

Flex Class Library Reference

mx.core

- C Application**
extends mx.containers.Container
 Flex defines a default, or Application, container that lets you start adding content to your application without explicitly defining another container.
mx `<mx:Application/>`
- C DragSource**
extends mx.events.DragEvent
 The DragSource class contains drag data and specifies the formats for the data.
- C ExternalContent**
extends mx.events.ExternalContentEvent
 The ExternalContent class lets you determine how many times to try loading external content before determining that the load has failed.
- C MXMLObject**
interface MXMLObject
 The MXMLObject interface indicates that the implementing object can handle additional methods generated by the compiler.
- C MXMLObjectAdapter**
extends mx.events.ExternalContentEvent
 The MXMLObjectAdapter class is the default implementation of the MXMLObject interface, so that you can implement the interface without defining all of the methods.
- C Repeater**
extends mx.events.RepeaterEvent
 The Repeater class is the runtime object that corresponds to the `<mx:Repeater>` tag.
mx `<mx:Repeater/>`
- C ScrollView**
extends mx.core.View
 The ScrollView class is the base class for components with scroll bars and a mask.
- C UIComponent**
extends mx.core.UIObject
 The UIComponent class is the base class for all Flex components other than the Label and ProgressBar controls.
- C UIObject**
extends MouseEvent
 The UIObject class is the base class for all components and graphical objects.
- C UIObjectDescriptor**
extends mx.events.UIObjectEvent
 The UIObjectDescriptor contains the MXML information used to create a UIObject in a Flex application.
- C View**
extends mx.core.UIComponent
 The View class is an abstract base class.

mx.core.ext

- C FlexExtensions**
extends mx.events.Event
 The FlexExtensions class is for internal use only.
- C UIComponentExtensions**
extends mx.events.Event
 The FlexExtensions class is for internal use only.

mx.effects

- C CompositeEffect**
extends mx.effects.Effect
 The CompositeEffect class is the superclass for `<mx:Parallel>` and `<mx:Sequence>` effects.
- C Effect**
extends Object
 The Effect class is the superclass for all effects.
- C EffectManager**
extends Object
 The EffectManager class listens to events, such as the show and move events, that are fired by objects in the user's document.
- C Fade**
extends mx.effects.TweenEffect
 The Fade effect animates the component, either from transparent to opaque, or from opaque to transparent.
mx `<mx:Fade/>`
- C MaskEffect**
extends mx.effects.TweenEffect
 The MaskEffect class is an abstract base class for all effects that animate a mask, such as a wipe effect.
- C Move**
extends mx.effects.TweenEffect
 The Move effect changes the position of a component over a specified time interval.
mx `<mx:Move/>`
- C Parallel**
extends mx.effects.CompositeEffect
 The Parallel class plays combined effects at the same time.
mx `<mx:Parallel/>`
- C Pause**
extends mx.effects.Effect
 The Pause effect is useful when compositing effects.
mx `<mx:Pause/>`
- C Resize**
extends mx.effects.TweenEffect
 The Resize effect changes the width, height, or both dimensions of a component over a specified time interval.
mx `<mx:Resize/>`
- C Sequence**
extends mx.effects.Effect
 The Sequence class plays combined effects one after the other.
mx `<mx:Sequence/>`
- C Tween**
extends Object
 The Tween effect fills in the missing sequences in an effect, between a starting position or state and the ending position or state, to create a full animation.
- C TweenEffect**
extends mx.effects.Effect
 The TweenEffect class is the superclass for all effects that are based on the Tween object.
- C WipeDown**
extends mx.effects.MaskEffect
 The WipeDown class defines a bar wipe down effect.
mx `<mx:WipeDown/>`
- C WipeLeft**
extends mx.effects.MaskEffect
 The WipeLeft class defines a bar wipe left effect.
mx `<mx:WipeLeft/>`
- C WipeRight**
extends mx.effects.MaskEffect
 The WipeRight class defines a bar wipe right effect.
mx `<mx:WipeRight/>`
- C WipeUp**
extends mx.effects.MaskEffect
 The WipeUp class defines a bar wipe up effect.
mx `<mx:WipeUp/>`
- C Zoom**
extends mx.effects.MaskEffect
 The Zoom effect zooms the object in or out on a center point.
mx `<mx:Zoom/>`

