



*User Manual*  
v1.1

# Contents

<b>Introduction</b>	<b>3</b>
Some features	3
How does it work?	3
<b>So what's an .XML-file?</b>	<b>3</b>
<b>Let's set things up!</b>	<b>5</b>
Prepare folder structure	5
Create .XML-file	5
Put MonoSlideshow on your site!	5
<b>Extra's</b>	<b>6</b>
Disable startup logo	6
Load a custom .XML-file	6
Start with a specified image	6
Loading Flickr RSS Feeds	7
<b>Attributes</b>	<b>8</b>
<b>Global attributes</b>	<b>8</b>
<b>Attributes for images</b>	<b>8</b>
Main attributes	8
The image Info object	9
<b>Attributes for Albums</b>	<b>10</b>
Main attributes	10
The thumbnail window object	11
The album window object	12
The controls object	13
The loading icon object	14
The mark object	14

# Introduction

MonoSlideshow is a Flash slideshow to view your images in a web page. It can be placed on every website, in whatever size you want and filled with whatever content you like. It's designed for maximum flexibility, whether you want to have a professional looking portfolio, or just an image rotator on your website.

## Some features

- Resize and position everything to suit your needs
- Color every object you see
- Attach links to images
- Supports sound and music
- Caching and preloading images

## How does it work?

Monoslideshow is a single .SWF-file. There's no need to edit the slideshow in the Flash IDE itself. Instead, all its information is directly read from a .XML-file. It's compact, extremely versatile and easy to set up. The fact that Monoslideshow uses XML to populate the albums and setting the preferences, means that it is flexible and easily integrated in your existing site.

*Tip: use the [demo page](http://www.monoslideshow.com) on [www.monoslideshow.com](http://www.monoslideshow.com) to automatically create .XML-files!*

## So what's an .XML-file?

MonoSlideshow reads all its information from a single textfile called `monoslideshow.xml`. To put images in your slideshow, you have to edit this file. This could easily be done by using a simple text-editor such as Notepad or TextEdit.

What exactly is an .XML-file? Well, basically it's just a plain text file. But .XML-files differ from ordinary files in that they use certain rules to structure the content. XML is a computer language, and actually it's a bit like HTML. Just as with .HTML-files, content in .XML-files is placed inside *tags*. Let's take a look at an example to illustrate how all this works.

```
<?xml version="1.0" encoding="utf-8"?>
<slideshow>
  <album thumbnail= "album1.jpg" imagePath="album1">
    
    
    
  </album>
</slideshow>
```

The first line says that this textfile is a file which uses the XML-language. You always have to include this line. The second line is an *opening tag* called `<slideshow>`. Its corresponding *closing tag* is located at the last line. Everything between these two tags is the content of it. An opening tag must always have the exact same name as the closing tag. The only way a closing tag differs from its opening tag, is an extra forward slash.

At the third line starts another opening tag. The tag is called `album` and its corresponding closing tag is located four lines further. Inside the opening tag are two *attributes*, called `thumbnail` and `imagePath`. *Attributes* are always contained inside a tag. The value of an attribute (in this case `album1.jpg` and `album1`, respectively) must always be enclosed

between quotes (" or '). They are to be used to specify something about the tag they are contained in. In this case, the thumbnail of the album is album1.jpg and the path to all its images is images.

The three next <img> tags are the actual image files. They all contain two attributes, called src and thumbnail. The attribute src specifies the link to the image inside the folder imagePath. The link to the thumbnail is specified by the attribute thumbnail. As you've perhaps noticed, these tags lack a regular closing tag. Instead of writing </img>, you could also write . These type of tags are called *self closing tags*. We use them here because the <img> tag has no content by itself. All the information is extracted out of its attributes.

