

Creating a Gallery that gets it's images dynamically from Flickr

Intro

In this tutorial we'll show you how easy it is to create a dynamic photo album with <u>3D ImageFlow Gallery For</u> <u>Flash</u> that loads the images from Flickr. You can even use the search function of Flickr and display the photos inside your gallery! Below is a screenshot of the gallery that we're going to create.





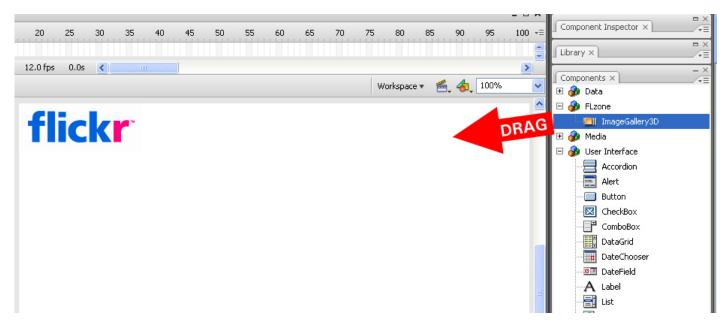


Building the Gallery

1. Install the <u>3D ImageFlow Gallery For Flash</u> extension and create a **new Flash document** (ActionScript 2.0) and **save the page**.

New Document	
Seneral Templates Type: Image: Flash File (ActionScript 3.0) Image: Flash File (ActionScript 2.0) Image: Flash File (Mobile) Image: Flash File (Mobile) Image: Flash Slide Presentation Image: Flash File (Mobile) Image: Flash Slide Presentation Image: Flash File (Mobile) Image: Flash Slide Presentation Image: Flash Form Application Image: Flash Form Application Image: ActionScript File Image: Flash JavaScript File Image: Flash JavaScript File Image: Flash Project	Description: Create a new Flash document (*.fla) in the Flash Document window. The Publish Settings will be set for ActionScript 2.0. Use Flash documents to set up the media and structure for Flash movies and applications.
	OK Annuleren

- 2. Set the stage size to the desired width and height, we use a stage size of 680 x 520
- 3. Select File > Import > Import to Stage.
- 4. Select the flickr logo (logo_home.png).
- 5. Drag the ImageGallery3D component from the components panel to the Stage.



Tip: Instead of dragging the component, you can also Double-click the ImageGallery3D component.

6. Select the ImageGallery3D component on the Stage. In the Property inspector, enter the instance name **flickrGallery and** set **W** to **680**, **H** to **450**, **X** to **0** and **Y** to **0**.

¢ Pr	oper	ties × Filters Parame	eters			- × +=
	0	¥	Instance of: ImageGallery3D	Color:	None	? Ø
		flickrGallery	Swap			0
	w	680.0 X: 0.0		Blend:	Normal	0
ĥ	н	450.0 Y: 0.0			Use runtime bitmap caching	llt.

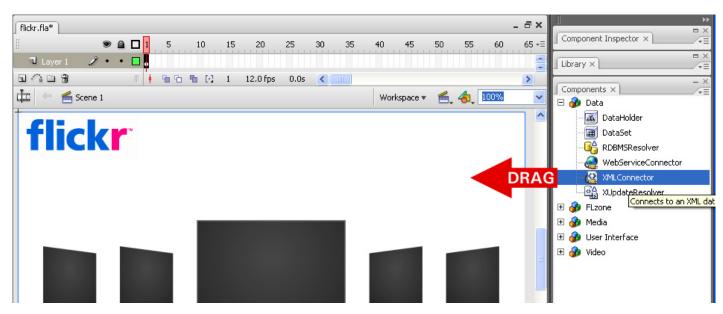


7. Click the Parameters tab and specify the following parameters for the gallery instance: autoFlip -> true

backgroundTransparent -> true descriptionColor -> #0063DC descriptionSize -> 16 imageHeight -> 250 imageOffset -> 120 imageSpaceBack -> 103 imageSpaceMain -> 111 imageWidth -> 250 reflectionSize -> 30

Properties Filters Paramet	ers ×		->
Component	imageSpaceMain	111	<u> </u>
	imageWidth	250	
flickrGallery	motionBlur	false	
	reflections	true	
W: 680.0 X: 0.0	reflectionSize	30	
H: 450.0 Y: 0.0	reflectionStrength	30	~
		i	

8. Drag a XMLConnector component from the components panel to the Stage.





9. Select the **XMLConnector** component on the Stage and go to the parameters tab on the Property inspector. Enter **xmlConnector** in the instance name field.

