mono SLIDESHOW

User Manual
v1.32, rev1

Contents

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 on 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">
      <img src="photo1.jpg" thumbnail="thumb1.jpg" />
      <img src="photo2.jpg" thumbnail="thumb2.jpg" />
      <img src="photo3.jpg" thumbnail="thumb3.jpg" />
   </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>`. It's 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 src="photo1.jpg"></img>, you could also write <img src="photo1.jpg" />. 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">
      <img src="photo1.jpg" thumbnail="thumb1.jpg" />
      <img src="photo2.jpg" thumbnail="thumb2.jpg" />
      <img src="photo3.jpg" thumbnail="thumb3.jpg" />
   </album>
   <album thumbnail="album2.jpg" imagePath="album2" thumbnailPath="thumbs">
      <img src="photo1.jpg" />
      <img src="photo2.jpg" />
      <img src="photo3.jpg" />
      <img src="photo4.jpg" />
      <img src="photo5.jpg" />
      <img src="photo6.jpg" />
   </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. The tag is self-closing and consists only of attributes. Here you would insert the attributes which you'd like to change for the entire slideshow. 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). So, 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*.

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&reg; 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>
```

Disable version info

By default, Monoslideshow displays the version info in the context menu. This menu is shown if you right-click on the slideshow. To disable this menu, you have to set the Flash® parameter showVersion to `false`. Just like before, paste this code in your web page:

```
script type="text/javascript">
var so = new SWFObject("monoslideshow.swf", "mss", 480, 360, "7", "#ffffff");
so.addVariable("showVersionInfo", "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">
   <img src="photo1.jpg" />
   <img src="photo2.jpg" id="idName" />
   <img src="photo3.jpg" />
</album>
```

Loading Flickr RSS Feeds

Monoslideshow supports Flickr RSS feeds. You can find links to these .RSS feeds on group pages and personal pages on *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:

`size` `"tiny"` `"thumbnail"` `"small"` `"medium"` `"large"` `[medium]` *Force the size of the images of a Flickr .RSS feed. The large version only exists for very large original images.*
`linkToImage` `true,` `false` `[false]` *If true, all images will be clickable, leading the user to a new window where the original image is displayed.*
`linkToImageSize` `"tiny"` `"thumbnail"` `"small"` `"medium"` `"large"` `[medium]` *Force the size of the image that pops up if* `linkToImage` *is set to true. The large version only exists for very large original images.*

Loading Monoslideshow inside another .SWF-file

You can dynamically load `monoslideshow.swf` inside another .swf-file. To do this, you can insert this code in your main .swf-file:

```
function loadingFinished(target:MovieClip):Void {
   var monoslideshow:Object = new Monoslideshow(target, "monoslideshow.xml");
   monoslideshow.start(Stage.width, Stage.height, true);
}

var listener:Object = new Object();
var movieClipLoader:MovieClipLoader = new MovieClipLoader();
listener.onLoadInit = loadingFinished;
movieClipLoader.addListener(listener);
this.createEmptyMovieClip("holder", this.getNextHighestDepth());
movieClipLoader.loadClip("monoslideshow.swf", this.holder);
```

Once `monoslideshow.swf` is loaded, the class `Monoslideshow` comes available in _global. You can then instantiate this class and call the `start` function on the instance. This function takes three parameters which specify the width, the height of Monoslideshow and showLogo. So to disable the logo, you have to write this:

```
monoslideshow.start(Stage.width, Stage.height, false);
```

Loading custom fonts

Monoslideshow can dynamically load fonts from external .SWF-files. The process is as follows:

1. Create a .SWF-file containing one or more fonts.
2. Load this .SWF-file in Monoslideshow.
3. Specify your custom font names in monoslideshow.xml.

You can create the .SWF-files either with the standard Flash® IDE, or with SWFMill (free and open source). Using SWFMill is more reliable then using the Flash® IDE, but SWFMill can only load .TTF files. Let's quickly review both ways.

Create a font file using Flash® IDE (experimental)

In the Flash® IDE, you first have to create a new file. Then, in the library, add as many fonts as you wish (place your mouse on the library window, right click and select *"new font"*).

The next step is to give each font symbol in the library an identifier name you can use in `monoslideshow.xml`. You also have to tick "export for runtime sharing" and specify the url of the exported .SWF-file. This url should be relative to the .SWF-file that's loading it, in this case, `monoslideshow.swf`. So, for each font symbol, right click, select *"linkage properties"* and do the following:

- set Export for runtime sharing
- set Export in first frame
- set Identifier name (i.e. *"verdana"*)
- set URL (i.e. *"fonts.swf"*)

After this, you have to create an empty MovieClip in the library, called "ForceShared". in the linkage properties, do the following:

- set Import for runtime sharing
- set identifier name to "ForceShared"
- set URL (i.e. *"fonts.swf"*)

You then have to place the symbol ForceShared on the stage. After this, the Flash® file is ready to publish.

Create a font file using SWFMill

SWFMill takes as input an .XML-file and from this it produces a .SWF-file. It's operated from the command line. SWFMill is open source and there are binaries available for both MacOSX and Windows. You can download SWFMill for free from *www.swfmill.org*.

SWFMill is commonly used to create asset libraries that contain graphics, fonts and sounds. We'll use it here for the sole purpose to create an asset library of fonts. Let's take a look at the following example of an SWFMill .XML-file:

```
<?xml version="1.0" encoding="utf-8"?>
<movie width="1" height="1" framerate="40">
   <frame>
       <library>
              <font name="arial" import="arial.ttf" />
              <font name="verdana" import="verdana.ttf" />
       </library>
       <Import url="fonts.swf"></Import>
   </frame>
</movie>
```

The above piece of code will generate an .SWF-file with a library that contains two fonts, *arial* and *verdana*. The `import` attribute of the `<font>` tag says where the font is located. The `name` attribute says what this font should be called. The font names in `monoslideshow.xml` should exactly match these `name` values.

Furthermore, the `<Import>` tag makes the fonts of this .SWF-file available for other .SWF-files that load this file. The `url` attribute should specify where the file is placed relative to the .SWF-file that loads it. So, in this example, `fonts.swf` should be placed in the same folder as `monoslideshow.swf`.

To convert the .XML-file above to a .SWF-file, type the following on the command line:

```
swfmill simple fonts.xml fonts.swf
```

Load the font file in Monoslideshow

Before Monoslideshow can display your custom fonts, you have to specify which font file to load. Generally, you would place the generated `fonts.swf` in the same folder as `monoslideshow.swf`. After this, you have to paste the following in `monoslideshow.xml`:

```
<preferences
   fontFile = "fonts.swf"
/>
```

Specify font names

Inside `monoslideshow.xml`, you can now specify your own font names for the `font` attribute. The global `font` attribute will be overridden by the following font attributes:

