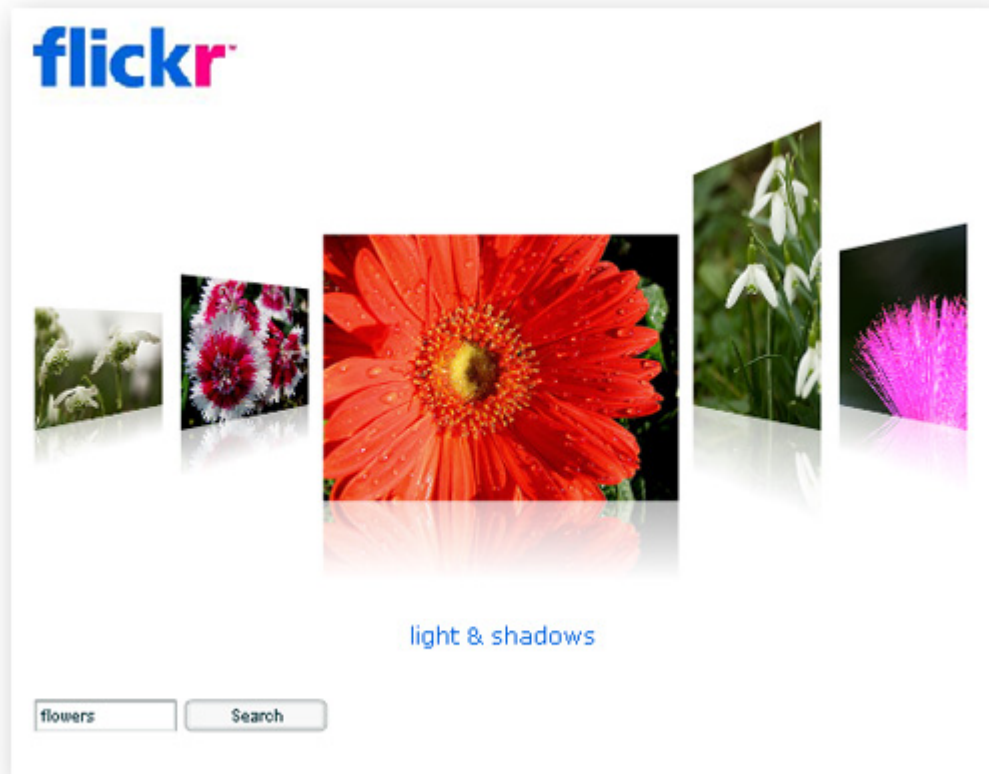


## Creating a Gallery that gets it's images dynamically from Flickr

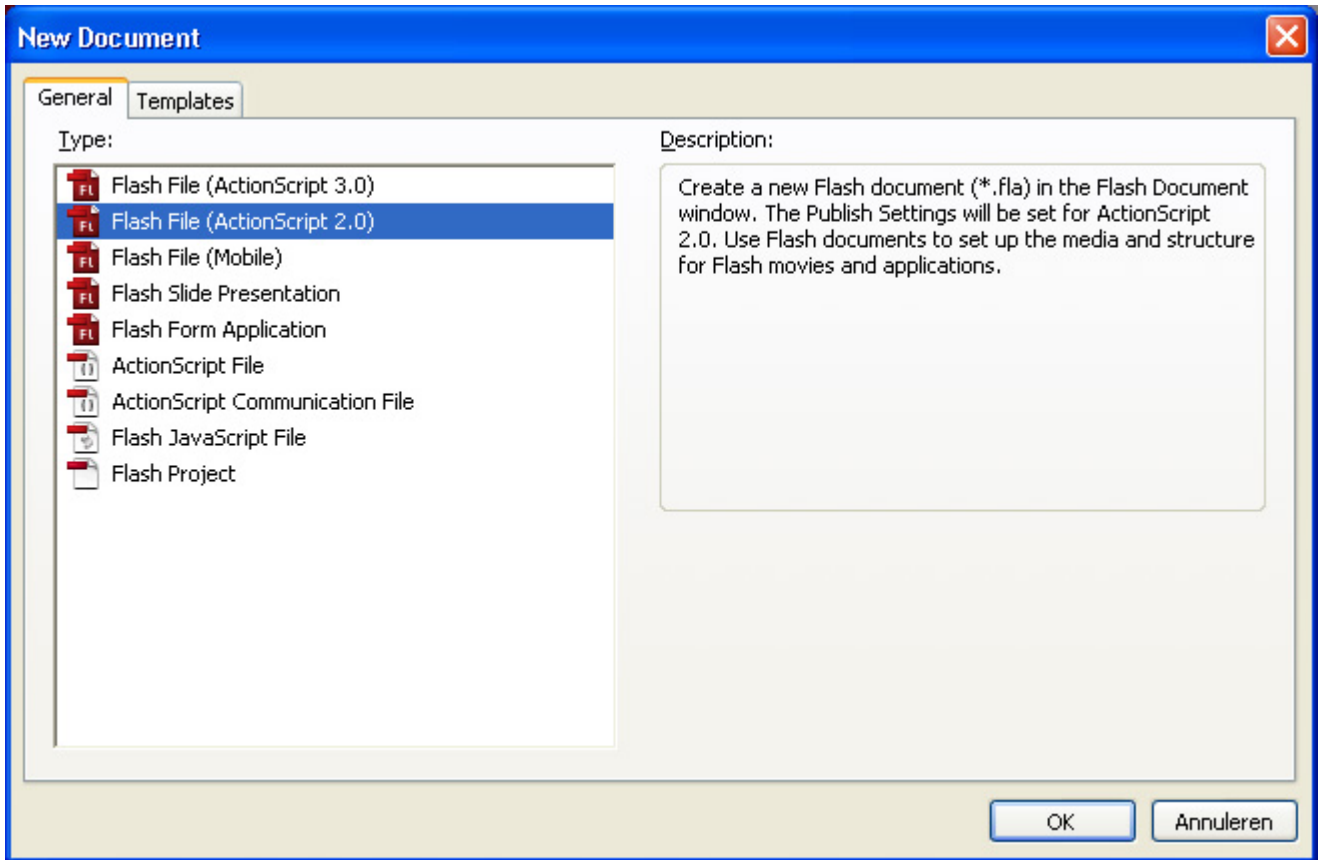
### Intro

In this tutorial we'll show you how easy it is to create a dynamic photo album with [3D ImageFlow Gallery For