

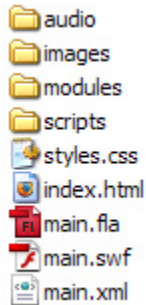
# XML Portfolio Template v4 Help

## Table of Contents

1. Intro – p.2
2. Site Customization – p.2
3. Creating Navigation – p.3
4. Using Modules – p.4
  - a. Slideshow Module – p.4
  - b. Projects Module – p.5
  - c. Traditional Gallery Module – p.5
  - d. Scrolling Gallery Module – p.6
  - e. Video Gallery Module – p.6
  - f. MP3 Player Module – p.7
  - g. Contact Module – p.7
  - h. Staff Module – p.8
  - i. Simple News Module – p.8
  - j. Traditional News Module – p.8
  - k. Simple Text Module – p.8
  - l. Text + Image Module – p.8
  - m. Text + Sections Module – p.8
5. Using your own SWF files – p.8
6. Linking to other web pages with links – p.8
7. Inner-linking within text areas – p.9
8. How to upload to your server – p.9

## 1. Intro

The XML Portfolio Template v4 is a full-Flash website that can be easily customized with XML. All FLA files are saved in Flash Version 8. For best results it is best that the site builder is familiar with the concept of modifying XML. The template comes with sample data which will make things easier for you as you go about changing its contents. The template's files are stored in the /Source folder.



The template uses **main.fla** as the root, which uses **main.xml** for its main configurations – plus all of the site's navigation. The stylesheet called **styles.css** controls the text appearance for all text areas including all of the modules. The file called index.html is the HTML wrapper that displays the Flash template and all of its contents. Here is where you will need to change the title and meta data, etc.

### Folders

- /audio – contains the background music
- /images – contains your logo file and background images
- /modules – contains all of the modules source files and data
- /scripts – contains swfobject.js

## 2. Site Customization

Many aspects of the template can be customized just with XML. In most cases, you can customize it to your liking without having to open Flash. However, not everything can be controlled with XML because the amount of options to customize would be overwhelming to the average user. So in certain cases Flash will be needed to make extra customizations.

Let's examine **main.xml** located in the /Source folder. This controls the main template and global elements.

Here are the elements that control the site's visual attributes.

<mods

```
    navPadding="25" – Spacing between links
    myLogo = "images/logo.png" – Location of your logo (png, swf, jpg, and gif are accepted)
    backgroundMusic = "yes" – Enable/Disable background music "no" and "yes" are acceptable
    myMusic = "audio/adg3.com_crypticPsyche.mp3" – Location of your background music

    navActiveColor = "0xFFFFFF" – Color of links
```

```

    navRollOverColor = "0xff7566" – Color of links rolled over

    navBGOutlineColor = "0xff0000" – Color of main navigation bar outline
    navBGtopGradientColor = "0xa91100" – Color of the main navigation bar – top
part of the gradient
    navBGbottomGradientColor = "0x5d0a00" – Color of the main navigation bar –
bottom part of the gradient

    subnavActiveColor = "0xcccccc" – Color of sub links
    subnavRollOverColor = "0xff0000" – Color of sub links rolled over

    subnavBGOutlineColor = "0x666666" – Color of sub nav bar outline
    subnavBGtopGradientColor = "0x333333" – Color of sub nav bar – top part of
the gradient
    subnavBGbottomGradientColor = "0x121212" – Color of sub nav bar – bottom
part of the gradient

    footerColor = "0x000000" – Color of the footer background
    navFontSize = "8" – Font size of main navigation
    subnavFontSize = "8" – Font size of sub navigation
    logoX = "25" – Position of logo horizontally
    logoY = "40" – Position of logo vertically

    />

```

This does it for customizing the visuals for the main part of the site. Changing modular content will be discussed later on. A question I might hear is: “How do I change the font in the navigation?” Answer: In main.fla, you need to open the movie clip called “link” in the library and change it there.

### 3. Creating Navigation

We still have main.xml open. This is one of the great things about v4 – it's that the main configurations and site navigation are contained in one file. Our links are in the <navigation> tags. A <link> tag indicates a new main navigation item. A <sublink> tag indicates a new sub navigation item – this must be nested in a <link>...</link>.

<links> and <sublinks> work in the same way and they both have 3-4 parts. 3 of the 4 parts are required:

```

    name = "Your link name"
    module = "module location"
    data = "module data location"
    background = "background image for this link" – this is optional.

```

Let's examine a navigation link.

```

<link name="HOME" module="modules/slideshow/slideshow.swf"
data="modules/slideshow/slideshow.xml"
background="images/background1.jpg" />

```

The above example is showing a home page link. The name of the link is “HOME”, which is displayed as text in the navigation bar. The module is pointing to the slideshow SWF file. The

data is pointing to the slideshow XML file. This particular link has a background image. If you do not desire a background image for your link, you can remove the background attribute altogether like the MUSIC link in the sample main.xml.

