter:Array = null):void
browseForSave(title:String):void
cancel():void
canonicalize():void
clone():File
copyTo(newLocation:FileReference,
clobber:Boolean = false):void
copyToAsync(newLocation:FileReference,
clobber:Boolean = false):void
createDirectory():void
createTempDirectory():File
createTempFile():File
deleteDirectory(deleteDirCont:Boolean = false):void
deleteDirectoryAsync(deleteDirCont:Boolean = false):void
deleteFile():void
deleteFileAsync():void
listDirectory():Array
listDirectoryAsync():void
listRootDirectories():Array
moveTo(newLocation:FileReference,
clobber:Boolean = false):void
moveToAsync(newLocation:FileReference,
clobber:Boolean = false):void
moveToTrash():void
moveToTrashAsync():void
relativize(ref:FileReference,
useDotDot:Boolean = false):String
resolve(path:String):File

Events

cancel
complete
directoryListing
ioError
securityError
select
selectMultiple

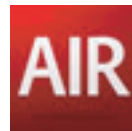
flash.filesystem.FileMode

Public Constants

APPEND : String = "append"
READ : String = "read"
UPDATE : String = "update"
WRITE : String = "write"

AIR Cheatsheet

actionscripcheatsheet.com



adobe.com/go/air

*

flash.filesystem.FileStream

Public Properties

bytesAvailable : uint
objectEncoding : uint
position : Number
endian : String
readAhead : Number

Public Methods

FileStream()
close():void
open(file:File, fileMode:String):void
openAsync(file:File,
fileMode:String):void
readBoolean():Boolean
readByte():int
readBytes(bytes:ByteArray,
offset:uint = 0,
length:uint = 0):void
readDouble():Number
readFloat():Number
readInt():int
readMultiByte(length:uint,
charSet:String):String
readObject():*
readShort():int
readUnsignedByte():uint
readUnsignedInt():uint
readUnsignedShort():uint
readUTF():String
readUTFBytes(length:uint):String
truncate():void
writeBoolean(value:Boolean):void
writeByte(value:int):void
writeBytes(bytes:ByteArray,
offset:uint = 0,
length:uint = 0):void
writeDouble(value:Number):void
writeFloat(value:Number):void
writeInt(value:int):void
writeMultiByte(value:String,
charSet:String):void
writeObject(object:*):void
writeShort(value:int):void
writeUnsignedInt(value:uint):void
writeUTF(value:String):void
writeUTFBytes(value:String):void

Events

close
complete
ioError
outputProgress
progress

*

flash.system.NativeWindowCapabilities

Public Properties

hasAlternateSystemChrome : Boolean
hasApplicationIcon : Boolean
hasWindowIcon : Boolean
windowHasMenu : Boolean
windowMaxSize : Point
windowMinSize : Point

flash.system.Shell

Public Properties

autoExit : Boolean
id : String
menu : NativeMenu
shell : Shell

Public Methods

addEventListener(type:String,
listener:Function,
useCapture:Boolean = false,
priority:int = 0,
useWeakReference:Boolean = false):void
dispatchEvent(event:Event):Boolean
exit(errorCode:int = 0):void

Events

activate
deactivate
invoke
networkChange

flash.system.Updater

Public Methods

Updater()
update(airFile:File, version:String):void

* mx.controls.FileSystemComboBox

Public Properties

COMPUTER : File
directory : File
indent : int
showIcons : Boolean

Public Methods

FileSystemComboBox()

Events

directoryChange

Styles

computerIcon
directoryIcon

mx.controls.FileSystemDataGrid

Public Properties

backHistory : Array
canNavigateBack : Boolean
canNavigateDown : Boolean
canNavigateForward : Boolean
canNavigateUp : Boolean
COMPUTER : File
creationDateColumn : DataGridColumn
dateFormatString : String
directory : File
enumerationMode : String
extensions : Array
filterFunction : Function
forwardHistory : Array
modificationDateColumn : DataGridColumn
nameColumn : DataGridColumn
nameCompareFunction : Function
showExtensions : Boolean
showHidden : Boolean
showIcons : Boolean
sizeColumn : DataGridColumn
sizeDisplayMode : String
typeColumn : DataGridColumn

Public Methods

FileSystemDataGrid()
clear():void
findItemMatching(file:File):File
navigateBack(index:int = 0):void
navigateDown():void
navigateForward(index:int = 0):void
navigateTo(directory:File):void
navigateUp():void
refresh():void

Events

directoryChange
directoryOpening
select

Styles

directoryIcon
fileIcon
refreshInterval

mx.controls.FileSystemEnumerationMode

Public Constants

DIRECTORIES_FIRST : String = "directoriesFirst"
DIRECTORIES_ONLY : String = "directoriesOnly"
FILES_AND_DIRECTORIES : String = "filesAndDirectories"
FILES_FIRST : String = "filesFirst"
FILES_ONLY : String = "filesOnly"

mx.controls.FileSystemHistoryButton

Public Properties

dataProvider : Object

Public Methods

FileSystemHistoryButton()

Events

itemClick

AIR Cheatsheet

actionsriptcheatsheet.com



adobe.com/go/air

* mx.controls.FileSystemList

Public Properties

backHistory : Array
canNavigateBack : Boolean
canNavigateDown : Boolean
canNavigateForward : Boolean
canNavigateUp : Boolean
COMPUTER : File
directory : File
enumerationMode : String
extensions : Array
filterFunction : Function
forwardHistory : Array
nameCompareFunction : Function
showExtensions : Boolean
showHidden : Boolean
showIcons : Boolean