Flash](#) 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.

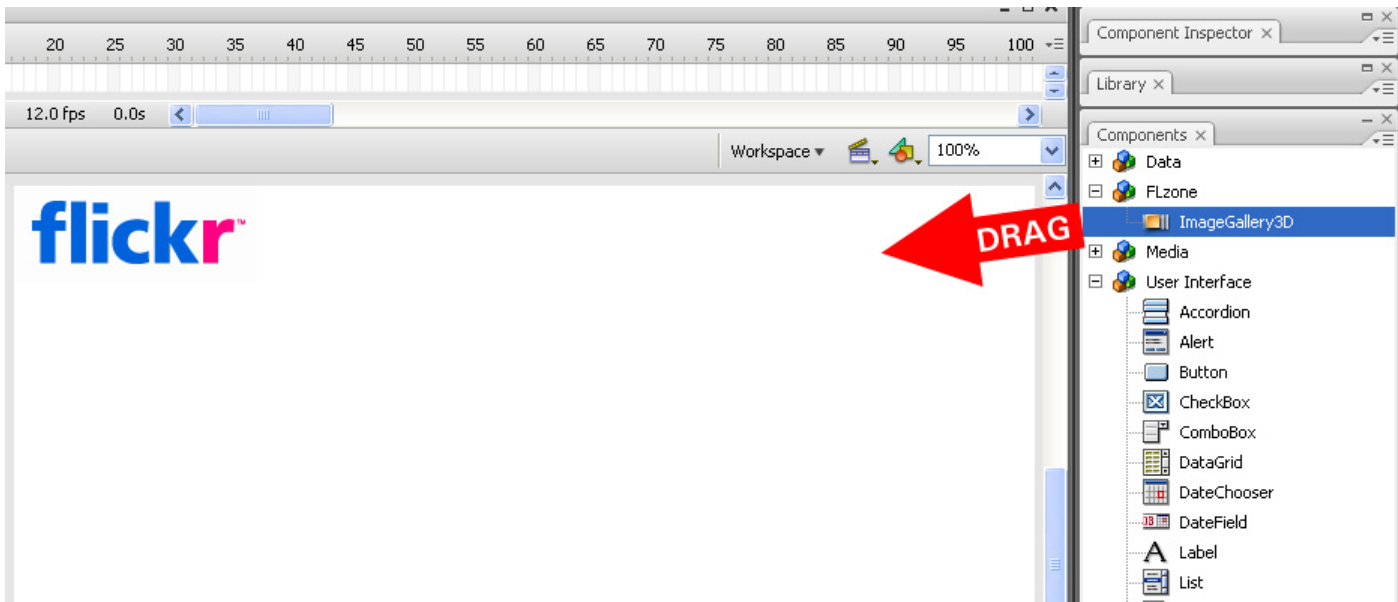


## Building the Gallery

1. Install the [3D ImageFlow Gallery For Flash](#) extension and create a **new Flash document** (ActionScript 2.0) and **save the page**.