Let's dissect this further...

```
<link name="HOME" - name of the link  
module="modules/slideshow/slideshow.swf" - location of the module you wish to use  
data="modules/slideshow/slideshow.xml" - location of the data the module needs to  
load  
background="images/background1.jpg" /> - location of this link's background (can be  
jpg, png, swf, gif or flv)
```

Each link and sublink needs a name, module and data attribute. If you want to load a slideshow for example, you need to specify the path and file name like above. Now, that module needs to load XML data as well. So you need to specify the data location as well. The great thing is about this is that if you want to have several slideshows or multiple instances of any module, all you need to do is create another XML file (e.g. slideshow2.xml or myslideshow.xml) and link the data attribute to it.

Every <link> can have sub links. When you desire a sub link, you must format it properly. For example:

```
<link>  
    <sublink />  
</link>
```

A link without a sub link looks like:

```
<link />
```

Be mindful of this that it doesn't look like, or it will break.:

```
<link />  
    <sublink />  
</link>
```

Tip: If you open your main.xml file in Firefox, it will tell you if there is an error in your XML file. So if you're experiencing problems, try this method of problem solving.

## 4. Using Modules

Now that you're familiar with creating links, we'll discuss the modules. There are many modules that can be taken advantage of. Some of which you may find very useful. You can link to just the ones that you need. All modules are stored in the modules/ folder – and each module resides in its own folder.

### 4 a. Slideshow Module

Location: modules/slideshow/

This is what you see when you view the template's home page. This module requires an XML file to run. There is currently a file called `slideshow.xml` that is used for the home page. You can easily duplicate that file and change it to make another slideshow. The slideshow uses fixed-size images (940x360).

Let's examine parts of XML file structure of **slideshow.xml**

```
<slideshow
  speed="5"   - Number of seconds between slides
  fadeInSpeed="2" - Number of seconds it takes to fade in
  fadeOutSpeed="1" - Number of seconds it takes to fade out
  captions="on" - Enables/Disables text captions
  links="off" - Enables/Disables the slide having it's own link
  buttons="on" - Enables/Disables the thumbnails for loading new slides
>
<slide>
  <asset>image1.jpg</asset> - Image for slide (can use jpg, png, gif)
  <caption><![CDATA[v4]]></caption> - Caption text
  <url><![CDATA[http://flashden.net/]]></url> - Slide link
</slide>
```

## 4 b. Projects Module

Location: `modules/projects/`

This module is my favorite. It is very useful for showing many different projects – and each project has its own images, videos and text information. Like all other modules, this has its own XML file.

Let's take a look at `projects.xml`

```
<project title="Project Title 1" client="Client Name" date="January 1,
2009" url="http://www.flashden.net/">
  <media thumbnail="media/project1.jpg">
    <asset src="media/video1.flv" />
    <asset src="media/large2.jpg" />
    <asset src="media/video2.flv" />
  </media>
  <description><![CDATA[Text information... ]]></description>
</project>
```

Each project has its own title, name, date, url (optional) and media. JPG, PNG, SWF, GIF and FLV are supported media types. You can have several items for your media. You can have as many projects as you wish.

## 4 c. Traditional Gallery Module

Location: `modules/galleries-traditional/`

This module is an image and video gallery. It is useful for photography and video portfolios. I've created several XML files that contain different galleries to reinforce the idea that you can easily create more galleries and modules just by creating a new XML file.

Here's **gallery.xml**

```
<asset type="image" thumbnail="images/thumbnails/large1.jpg"
large="images/large1.jpg">
    <description><![CDATA[Test description 1]]></description>
</asset>
```

Each asset has a **type**. You must specify if it's an "image" or "video". You then specify your thumbnail location. The thumbnails aren't necessarily a fixed size, but do try to maintain some level of consistency to improve the appearance of your thumbnails. Then you specify your large image/video file. For the description, you may add as much text as you please.

## 4 d. Scrolling Gallery Module

Location: modules/galleries-scrolling/

This module functions as a gallery that show large thumbnails that are scrolled horizontally. Each thumbnail can be clicked to show a large image or video.

Let's take a look at **gallery1.xml**

```
<window type="image" picture="images/image2.jpg">
    <title><![CDATA[<b>Image Title</b>]]></title>
    <asset>images/large2.jpg</asset>
    <caption><![CDATA[Short text info]]></caption>
    <longdescription>Long text info</longdescription>
</window>
```

Each thumbnail that is loaded is contained in a <window> tag. Each must specify a type "image" or "video" are supported.