You can completely build your own slideshow by writing your own XML-file. You only have to know which attribute names MonoSlideshow uses to customize the slideshow. They are all summed up a few pages further. Let's first take a look at another example:

```
<?xml version="1.0" encoding="utf-8"?>
<slideshow>
  <preferences
    imageTransition="bubblesBlend"
    controlAutoHide="false"
  />
  <album thumbnail="album1.jpg" imagePath="album1">
    
    
    
  </album>
  <album thumbnail="album2.jpg" imagePath="album2" thumbnailPath="thumbs">
    
    
    
    
    
    
  </album>
</slideshow>
```

This slideshow consists of two albums. Also, a new attribute called thumbnailPath is introduced in the second album tag. This attribute defines the path to all the images. Now, if an <img> tag doesn't contain a thumbnail attribute, it looks for a thumbnail in this folder for a file with the **exact same name** as in the src attribute. This way, you could easily set up two folders: One for all the images, and one for all the thumbnails.

Another new tag is introduced here: the <preferences> tag. Here you would insert the attributes which you'd like to change for the entire slideshow. The tag is self-closing and consists only of attributes. For example, imageTransition is an attribute for the <image> tag, but could also be inserted in the <album> tag (to affect all images in it), or even in the <preference> tag (to affect all images in all albums). Attributes inserted at a lower level override those at a higher level.

## Let's set things up!

1. Prepare folder structure.
2. Create .XML-file.
3. Put MonoSlideshow on your site.

### Prepare folder structure

The following example shows a basic structure for your folders:

```
/index.html  
  
/slideshow/  
/slideshow/monoslideshow.xml  
/slideshow/monoslideshow.swf  
  
/slideshow/images/  
/slideshow/images/photo1.jpg  
/slideshow/images/photo2.jpg  
/slideshow/images/photo3.jpg  
  
/slideshow/thumbnails/  
/slideshow/thumbnails/photo1.jpg  
/slideshow/thumbnails/photo2.jpg  
  
/slideshow/thumbnails/photo3.jpg
```

### Create .XML-file

Put all references to all images in your .XML-file. Use the examples above to see how the .XML-file is structured. If you're stuck, you could get help in the forums on [www.monoslideshow.com](http://www.monoslideshow.com).

*Note: Be sure to correctly implement your links. For example, if Monoslideshow can't find your mark file or your sound files, Monoslideshow will not run.*

### Put MonoSlideshow on your site!

To put monoslideshow.swf on your site, you have to embed it in your webpages. Nowadays, Flash content is mostly placed in websites via Javascript. A good technique for this is using Deconcept's SWFObject. You can find instructions of how to use it here: <http://blog.deconcept.com/swfobject/>

## Extra's

Monoslideshow has some default behaviours. These behaviours can't be altered via the .XML-file. Instead, these can be customized by passing Flash parameters to Monoslideshow. Below are the instructions.

### Disable startup logo

By default, Monoslideshow displays a logo when it's loading the slideshow. To disable this, you have to do set a Flash parameter `showLogo` to `false`. Using SWFObject, you can do the following:

In the head of your page, place a link to SWFObject:

```
<script src="swfobject.js" type="text/javascript" ></script>
```

On your web page (i.e., `index.html`), place your slideshow in a `<div>` between `<body>` and `</body>` like this:

```
<div id = "mssHolder">
  <p>Temporary holder for the Flash object</p>
</div>
```

Place this piece of javascript on the same page, below the `<div>`:

```
script type="text/javascript">
var so = new SWFObject("monoslideshow.swf", "mss", 480, 360, "7", "#ffffff");
so.addVariable("showLogo", "false");
so.write("mssHolder");
</script>
```

### Load a custom .XML-file

By default, Monoslideshow loads a file called `monoslideshow.xml`. To load a custom file, you have to pass the parameter `dataFile` to Flash. It works like this:

```
script type="text/javascript">
var so = new SWFObject("monoslideshow.swf", "mss", 480, 360, "7", "#ffffff");
so.addVariable("dataFile", "yourfile.xml");
so.write("mssHolder");
</script>
```

### Start with a specified image

To start with a specified image, you can place the `startWithImageID` attribute in the `<preferences>` tag. You could also pass the parameter to Flash. It works like this:

```
script type="text/javascript">
var so = new SWFObject("monoslideshow.swf", "mss", 480, 360, "7", "#ffffff");
so.addVariable("startWithImageID", "idName");
so.write("mssHolder");
</script>
```

To start with the second photo, your .XML file would be looking something like this:

```
<album thumbnail="album1.jpg" imagePath="album1" thumbnailPath="thumbs1">
  
```

```


</album>
```

## Loading Flickr RSS Feeds

Monosideshow supports Flickr RSS feeds. You can find links to these .RSS feeds on group pages and personal pages on [www.flickr.com](http://www.flickr.com). To load a Flickr album, just copy the URL of the .RSS file and paste it in your .XML file like so:

```
<album size="medium">
  <flickr><![CDATA[url_to_rss_feed]]></flickr>
</album>
```

There are a few attributes you can set in the <album> tag:

**linkToImage** *true, false [false]* If true, all images will be clickable, leading the user to a new window where the original image is displayed.