mx.utils

- C ArrayUtil**
extends Object
 The ArrayUtil class is a static class with methods for working with XML within Flex.
- C ClassUtil**
extends Object
 The ClassUtil class is used internally with the tag to translate server-side Java objects to ActionScript.
- C Delegate**
extends Object
 The Delegate class creates a function wrapper to let you run a function in the context of the original object, rather than in the context of the second object, when you pass a function from one object to another.
- C ObjectCopy**
extends Object
 Static class with methods for performing copies of in memory objects including nested properties arrays etc.
- C URLUtil**
extends Object
 The URLUtil class is a static class with methods for working with full and relative URLs within Flex.
- C XMLUtil**
extends Object
 The XMLUtil class is a static class with methods for working with XML within Flex.

mx.formatters

- C CurrencyFormatter**
extends mx.formatters.Formatter
 The CurrencyFormatter class formats a valid number.
mx `<mx:CurrencyFormatter/>`
- C DataBase**
extends mx.events.Event
 The DataBase class contains the localized string information used by the DateFormatter class and the parsing function that renders the pattern.
- C DateFormatter**
extends mx.formatters.Formatter
 The DateFormatter uses a valid user defined pattern string to render date and time strings from a Date object.
mx `<mx:DateFormatter/>`
- C Formatter**
extends Object
 The Formatter class is the superclass for the formatter package.
- C NumberBase**
extends Object
 The NumberBase class is a utility that contains general number formatting capabilities, including rounding, precision, thousands formatting, and negative sign formatting.
- C NumberFormatter**
extends mx.formatters.Formatter
 The NumberFormatter class formats a valid number by adjusting the decimal rounding and precision, the thousands separator, and the negative sign.
mx `<mx:NumberFormatter/>`
- C PhoneFormatter**
extends mx.formatters.Formatter
 The PhoneFormatter formats a valid number into a phone format, including international configurations.
mx `<mx:PhoneFormatter/>`
- C StringFormatter**
extends mx.formatters.Formatter
 The StringFormatter class is taken from the Flash MX 2004 Data package.
- C SwitchSymbolFormatter**
extends mx.formatters.Formatter
 The SwitchSymbolFormatter is a simple utility that creates a formatted string by parsing through a pattern and either rendering the pattern character or rendering a number from a second source, if the pattern character happens to be a number placeholder.
- C ZipCodeFormatter**
extends mx.formatters.Formatter
 The ZipCodeFormatter class formats a valid number into one of the following formats, based on a user-supplied formatString.
mx `<mx:ZipCodeFormatter/>`

mx.styles

- C CSSSetStyle**
extends Object
 You must use this package to make style changes of CSS properties at runtime.
- C CSSStyleDeclaration**
extends Object
 The CSSStyleDeclaration class represents a set of CSS style rules.
- C CSSTextStyles**
extends Object
 The CSSTextStyles class is the factory for adding CSS Text styles to the StyleDeclarations and UIObject classes.
- C StyleManager**
extends Object
 The StyleManager class manages the following:
 - Which CSS style properties the class inherits
 - Which style properties are colors, and therefore get special handling
 - A list of strings that are aliases for color values