```
imageInfoTitleFont
imageInfoDescriptionFont
albumWindowInfoFont
thumbnailWindowInfoFont
albumInfoTitleFont
albumInfoDescriptionFont
```

Note: to specify a font name, use the real font name, not the identifier name.

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. Note: the sounds are preloaded to ensure direct playback. The more sounds you define in the XML file, the longer it lasts before the slideshow starts.

Main attributes

`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”, “albumsThenThumbnails”` `[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.*

Sounds

`hoverSound` `url` *link to an mp3 file that is played when hovering over a control button. This is the global sound for all hover actions. It is overridden if you specify one of the other hover sounds below.*
`selectSound` `url` *link to an mp3 file that is played when pressing a control button. This is the global sound for all select actions. It is overridden if you specify one of the other select sounds below.*
`albumWindowButtonHoverSound` `url` *link to an mp3 file that is played when you hover over the button for opening the album window.*
`albumWindowButtonSelectSound` `url` *link to an mp3 file that is played when you press the button for opening the album window.*
`thumbnailWindowButtonHoverSound` `url` *link to an mp3 file that is played when you hover over the button for opening the thumbnail window.*
`thumbnailWindowButtonSelectSound` `url` *link to an mp3 file that is played when you press the button for opening the thumbnail window.*
`muteButtonHoverSound` `url` *link to an mp3 file that is played when you hover over the button for muting the slideshow.*
`muteButtonSelectSound` `url` *link to an mp3 file that is played when you press the button for muting the slideshow.*
`previousButtonHoverSound` `url` *link to an mp3 file that is played when you hover over the previous button.*
`previousButtonSelectSound` `url` *link to an mp3 file that is played when you press the previous button.*
`pauseButtonHoverSound` `url` *link to an mp3 file that is played when you hover over the pause / play button.*
`pauseButtonSelectSound` `url` *ink to an mp3 file that is played when you press the pause button.*

playButtonSelectSound url *ink to an mp3 file that is played when you press the play button.*
nextButtonHoverSound url *link to an mp3 file that is played when you hover over the next button.*
nextButtonSelectSound url *link to an mp3 file that is played when you press the next button.*
albumHoverSound url *link to an mp3 file that is played when you hover over the album button.*
albumSelectSound url *link to an mp3 file that is played when you press the album button.*
albumWindowCloseButtonHoverSound url *link to an mp3 file that is played when you hover over the close button in the album window.*
albumWindowCloseButtonSelectSound url *link to an mp3 file that is played when you press the close button in the album window.*
albumWindowPreviousButtonHoverSound url *link to an mp3 file that is played when you hover over the previous button in the album window.*
albumWindowPreviousButtonSelectSound url *link to an mp3 file that is played when you press the previous button in the album window.*
albumWindowNextButtonHoverSound url *link to an mp3 file that is played when you hover over the next button in the album window.*
albumWindowNextButtonSelectSound url *link to an mp3 file that is played when you press the next button in the album window.*
thumbnailHoverSound url *link to an mp3 file that is played when you hover over the thumbnail button.*
thumbnailSelectSound url *link to an mp3 file that is played when you press the album button.*
thumbnailWindowCloseButtonHoverSound url *link to an mp3 file that is played when you hover over the close button in the thumbnail window.*
thumbnailWindowCloseButtonSelectSound url *link to an mp3 file that is played when you press the close button in the thumbnail window.*
thumbnailWindowPreviousButtonHoverSound url *link to an mp3 file that is played when you hover over the previous button in the thumbnail window.*
thumbnailWindowPreviousButtonSelectSound url *link to an mp3 file that is played when you press the previous button in the thumbnail window.*
thumbnailWindowNextButtonHoverSound url *link to an mp3 file that is played when you hover over the next button in the thumbnail window.*
thumbnailWindowNextButtonSelectSound url *link to an mp3 file that is played when you press the next button in the thumbnail window.*

Attributes for images

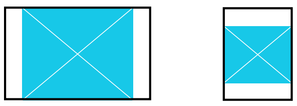
Every image tag <img> can have the attributes specified below. Every attribute below could also be placed in the <preferences> or the <album> tag to specify global values for each of the images the tag contains (with the exception of title, description, src, thumbnail, link and target).

Main attributes

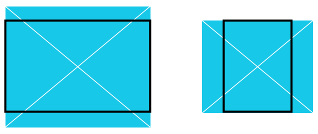
title text *Title of the image.*
description text *Description of the image.*
src url *Source of the image. Depends on "imagePath" defined in the <album> tag.*
thumbnail url *Link to a thumbnail of the image. Depends on "thumnailPath" defined in the <album> tag.*
link url *When the image is clicked with 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. Depends on "soundPath" defined in the <album> tag.*
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:

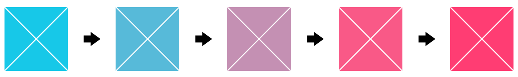


scaleToFill:

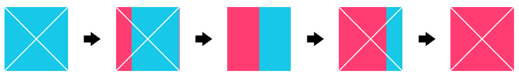


imagePause number 0.5, 86400 [3] *Time to wait in seconds until the next image is displayed.*