**size** *"tiny" "thumbnail" "small" "medium" "large" "original"* Force the size of the images of a Flickr .RSS feed. If no size is specified, the images are displayed according to how Flickr serves this image in the .RSS feed.

## Attributes

To customize MonoSlideshow, you have to edit the .XML file. There are a lot of options available. The following objects each have their own attributes: *Global*, *images*, *image window*, *thumbnail window*, *loading icon*, *controls*. All available attributes are defined below. The standard values are displayed between square brackets.

## Global attributes

The following attributes are global attributes. They are to be defined only once in the <preferences> tag.

backgroundColor **color** [ffffff] *The background color of the slideshow.*  
randomizeAlbums **true, false** [false] *If true, the order of the albums is shuffled.*  
loadAlbum **text** [none] *Type a title of an album here. This album will be displayed when Monoslideshow loads. Note: will soon be deprecated, please use startWithAlbumID.*  
startWith **“photos”, “thumbnails”, “albums”** [photos] *Determines what section to start when Monoslideshow loads.*  
startWithAlbumID **text** [none] *Specify an ID of an album to start your slideshow with.*  
startWithImageID **text** [none] *Specify an ID of an image to start your slideshow with.*

## Attributes for images

Every image tag <img> can have the attributes specified below. Every attribute below could also be placed in the <preferences> or the <album> tag to specify global values for each of the images the tag contains (with the exception of title, description, src, thumbnail, link and target).

### Main attributes

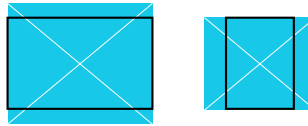
title **text** *Title of the image.*  
description **text** *Description of the image.*  
src **url** *Source of the image.*  
thumbnail **url** *Link to a thumbnail of the image.*  
link **url** *When the image is clicked with the mouse, go to this link.*  
target **text** [\_blank] *Targetwindow in which the link is displayed.*  
id **text** *Specify an ID name to use with startWithImageID.*  
sound **string** *Soundfile to be played when image is displayed.*  
soundVolume **number** 0 - 100 [100] *Volume of the sound file.*  
waitForSound **true, false** [true] *If true, the image will wait to fade out until the sound file is completely played.*  
imageAlign **“topLeft”, “topRight”, “bottomLeft”, “bottomRight”, “leftCenter”, “rightCenter”, “bottomCenter”, “topCenter”, “center”** [center] *How to position the image.*  
imageMarginX **number** 0 - 1000 [0] *Horizontal margin from the left or right side in pixels.*  
imageMarginY **number** 0 - 1000 [0] *Vertical margin from the top or bottom side in pixels.*  
imageScaleMode **“scaleToFit”, “scaleToFill”, “noScale”, “downscaleToFit”, “downscaleToFill”** [scaleToFill] *How the image will be scaled. noScale doesn't scale the image. downscaleToFit and downscaleToFill will only be applied if the resulting scaled image will be smaller than the original. Some examples of how scaleToFit and scaleToFill work:*



scaleToFit:



scaleToFill:



imagePause number 0.5, 86400 [3] Time to wait in seconds until the next image is displayed.

imageTransitionTime number 0.1 - 1000 [0.5] Time of transition in seconds.

imageTransition: "blend", "leftToRight", "rightToLeft", "topToBottom", "bottomToTop", "leftToRightBlend", "rightToLeftBlend", "topToBottomBlend", "bottomToTopBlend", "leftToRightFadeOutBackwards", "rightToLeftFadeOutBackwards", "topToBottomFadeOutBackwards", "bottomToTopFadeOutBackwards", "pinhole", "pinholeBlend", "fadeInOut", "bubbles", "bubblesBlend", "photoFlash", "noTransition" [blend] Choose a transition type. Some examples:

blend:



leftToRightBlend:



fadeInOut:



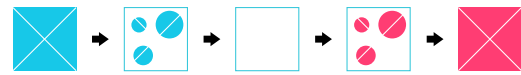
bubblesBlend:



leftToRight:



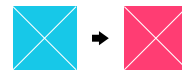
bubbles:



leftToRightFadeOutBackwards:



noTransition:



kenBurnsMode "random", "randomPan", "randomZoomIn" "randomZoomOut", "none" [none] The Ken Burns effect slowly zooms or pans across the image. Note: if active, imageScaleMode and imageAlign will have no effect. The Ken Burns effect scales the images according to scaleToFit and imageAlign will be center. This way, there's always room to smoothly pan and zoom across the image in the slideshow. There will be no left-over space.

kenBurnsVariationPercent number 0 - 100 [25] Specify how much variation will be applied to the Ken Burns effect.

kenBurnsTime number 0.1 - 1000 [5] The time in seconds the Ken Burns effect is active.

kenBurnsStart coordinates x1, y1, x2, y2 [-] Coordinates of the rectangle on the image to start the Ken Burns effect with. Use coordinates from the unscaled image.

kenBurnsEnd coordinates x1, y1, x2, y2 [-] Coordinates of the rectangle on the image the Ken Burns effect will zoom and pan to. Use coordinates from the unscaled image.

## The image Info object

imageInfoColor color [000000] Color of the background.

imageInfoAlpha number 0 - 100 [75] Transparency value of the background.

imageInfoRoundedCorners number 0 - 100 [10] Value specifying how much the corners are rounded.

imageInfoShadowSize number 0 - 10 [0] The size of the shadow.

**imageInfoShadowColor** `color [000000]` Color of the shadow.  
**imageInfoShadowAlpha** `number 0 - 100 [100]` Transparency value of the shadow.  
**imageInfoLineWidth** `number 0 - 20 [0]` Line width (stroke) of the background.  
**imageInfoLineColor** `color [ffffff]` Color of the line.  
**imageInfoAlign**: “topLeft”, “topRight”, “bottomLeft”, “bottomRight”, “leftCenter”, “rightCenter”, “bottomCenter”, “topCenter”, “center” `[topLeft]`  
*How to position the image info in the slideshow.*  
**imageInfoMarginX** `number 0 - 1000 [5]` Horizontal margin from the left or right side in pixels.  
**imageInfoMarginY** `number 0 - 1000 [5]` Vertical margin from the top or bottom side in pixels.  
**imageInfoTransitionTime** `number 0 - 1000 [0.5]` Fade in time in seconds.  
**imageInfoWidth** `number 50 - 1000 [200]` Width in pixels.  
**imageInfoMaxSize** `true, false [true]` If true, imageInfoWidth will be overridden with the maximum size available.  
**imageInfoPadding** `number 0 - 100 [5]` Padding in pixels.  
**imageInfoTextAlign** “left”, “right”, “center” `[left]` Alignment of text.  
**imageInfoTitleSize** `number 1 - 1000 [12]` Font size of the title.  
**imageInfoTitleColor** `color [ffffff]` Color of the title.  
**imageInfoTitleContainsNumber** `true, false [true]` If true, the image info will contain the index of the current image and the length of the current album.  
**imageInfoTitleMaxCharacters** `number -1 - 1000 [-1]` Limit the number of characters displayed in the title. If the value is -1, there’s no limit.  
**imageInfoDisplayNoTitle** `text [No title]` Display “No title” if the image doesn’t have a title.  
**imageInfoDescriptionMargin** `number -1000 - 1000 [0]` Margin between title and description.  
**imageInfoDescriptionSize** `number 1 - 1000 [10]` Font size of the description.  
**imageInfoDescriptionColor** `color [ffffff]` Color of the description.  
**imageInfoDescriptionMaxCharacters** `number -1 - 1000 [-1]` Limit the number of characters displayed in the description. If the value is -1, there’s no limit.  
**imageInfoDisplayNoDescription** `text [No description]` Display “No description” if the image doesn’t have a description.