1	Properties Filters Parameters	×		- × -=
	W: 32.0 X: 580.3	URL direction ignoreWhite multipleSimultaneousAll suppressInvalidCalls	send/receive true true false	() () () () () () () () () () () () () (
	H: 32.0 Y: 45.4	J		

- 10. Enter http://api.flickr.com/services/feeds/photos_public.gne?format=rss2 for the URL property.
- 11. Change the direction parameter to **receive**.
- 12. The gallery doesn't understand the format of the RSS feed, so we need to make a Formatter class to convert the data. Create a new **ActionScript File**.

New Document	
General Templates Type: Image: Templates Image: Templates Type: Image: Templates Type: Templates Image: Templates Templates Image: Templates Te	<u>Description:</u> Create a new external ActionScript file (*.as) and edit it in the Script window. ActionScript is the Flash scripting language and is used to control actions, operators, objects, classes, and other elements in movies and applications. You can use code hinting and other Flash script editing tools to help create scripts. You can reuse external scripts in multiple applications. Set up a Flash Project file to link external scripts with Flash documents and to provide source control.
	OK Annuleren

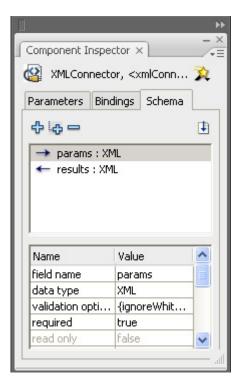


Add the following code:

```
import mx.xpath.XPathAPI;
class MyFormatter extends mx.data.binding.CustomFormatter
{
    // Format a Number, return a String
    function format(rawValue) {
        var returnValue:Array = new Array();
            for (var i:Number = 0; i < rawValue.length; i++)</pre>
            {
                  var item:Object = new Object();
                  var node:XMLNode = rawValue[i];
                  item.description = XPathAPI.selectSingleNode(node,
"item/title").firstChild.nodeValue;
                  item.url = XPathAPI.selectSingleNode(node,
"item/media:thumbnail").attributes.url.split("_s").join("_m");
                  returnValue.push(item);
            }
        return returnValue;
    }
      // convert a formatted value, return a raw value
    function unformat(formattedValue) {
            return formattedValue;
    }
```

- 17. Save the file as MyFormatter.as and save it in the same folder as the fla.
- 18. Go back to the flash document.
- 19. Make sure the XMLConnector component on your Stage is selected.

20. Open the Component Inspector and go to the Schema tab.



- 21. We need the XML Schema of the RSS feed to make our binding.
- 22. Open the <u>http://api.flickr.com/services/feeds/photos_public.gne?format=rss2</u> URL in your browser and save the page as xml.
- 23. Go back to Flash and in the Schema tab of the Component Inspector select results : XML.
- 24. Click the import Schema button (icon of a page with an arrow in it).

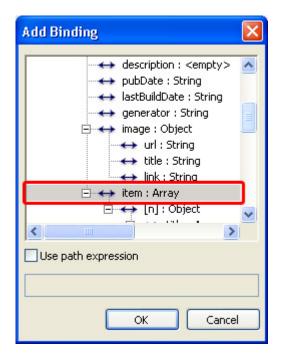
		-
Component Insp	ector ×	
🗳 XMLConnec	tor, <xmlconn< td=""><td>X</td></xmlconn<>	X
Parameters Bir	ndings Schema	
ф i _ф —		Ð
		_
→ params : >	(ML)	
→ params : > ← results : X		
		~
← results : X	ML Value	•
results : X Name field name	ML	
results : X Name field name data type	ML Value results XML	
results : X Name field name	ML Value results XML	



- 25. Select the downloaded xml file and press **Open**.
- 26. Go to the Bindings tab.
- 27. Add a Binding by clicking the plus icon.