The **picture=""** attribute is what loads the thumbnail

The **title** and caption display as text below the thumbnail

The **asset** is the large version that is displayed when the thumbnail is clicked

**Longdescription** is the text information that is displayed when the enlarged version is active

## 4 e. Video Gallery Module

Location: modules/galleries-video/

This module only showcases videos. The thumbnails and short descriptions are shown on a pane on the left, when one is clicked, it will play a video to the right. Each video player has an "i" icon that when clicked, will reveal a long description.

**videos.xml**

```
<item>
    <title><![CDATA[Video Name 1]]></title>
    <video>videos/video1.flv</video>
    <thumbnail>thumbnails/video1.jpg</thumbnail>
    <blurb><![CDATA[Short description]]></blurb>
    <description><![CDATA[Long description....]]></description>
</item>
```

The **title**, **thumbnail** and **blurb** elements are displayed as thumbnails on the left. The **video** and **description** are shown on the right when a video has been selected.

#### 4 f. MP3 Player Module

Location: modules/mp3player/

This module is essentially a playlist of MP3s.

Let's examine part of **album1.xml**

```
<album artwork="background.jpg" autoplay="yes">
<song>
  <artist><![CDATA[ADG3Studios]]></artist>
  <trackname><![CDATA[Busted Chump]]></trackname>
  <file>adg3.com_bustedChump.mp3</file>
  <duration>1:25</duration>
</song>
```

artwork – the background image that is behind this module

autoplay – enables/disables the module from automatically playing the first song

artist- artist text info

rest is self-explanatory

#### 4 g. Contact Module

Location modules/contact/

This module sends you mail from the contact form using PHP and shows some text information.

**contact.php**

```
<?php
# Send to your email address
$sendTo = "youremailaddress@yourdomain.com";
# Subject line
$subject = "Information Request";
# Send from address
$sendFrom = "FROM: info@yourdomain.com";
# Body
$body = "A user has left the following information \n \n Name: " .
stripslashes($_POST["yourName_txt"]) . " \n Phone #: " .
stripslashes($_POST["phone_txt"]) . " \n Email Address: " .
stripslashes($_POST["email_txt"]) . " \n Comments: " .
stripslashes($_POST["comments_txt"]);
# Send mail
mail($sendTo, $subject, $body, $sendFrom);
?>
```

**contactinfo.xml**

```
<contactinfo>
  <info><![CDATA[<b>My Company Name</b><BR>1234 Street Dr<BR>Suite
200<BR>Dallas, TX 75206<BR>P: (972) 123 4567<BR>F: (972) 123 7890<BR><a
href="mailto:info@mycompany.com">info@mycomany.com</a><BR>Special Characters
ääâ<B>ääß</B> é <l>ê</l> í <B><l>ü</l></B>]]>
  </info>
</contactinfo>
```

- Just text content.

#### **4 h. Staff Module**

Location modules/staff/

I call this the staff module because I found it useful for showing employees. I have seen many people use this module for many different purposes.

#### **4 i. Simple News Module**

Location modules/news-simple/

I call this a simple news module, but it can be used for many other text display purposes.

For this and the remaining modules, they follow very common rules for using text and images that don't need explaining.

#### **4 j. Traditional News Module**

Location modules/news-traditional/

#### **4 k. Simple Text Module**

Location modules/text-simple/

#### **4 l. Text + Image Module**

Location modules/text-image/

#### **4 m. Text + Sections Module**

Location modules/text-sections/

### **5. Using your own SWF files**

Chances are, you've purchased some other really cool files from Flashden that you'd like to integrate in this template. Please note this template is created in ActionScript 2. So any SWF you use that uses ActionScript 3 will not be compatible.

To avoid certain path issues, place your own SWFs in the /Source/ folder along with the main site files. To link to your own swf you need to specify the module="yourswffile.swf" (or whatever it's called). You must leave the data attribute there, but it must be left blank like this:

```
<link name="My SWF" module="myfile.swf" data="" />
```

### **6. Linking to other web pages with links**

To have a link that you'd like to link up to another website or web page format your link or sublink like this:

```
<link name="My Blog" module="url" data="http://blog.flashden.net/" />
```



## 7. Inner-linking within text areas

Let's take a look at /Source/modules/text-simple/about.xml to see what this means.

You will notice a link that says "View Our Projects"

This is a link within text that can navigate a user to a module within this template.

It is treated very much like how we create links and sublinks since we specify a module and module data

Example:

```
<a  
href="asfunction:_root.catchLink,modules/projects/projects.swf,modules/  
projects/projects.xml">View Our Projects</a>
```

In you're <a href> tag, you must call asfunction:\_root.catchLink.

This is the function in Flash that will detect this event and execute it according to your link.

Then after your comma, you specify your module, then you specify your module data.

## 8. How to upload to your web server

Upload all of the contents inside the folder called /Source/ so that index.html is in the root level of your server. Please do not upload the FLA files.