## Attributes for Albums

Every album tag <album> can have the attributes specified below. Every attribute could also be placed in the <preferences> tag to specify global values for each of the albums it contains (with the exception of title, description, thumbnail, imagePath, thumbnailPath, soundPath, backgroundMusic, backgroundMusicVolume and size).

### Main attributes

**font** “helvetica”, “unibody” `[ffffff]` The font used in this album.  
**title** `text` Title text.  
**description** `text` Description text.  
**thumbnail** `url` link to the thumbnail.  
**imagePath** `url` Path to the folder containing the images.  
**thumbnailPath** `url` Path to the folder containing the thumbnails.  
**id** `text` Specify an ID name to use with startWithAlbumID.  
**soundPath** `text` Path to the folder containing the sound and music files.  
**backgroundMusic** `url` A link to an mp3 file for the background music.  
**backgroundMusicVolume** `number 0 - 100 [100]` Volume of the background music.  
**randomizeImages** `true, false [false]` If true, images inside this album are shuffled.  
**autoPlay** `true, false [true]` If true, the slideshow is automatically started.

**autoPause** `true, false [true]` If true, the slideshow pauses when a new image is selected by the control buttons or the thumbnail window.

**onFinished** `"loop", "loadNextAlbum", "stop" [loop]` Determines what to do after an album has completed. "loop" tells the slideshow to play the album all over again. "loadNextAlbum" loads the next album and "stop" just lets the slideshow stop.

**preloadImages** `number 0 - 1000 [0]` Set the number of images to preload.

**showLoadingIcon** `true, false [true]` If true, show the loading icon.

**showAlbumsButton** `true, false [true]` If true, show the albums button.

**showThumbnailsButton** `true, false [true]` If true, show the thumbnails button.

**showMuteButton** `true, false [false]` If true, show the mute/unmute button.

**showPreviousButton** `true, false [true]` If true, show the previous button.

**showPauseButton** `true, false [true]` If true, show the play/pause button.

**showNextButton** `true, false [true]` If true, show the next button.

**showControls** `true, false [true]` If true, show the controls.

**showImageInfo** `"always", "never", "ifAvailable", "onRollOver", "onRollOverIfAvailable" [ifAvailable]` Determines if the image info will appear. "ifAvailable" only shows the info when info is available, "onRollOver" only shows the info when you roll your mouse over the image. "onRollOverIfAvailable" only shows info if it's available and you roll your mouse over the image.

## The thumbnail window object

**thumbnailBackgroundColor** `color [000000]` Color of the background overlay behind the thumbnail window.

**thumbnailBackgroundAlpha** `number 0 - 100 [75]` Transparency value of the background overlay behind the thumbnail window.

**thumbnailWindowColor** `color [000000]` Color.

**thumbnailWindowAlpha** `number 0 - 100 [60]` Transparency value.

**thumbnailWindowRoundedCorners** `number 0 - 100 [15]` Value specifying how much the corners are rounded.

**thumbnailWindowShadowSize** `number 0 - 10 [5]` Shadow size.

**thumbnailWindowShadowColor** `color [000000]` Shadow color.

**thumbnailWindowShadowAlpha** `number 0 - 100 [50]` Transparency value.

**thumbnailWindowAlign**: `"topLeft", "topRight", "bottomLeft", "bottomRight", "leftCenter", "rightCenter", "bottomCenter", "topCenter", "center" [center]`  
 How to position the image info in the slideshow.

**thumbnailWindowMarginX**: `number 0 - 1000 [0]` Horizontal margin from the left or right side in pixels.

**thumbnailWindowMarginY**: `number 0 - 1000 [0]` Vertical margin from the top or bottom side in pixels.

**thumbnailWindowRows** `number 1 - 1000 [4]` Number of rows.

**thumbnailWindowColumns** `number 1 - 1000 [4]` Number of columns.

**thumbnailWindowAutoSize** `true, false [true]` If true, the rows and columns specified will be overridden with values which are automatically calculated.