mx.validators

- C CreditCardValidator**
extends mx.validators.Validator
 The CreditCardValidator class validates that a credit card number is the correct length for the specified card type.
mx `<mx:CreditCardValidator/>`
- C DateValidator**
extends mx.validators.Validator
 Validates that a string or object is a proper date and matches a specified format.
mx `<mx:DateValidator/>`
- C EmailValidator**
extends mx.validators.Validator
 The EmailValidator class validates that a string has an at sign and a period in the domain.
mx `<mx:EmailValidator/>`
- C NumberValidator**
extends mx.validators.Validator
 The NumberValidator class validates that a string is a valid number between the minimum and maximum values.
mx `<mx:NumberValidator/>`
- C PhoneNumberValidator**
extends mx.validators.Validator
 The PhoneNumberValidator class validates that a string is a valid phone number.
mx `<mx:PhoneNumberValidator/>`
- C SocialSecurityValidator**
extends mx.validators.Validator
 The SocialSecurityValidator class validates that a string is a valid United States Social Security number.
mx `<mx:SocialSecurityValidator/>`
- C StringValidator**
extends mx.validators.Validator
 StringValidator validates that a string length is within a specified range.
mx `<mx:StringValidator/>`
- C ValidationFailedEvent**
extends mx.validators.Validator
 ValidationFailedEvent is dispatched to the listener if an error is encountered during the validation process.
- C Validator**
extends Object
 The Validator class is the base class for all other validators.
- C ZipCodeValidator**
extends mx.validators.Validator
 The ZipCodeValidator class validates that a string has the correct length for a United States ZIP code, ZIP-4 code, or Canadian postal code.
mx `<mx:ZipCodeValidator/>`

mx.servicetags

- C HTTPService**
extends mx.events.HTTPServiceEvent
 All implementations of this tag must implement MXMLObject. You use the `<mx:HTTPService>` tag to represent an HTTPService object in an MXML file.
mx `<mx:HTTPService/>`
- C RemoteObject**
extends mx.servicetags.Service
 The `<mx:RemoteObject>` tag gives you access to the methods of Java objects without manually configuring the objects as SOAP-based web services.
mx `<mx:RemoteObject/>`
- C Service**
extends mx.events.HTTPServiceEvent
 The `<mx:WebService>` tag is used to represent a web service in an MXML file.
mx `<mx:WebService/>`

mx.managers

- C CursorManager**
extends mx.events.Event
 The CursorManager class controls a prioritized list of cursors, where the cursor with the highest priority is currently visible.
- C DepthManager**
extends mx.events.Event
 The DepthManager class manages the depth (z-order) of objects.
- C DragManager**
extends mx.events.Event
 The DragManager class manages drag-and-drop operations, which let you move data from one place to another in a Flex application.
- C FocusManager**
extends mx.core.UIComponent
 The FocusManager class manages the focus on components.
- C HistoryManager**
extends mx.events.Event
 History management lets users navigate through a Flex application using the web browser's Back and Forward navigation commands.
- C LayoutManager**
extends mx.events.Event
 The LayoutManager class performs a full layout during initialization, and then performs an incremental layout whenever the size of any object changes.
- C OverlappedWindows**
extends mx.events.Event
 The OverlappedWindows class manages the display of overlapped windows, including determining which window has focus and responding to mouse clicks that change which window should display on top.
- C PopUpManager**
extends mx.events.Event
 The PopUpManager class creates new top-level windows.
- C SystemManager**
extends mx.events.Event
 The SystemManager class manages which top-level window is activated.
- C ToolTipManager**
extends mx.events.Event
 The ToolTipManager lets you set basic ToolTip functionality, such as display delay and the disabling of ToolTips.

mx.managers.layoutClasses

- C PriorityQueue**
extends mx.events.Event
 The PriorityQueue class provides a general purpose priority queue.

mx.preloaders

- C DownloadProgressBar**
extends mx.events.Event
 The DownloadProgressBar class displays download progress.

mx.events

- C EventDispatcher**
extends Object
 The EventDispatcher class is the base class for event listening and dispatching.
- C LowLevelEvents**
extends mx.events.Event
 The LowLevelEvents class provides support for low-level DOM mouse events.
- C UIEventDispatcher**
extends mx.events.EventDispatcher
 The UIEventDispatcher class performs event listening and dispatching for UIObjects.