Component Inspe	ector ×	1	
🗿 XMLConnect	or, <xr< th=""><th>nlConn</th><th>3</th></xr<>	nlConn	3
Parameters Bin	dings	Schema	
<u></u>			
Name	Value		
direction	Value <emp< td=""><td>ty></td><td></td></emp<>	ty>	
direction	<emp< td=""><td>ty></td><td></td></emp<>	ty>	

28. From the list select results > rss > channel > item: Array and click OK.





29. Back in the Component Inspector make sure that the direction is set to **out**.

Parameters Bindings Schema Parameters Bindings Schema	🗿 XMLCor	nnector, <×	mlConn 🤉	Ŷ
Image: Second state of the second	arameters	Bindings	Schema	
Name Value direction out bound to in formatter out	ф <mark>—</mark>			
direction out bound to in formatter out in(out)	OD res	ults.rss.cha	nnel.item	
bound to in formatter out				
formatter out	Name	Value		
infout			8	•
formatter opti	direction	out in	3	•
	direction bound to	out in out		•

- 30. Click on the **Bound to** parameter.
- 31. Select the ImageFlow3D component and choose the items parameter.

Bound To	
Component path:	Schema location:
Scene 1	 ↔ items : Array ← percentLoaded : Number ↔ selectedIndex : Number
Use path expression	
Use constant value	
	OK Cancel



32. As formatter choose Custom Formatter.

C			
Component Inspector ×			
🗿 XMLConnector, <xmlconn td="" 💢<=""></xmlconn>			
Parameters Bi	ndings Schema		
Ф 🗖			
🔘 results.	.rss.channel.item		
Name	Value		
Name direction	Value		
direction	out		
direction bound to	out flickrGallerv:items Custom Formatter		

- 33. Enter **MyFormatter** in the formatter options field to link it to the ActionScript File with the **MyFormatter** class we created earlier.
- 34. Add the following code to Frame 1 of your movie:

```
// This code loads cross-domain policy files from flickr.com . Flash Player uses
policy files as a permission mechanism to permit SWF files to load data from
servers other than their own.
System.security.loadPolicyFile("http://static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm1.static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm2.static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm3.static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm4.static.flickr.com/crossdomain.xml");
Make the gallery active
flickrGallery.setFocus();
```

35. Open the Publish Settings and go to the Flash tab.



36. In the Local playback security dropdown select Access network only to enable local testing.

Publish Settings	×
Current profile: Default Formats Flash HTML	▼ ⊕, + ⊕ 0 0
<u>V</u> ersion:	Flash Player 9 🔽 Info
Load order:	Bottom up 🗸
<u>A</u> ctionScript version:	ActionScript 2.0 🗸 Settings
Options:	Generate size report Protect from import Omit trace actions Permit debugging Compress movie Optimize for Flash Player 6 r65 Export hidden layers Export SWC
Password:	
Script time limit:	15 seconds
JPEG guality:	0 100
Audio <u>s</u> tream:	MP3, 16 kbps, Mono Set
Audio <u>e</u> vent:	MP3, 16 kbps, Mono Set
Local playback security:	Override sound settings Export device sounds Access network only
	Publish OK Cancel