**thumbnailWindowLineWidth** `number 0 - 20 [2]` Line width (stroke) of the window.

**thumbnailWindowLineColor** `color [ffffff]` Color of the line.

**thumbnailWindowPadding** `number 0 - 100 [15]` Padding in pixels.

**thumbnailWindowIconMargin** `number 0 - 1000 [15]` Margin between icons and thumbnails.

**thumbnailWindowIconSize** `number 5 - 100 [12]` Size of the icons in pixels.

**thumbnailWindowIconSpacing** `number 5 - 100 [10]` Spacing between icons in pixels.

**thumbnailWindowIconColor** `color [ffffff]` Color of the icons.

**thumbnailWindowIconDimColor** `color [444444]` Color of the icons when they're unclickable.

**thumbnailWindowIconRollOverColor** `color [10a4f0]` Color of the icons when the mouse rolls over them.

thumbnailWindowInfoSize **number** 1 - 1000 [12] *Size of the info text.*  
 thumbnailWindowInfoColor **color** [ffffff] *Color of the info text.*  
 thumbnailWidth **number** 5 - 1000 [50] *Width of the individual thumbnails in pixels.*  
 thumbnailHeight **number** 5 - 1000 [50] *Height of the individual thumbnails in pixels.*  
 thumbnailRoundedCorners **number** 0 - 100 [3] *Value specifying how much the corners are rounded.*  
 thumbnailSpacing **number** 0 - 1000 [2] *Spacing between thumbnails in pixels.*  
 thumbnailShadowSize **number** 0 - 10 [0] *Shadow size.*  
 thumbnailShadowColor **color** [000000] *Shadow color.*  
 thumbnailShadowAlpha **number** 0 - 100 [50] *Transparency value.*  
 thumbnailHoverDistance **number** 0 - 100 [8] *Distance to hover the thumbnails when the mouse rolls over them.*  
 thumbnailHoverShadowSize **number** 0 - 10 [8] *Shadow size to morph to when the mouse rolls over them.*  
 thumbnailBrightnessAdjustment **number** -255 - 255 [0] *Brightness adjustment of the thumbnail*  
 thumbnailHoverBrightnessAdjustment **number** -255 - 255 [128] *Brightness adjustment of the thumbnail when the mouse hovers over them*

### **The album window object**

albumBackgroundColor **color** [000000] *Color of the background overlay behind the thumbnail window.*  
 albumBackgroundAlpha **number** 0 - 100 [75] *Transparency value of the background overlay behind the thumbnail window.*  
 albumWindowColor **color** [000000] *Color.*  
 albumWindowAlpha **number** 0 - 100 [60] *Transparency value.*  
 albumWindowRoundedCorners **number** 0 - 100 [15] *Value specifying how much the corners are rounded.*  
 albumWindowShadowSize **number** 0 - 10 [5] *Shadow size.*  
 albumWindowShadowColor **color** [000000] *Shadow color.*  
 albumWindowShadowAlpha **number** 0 - 100 [50] *Transparency value.*  
 albumWindowAlign: "topLeft", "topRight", "bottomLeft", "bottomRight", "leftCenter", "rightCenter", "bottomCenter", "topCenter", "center" ["center"]  
*How to position the image info in the slideshow.*  
 albumWindowMarginX: **number** 0 - 1000 [0] *Horizontal margin from the left or right side in pixels.*  
 albumWindowMarginY: **number** 0 - 1000 [0] *Vertical margin from the top or bottom side in pixels.*  
  
 albumWindowRows **number** 1 - 1000 [2] *Number of rows.*  
 albumWindowColumns **number** 1 - 1000 [2] *Number of columns.*  
 albumWindowAutoSize **true, false** [true] *If true, the rows and columns specified will be overridden with values which are automatically calculated.*  
 albumWindowLineWidth **number** 0 - 20 [2] *Line width (stroke) of the window.*  
 albumWindowLineColor **color** [ffffff] *Color of the line.*  
 albumWindowPadding **number** 0 - 100 [15] *Padding in pixels.*  
 albumWindowIconMargin **number** 0 - 1000 [15] *Margin between icons and albums.*  
 albumWindowIconSpacing **number** 5 - 100 [10] *Spacing between icons in pixels.*  
 albumWindowIconSize **number** 5 - 100 [12] *Size of the icons in pixels.*  
 albumWindowIconColor **color** [ffffff] *Color of the icons.*  
 albumWindowIconDimColor **color** [444444] *Color of the icons when they're unclickable.*  
 albumWindowIconRollOverColor **color** [10a4f0] *Color of the icons when the mouse rolls over them.*  
 albumWindowInfoSize **number** 1 - 1000 [12] *Size of the info text.*  
 albumWindowInfoColor **color** [ffffff] *Color of the info text.*  
 albumWidth **number** 5 - 1000 [50] *Width of the individual thumbnails in pixels.*