imageTransitionTime number 0.1 - 1000 [0.5] *Time of transition in seconds.*
imageTransition: "blend", "leftToRight", "rightToLeft", "topToBottom", "bottomToTop", "leftToRightBlend", "rightToLeftBlend", "topToBottomBlend", "bottomToTopBlend", "leftToRightFadeOutBackwards", "rightToLeftFadeOutBackwards", "topToBottomFadeOutBackwards", "bottomToTopFadeOutBackwards", "pinhole", "pinholeBlend", "fadeInOut", "bubbles", "bubblesBlend", "photoFlash", "starWipe", "starWipeBlend", "noTransition" [blend] *Choose a transition type. Some examples:*

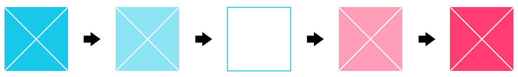
blend:



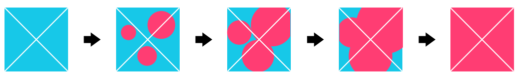
leftToRightBlend:



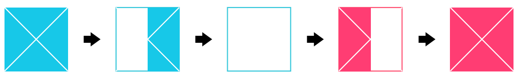
fadeInOut:



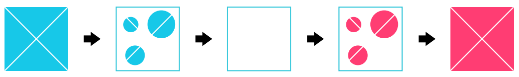
bubblesBlend:



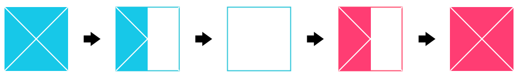
leftToRight:



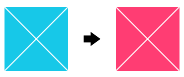
bubbles:



leftToRightFadeOutBackwards:



noTransition:



kenBurnsMode “random”, “randomPan”, “randomZoomIn” “randomZoomOut”, “none” [none] *The Ken Burns effect slowly zooms or pans accross the image. Note: if active,* imageScaleMode, imageMarginX, imageMarginY *and* imageAlign *will have no effect. The Ken Burns effect scales the images according to* scaleToFit, *the margins will be zero 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 wil 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.*
imageInfoTitleFont text *Font of the title (if other than "unibody” or "helvetica”, please refer to the instructions about customizing fonts).*
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.*
imageInfoDescriptionFont text *Font of the description (if other than "unibody” or "helvetica”, please refer to the instructions about customizing fonts).*
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, size and linkToImageSize).

Main attributes

font "helvetica", "unibody" [ffffff] *The global font used in this album. Overridden by the attributes "imageInfoTitleFont", "imageInfoDescriptionFont", "albumWindowInfoFont", "thumbnailWindowInfoFont", "albumInfoTitleFont" and "albumInfoDescriptionFont".*
title text *Title text.*
description text *Description text.*
thumbnail url *link to the thumbnail.*
id text *Specify an ID name to use with startWithAlbumID.*
imagePath url *Path to the folder containing the images.*
thumbnailPath url *Path to the folder containing the thumbnails.*
soundPath text *Path to the folder containing the sound files.*
backgroundMusic url *A link to an mp3 file for the background music. You can specify multiple mp3 files by separating them with a comma.*
backgroundMusicVolume number 0 - 100 [100] *Volume of the background music.*
backgroundMusicLoop true, false [true] *If true, the background music loops.*
backgroundMusicFadeIn true, false [false] *If true, the background music fades in during the time the first image in the album is shown.*
backgroundMusicFadeOut true, false [false] *If true, the background music fades out during the time the last image in the album is shown.*
backgroundMusicAlbumWindowVolume number 0 - 100 [100] *Specifies the volume of the background music when the album window is displayed.*
backgroundMusicAlbumWindowFadeTime number 0 - 1000 [1] *Specifies the fade in and fade out time of the background music when the album window is opened or closed.*
backgroundMusicThumbnailWindowVolume number 0 - 100 [100] *Specifies the volume of the background music when the thumbnail window is displayed.*
backgroundMusicThumbnailWindowFadeTime number 0 - 1000 [1] *Specifies the fade in and fade out time of the background music when the thumbnail window is opened or closed.*
startMuted true, false [false] *If true, the slideshow starts with sound disabled.*
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", "showAlbumWindow" [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. To go to the album selection screen after all photos have been shown, use "showAlbumWindow".*
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.*
viewport x1, y1, x2, y2 [0, 0, maxWidth, maxHeight] *Defines a rectangle of the viewport through which the images are displayed. You can set the rectangle by specifying the top left and bottom right coordinates separated by comma's.*

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.*
thumbnailWindowAlwaysOn true, false [false] *Doesn't hide the thumbnail window, but permanently displays it.*
thumbnailWindowTrack true, false [true] *If "thumbnailWindowAlwaysOn" is set to "true", the thumbnail window will display the page that contains the current image.*
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.*
thumbnailWindowInfoFont text *Font of the info text (if other than "unibody" or "helvetica", please refer to the instructions about customizing fonts).*
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.*
albumWindowInfoFont text *Font of the info text (if other than "unibody” or "helvetica”, please refer to the instructions about customizing fonts).*
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.*
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 "left", "right", "center" [left] *Alignment of text.*
albumInfoTitleFont text *Font of the title (if other than "unibody" or "helvetica", please refer to the instructions about customizing fonts).*
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.*
albumInfoDescriptionFont text *Font of the description (if other than "unibody" or "helvetica", please refer to the instructions about customizing fonts).*
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 "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.*