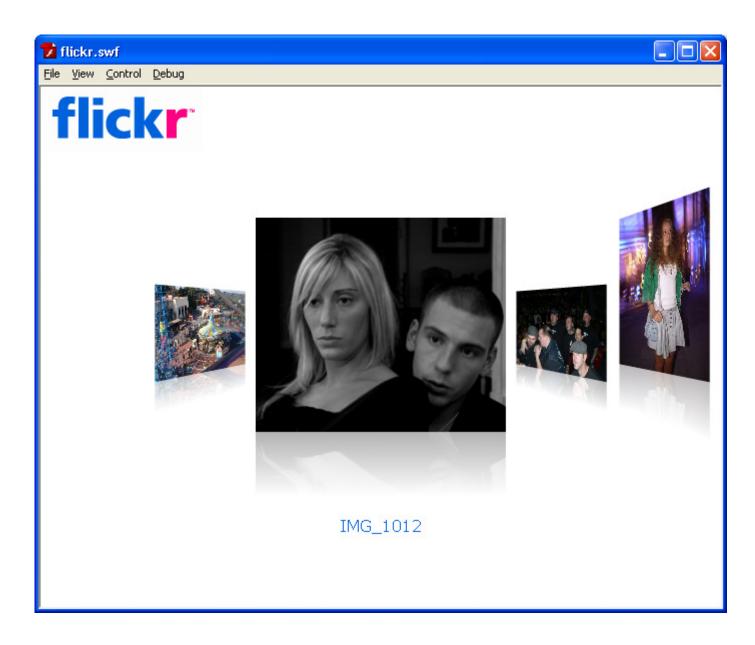
- 37. Click the **Settings...** button next to the ActionScript version.
- 38. Click the plus button and add . as Classpath.



ActionScript 2.0 Settings	
Export frame for classes: 1	+ - +
	OK Cancel

39. Save and test your Movie. You should see a gallery that loads its photo's from Flickr.

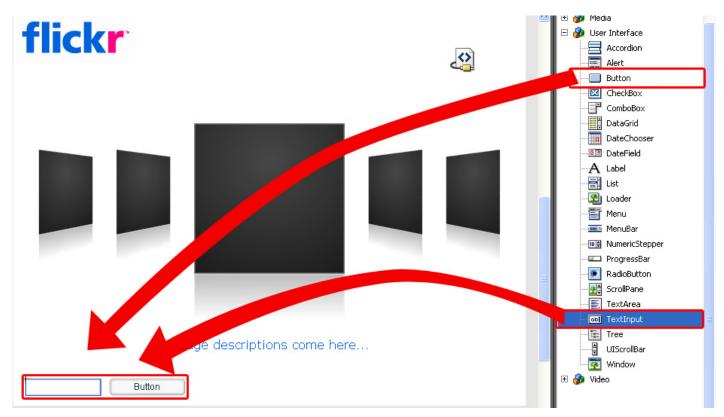






Adding a Search Option

1. Next we are going to add a search option. Drag a **TextInput** and a **Button** component from the Components panel to the Stage. We place them at the bottom left of our movie.



2. Give the TextField component the instance name **searchText** and enter **searchButton** in the Button instance name field. We'll also give the Button a label with the value **Search**.

Properties Filters Parameters ×			- × •=
Component	licon		· · · · · · · · · · · · · · · · · · ·
searchButton	label	Search	0
Bearchbacton	labelPlacement	right	
W: 100.0 X: 128.3	selected	false	
	toggle	false	•
н: 22.0 y; 484.3			
	1		





3. To make them work replace the code of Frame 1 with the following code:

```
// This code loads cross-domain policy files from flickr.com . Flash Player uses
policy files as a permission mechanism to permit SWF files to load data from
servers other than their own.
System.security.loadPolicyFile("http://static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm1.static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm2.static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm3.static.flickr.com/crossdomain.xml");
System.security.loadPolicyFile("http://farm4.static.flickr.com/crossdomain.xml");
// Add listeners for the textfield and the search button, when they are triggered
the search() function is executed
var listener:Object = new Object();
listener.enter = search;
listener.click = search;
searchText.addEventListener("enter", listener);
searchButton.addEventListener("click", listener);
function search() {
    // We set a new URL with our search string for the XMLConnector
   xmlConnector.URL =
"http://api.flickr.com/services/feeds/photos_public.gne?format=rss2&tags=" +
searchText.text;
    // We trigger the XMLConnector to get the new XML from flickr that is
customized by your search string
   xmlConnector.trigger();
    // We activate the gallery again as the focus was set to the text field or
the button
   flickrGallery.setFocus();
}
// get the XML from flickr
xmlConnector.trigger();
// Make the gallery active
flickrGallery.setFocus();
```

4. Now save the document and Publish your movie.

Used resources

Flickr logo (logo_home.png)