**albumHeight** **number** 5 - 1000 [50] *Height of the individual thumbnails in pixels.*  
**albumRoundedCorners** **number** 0 - 100 [3] *Value specifying how much the corners are rounded.*  
**albumSpacing** **number** 0 - 1000 [10] *Spacing between albums in pixels.*  
**albumShadowSize** **number** 0 - 10 [2] *Shadow size.*  
**albumShadowColor** **color** [000000] *Shadow color.*  
**albumShadowAlpha** **number** 0 - 100 [50] *Transparency value.*  
**albumHoverDistance** **number** 0 - 100 [8] *Distance to hover the thumbnails when the mouse rolls over them.*  
**albumHoverShadowSize** **number** 0 - 10 [8] *Shadow size to morph to when the mouse rolls over them.*  
**albumBrightnessAdjustment** **number** -255 - 255 [0] *Brightness adjustment of the album icon and info area*  
**albumHoverBrightnessAdjustment** **number** -255 - 255 [128] *Brightness adjustment of the album icon and info area when the mouse hovers over them*  
**albumInfoWidth** **number** 5 - 1000 [150] *Width of the album info text in pixels.*  
**albumInfoHeight** **number** 5 - 1000 [50] *Height of the album info text in pixels.*  
**albumInfoMargin** **number** 0 - 100 [2] *Margin between thumbnail and info text in pixels.*  
**albumInfoPadding** **number** 0 - 100 [5] *Padding in pixels.*  
**albumInfoRoundedCorners** **number** 0 - 100 [3] *Value specifying how much the corners are rounded.*  
**albumInfoColor** **color** [10a4f0] *Color of the album info text background.*  
**albumInfoAlpha** **number** 0 - 100 [75] *Transparency value of the album info text background.*  
**albumInfoShadowSize** **number** 0 - 10 [2] *Shadow size.*  
**albumInfoShadowColor** **color** [000000] *Shadow color.*  
**albumInfoShadowAlpha** **number** 0 - 100 [50] *Transparency value.*  
**albumInfoTextAlign** **text** "left", "right", "center" [left] *Alignment of text.*  
**albumInfoTitleSize** **number** 1 - 1000 [12] *Font size of the title text.*  
**albumInfoTitleColor** **color** [ffffff] *Font color of the title text.*  
**albumInfoTitleContainsNumber** **true, false** [true] *If true, the album info will contain the index of the current album and the number of images in the album.*  
**albumInfoDisplayNoTitle** **text** "No title" *Display "No title" if the image doesn't have a title.*  
**albumInfoTitleMaxCharacters** **number** -1 - 1000 [-1] *Font size of the*  
**albumInfoDescriptionMargin** **number** -1000 - 1000 [0] *Margin between title and description.*  
**albumInfoDescriptionSize** **number** 1 - 1000 [10] *Font size of the description text.*  
**albumInfoDescriptionColor** **color** [ffffff] *Font color of the description text.*  
**albumInfoDescriptionMaxCharacters** **number** -1 - 1000 [-1] *Limit the number of characters displayed in the description. If the value is -1, there's no limit.*  
**albumInfoDisplayNoDescription** **text** "No description" *Display "No description" if the image doesn't have a description.*

## The controls object

**controlAlign** **text** "topLeft", "topRight", "bottomLeft", "bottomRight", "bottomCenter", "topCenter" [bottomCenter] *How to position the image info in the slideshow.*  
**controlMarginX** **number** 0 - 1000 [10] *Horizontal margin from the left or right side in pixels.*  
**controlMarginY** **number** 0 - 1000 [10] *Vertical margin from the top or bottom side in pixels.*  
**controlTransitionHorizontal** **true, false** [true] *If true, the control hides away horizontally if its alignment is not centered.*