Public Methods

FileSystemList()
clear():void
findItemMatching(file:File):File
navigateBack(index:int = 0):void
navigateDown():void
navigateForward(index:int = 0):void
navigateTo(directory:File):void
navigateUp():void
refresh():void

Events

directoryChange
directoryOpening
select

Styles

directoryIcon
fileIcon
refreshInterval

mx.controls.FileSystemSizeDisplayMode

Public Constants

BYTES : String = "bytes"
KILOBYTES : String = "kilobytes"

mx.controls.FileSystemTree

Public Properties

COMPUTER : File
directory : File
enumerationMode : String
extensions : Array
filterFunction : Function
nameCompareFunction : Function
showExtensions : Boolean
showHidden : Boolean
showIcons : Boolean

Public Methods

FileSystemTree()
clear():void
closeSelectedSubdirectory():void
findItemMatching(file:File):File
openSelectedSubdirectory():void
refresh():void

Events

directoryChange
directoryClosing
directoryOpening
select

mx.controls.HTML

Public Properties

data : Object
htmlControl : HTMLControl
htmlText : String
javascriptDocument : JavaScriptObject
javascriptWindow : JavaScriptObject
listData : BaseListData
loaded : Boolean
location : String
useApplicationDomain : ApplicationDomain

Public Methods

HTML()

Events

complete
domInitialize
htmlRender
locationChange
uncaughtJavaScriptException

Styles

htmlControlClass
paddingBottom
paddingLeft
paddingRight
paddingTop

*

mx.core.IWindow

Public Properties

status : String
title : String
titleLabel : Class
window : NativeWindow

Public Methods

close():void
maximize():void
minimize():void
restore():void

mx.core.Window

mx.core.WindowedApplication

Public Properties

activated : Boolean
bounds : Rectangle
controlBar : UIComponent
height : Number
maximizable : Boolean
maximumHeight : Number
maximumWidth : Number
minimizable : Boolean
minimumHeight : Number = 100
minimumWidth : Number = 100
resizable : Boolean
showStatusBar : Boolean
status : String
systemChrome : String
title : String
titleLabel : Class
transparent : Boolean
type : String
width : Number
window : NativeWindow
windowVisible : Boolean

Protected Properties

statusBar : Object
titleBar : Object

Public Methods

Window()
activate():void
close():void
maximize():void
minimize():void
orderInBackOf(win:NativeWindow):void
orderInFrontOf(win:NativeWindow):void
orderToBack():void
orderToFront():void
restore():void

Protected Methods

mouseDownHandler(event:MouseEvent):void

Events

applicationActivate
applicationDeactivate
closing
displayStateChange
displayStateChanging
initialLayoutComplete
move
moving
networkChange

Styles

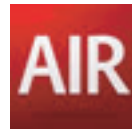
buttonAlignment
buttonPadding
closeButtonMacStyleName
closeButtonStyleName
headerHeight
maximizeButtonMacStyleName
maximizeButtonStyleName
minimizeButtonMacStyleName
minimizeButtonStyleName
restoreButtonStyleName
showFlexChrome
statusBackgroundSkin
statusBarBackgroundColor
statusBarClass
statusTextStyleName
titleLabelAlignment
titleLabelBackgroundSkin
titleLabelButtonPadding
titleLabelClass
titleLabelColors
titleLabelTextStyleName

Effects

closeEffect
minimizeEffect
unminimizeEffect

AIR Cheatsheet

actionscripcheatsheet.com



adobe.com/go/air

*

mx.core.windowClasses.StatusBar

Public Properties

data : Object
status : String

Public Methods

StatusBar()

Protected Methods

commitProperties():void
createChildren():void
measure():void
updateDisplayList(unscaledWidth:Number,
unscaledHeight:Number):void

mx.core.windowClasses.TitleBar

Public Properties

closeButton : Button
data : Object
maximizeButton : Button
minimizeButton : Button
title : String
titleLabel : Class

Protected Properties

titleLabel : UITextField

Public Methods

TitleBar()
styleChanged(styleProp:String):void

Protected Methods

commitProperties():void
doubleClickHandler(event:MouseEvent):void
placeButtons(alignment:String,
unscaledWidth:Number,
unscaledHeight:Number,
leftOffset:Number,
rightOffset:Number,
cornerOffset:Number):void
placeTitle(titleAlign:String,
leftOffset:Number,
rightOffset:Number,
buttonAlign:String):void

About The Author: seantheflashguy@gmail.com



Sean Moore is currently located in sunny San Diego CA working as a freelance ActionScript Programmer and Flash Developer. Please get in touch for his rates and availability.

Sean is a highly skilled ActionScript Developer with extensive experience developing for the Flash Platform.

Sean is currently focusing on Flex 2, Flex 3, Adobe AIR, ActionScript 3.0, Papervision 3D, OOP and Design Patterns. Applications Sean has helped create are being used today by several prominent Government Agencies including the USAF and NARA.

Some of Sean's specialties are: ActionScript 3.0 Programming, Application Architecture for the Flash Platform, Flash and ActionScript consultation, AIR development, Flex Development and Papervision 3D Programming. Sean also has extensive experience with ActionScript 1.0, ActionScript 2.0, Flash MX and Flash MX 2004.

<http://www.seantheflashguy.com/blog>