**controlColor** *color* [000000] *Color of the control object.*  
**controlAlpha** *number* 0 - 100 [75] *Transparency value.*  
**controlPadding** *number* 0 - 100 [11] *Padding in pixels.*  
**controlIconSize** *number* 5 - 100 [12] *Size of the icons in pixels.*  
**controlIconSpacing** *number* 0 - 100 [8] *Spacing between icons in pixels.*  
**controlIconColor** *color* [ffffff] *Color of the icons.*  
**controlIconRollOverColor** *color* [10a4f0] *Color of the icons when the mouse rolls over them.*  
**controlDelay** *number* 1 - 60 [2] *Delay before the controls fade away in seconds.*  
**controlFadeInAreaSize** *number* 0 - 1000 [30] *Determine the size of the invisible area where the mouse has to roll over before the controls pop up.*  
**controlLineWidth** *number* 0 - 20 [2] *Line width (stroke) of the control object.*  
**controlLineColor** *color* [ffffff] *Color of the line.*  
**controlRoundedCorners** *number* 0 - 100 [15] *Value specifying how much the corners are rounded.*  
**controlAutoHide** *true, false* [false] *If true, the controls hide away after "controlDelay" seconds.*  
**controlShowOnStartDelay** *number* 0 - 60 [3] *If > 0, the control object is displayed once during the given number of seconds.*  
**controlShadowSize** *number* 0 - 10 [5] *Size of the shadow.*  
**controlShadowColor** *color* [000000] *Color of the shadow.*  
**controlShadowAlpha** *number* 0 - 100 [75] *Transparency value.*

## The loading icon object

**LoadingIconAlign:** "topLeft", "topRight", "bottomLeft", "bottomRight", "leftCenter", "rightCenter", "bottomCenter", "topCenter", "center" [bottomRight] *How to position the image info in the slideshow.*  
**loadingIconMarginX** *number* 0 - 1000 [15] *Horizontal margin from the left or right side in pixels.*  
**loadingIconMarginY** *number* 0 - 1000 [15] *Vertical margin from the top or bottom side in pixels.*  
**loadingIconSize** *number* 0 - 100 [25] *Size of the loading icon in pixels.*  
**loadingIconColor** *color* [ffffff] *Color of the loading icon.*  
**loadingIconBackgroundColor** *color* [000000] *Color of the loading icon background.*  
**loadingIconAlpha** *number* 0 - 100 [75] *Transparency value of the loading icon.*  
**loadingIconLineWidth** *number* 0 - 20 [2] *Line width (stroke) of the control object.*  
**loadingIconLineColor** *color* [ffffff] *Color of the line.*  
**loadingIconShadowSize** *number* 0 - 10 [5] *Size of the shadow.*  
**loadingIconShadowColor** *color* [000000] *Color of the shadow.*  
**loadingIconShadowAlpha** *number* 0 - 100 [50] *Transparency value.*  
**loadingIconDelayBeforeFadeIn** *number* 0 - 60 [0.2] *Determines after how many seconds the loading icon will start to fade in. If the image is loaded before this delay, the loading icon won't show up.*

## The mark object

**markFile** *url* *Url to the file containing the mark. You can load .SWF files and .JPG files. Tip: for creating transparent overlays, you could place a .PNG file inside a .SWF file and load the .SWF.*  
**markAlign:** "topLeft", "topRight", "bottomLeft", "bottomRight", "leftCenter", "rightCenter", "bottomCenter", "topCenter", "center" [topRight] *How to position the image info in the slideshow.*  
**markMarginX** *number* 0 - 1000 [0] *Horizontal margin from the left or right side in pixels.*  
**markMarginY** *number* 0 - 1000 [0] *Vertical margin from the top or bottom side in pixels.*  
**markTransitionTime** *number* 0 - 1000 [0.5] *Time of transition in seconds.*



*All contents © 2006 Monokai, all rights reserved. Monoslideshow and the Monoslideshow logo are trademarks of Monokai. All other trademarks, service marks and logos used in this document are the trademarks, service marks or logos of their respective owners.*

[www.monoslideshow.com](http://www.monoslideshow.com)  
[www.monokai.nl](http://www.monokai.nl)