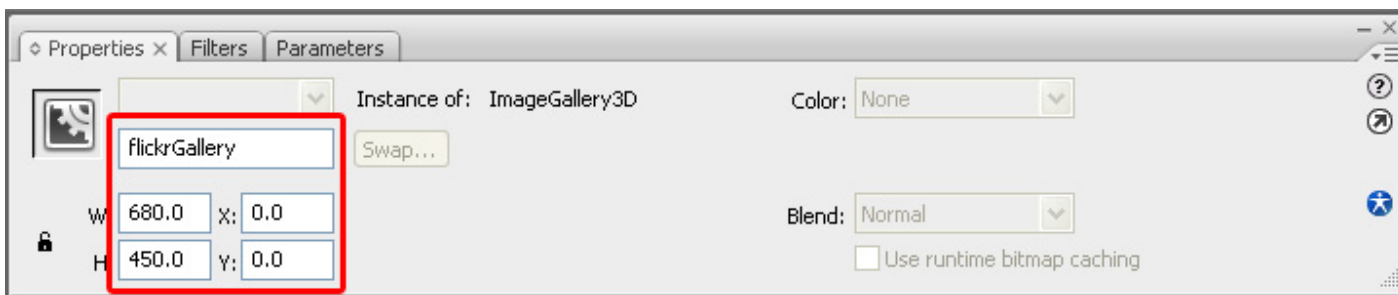


2. Set the stage size to the desired width and height, we use a stage size of **680 x 520**
3. Select File > Import > Import to Stage.
4. Select the flickr logo (logo\_home.png).
5. Drag the **ImageGallery3D component** from the components panel to the Stage.

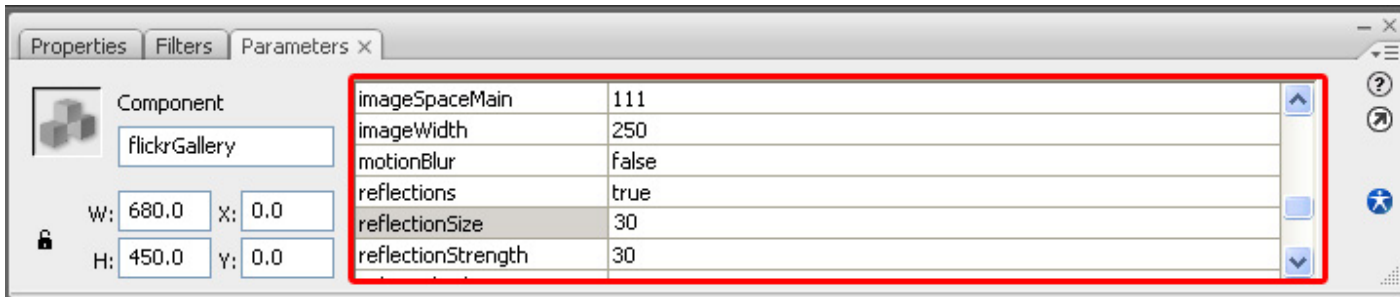


Tip: Instead of dragging the component, you can also Double-click the ImageGallery3D component.

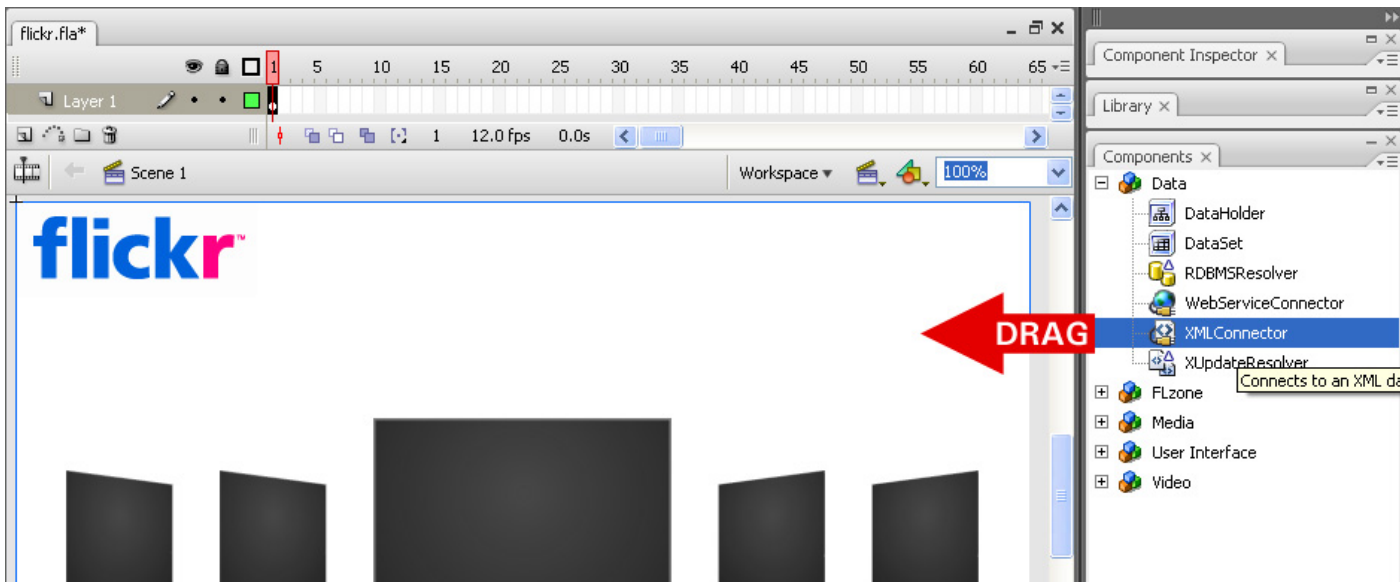
6. Select the ImageGallery3D component on the Stage. In the Property inspector, enter the instance name **flickrGallery** and set **W** to **680**, **H** to **450**, **X** to **0** and **Y** to **0**.



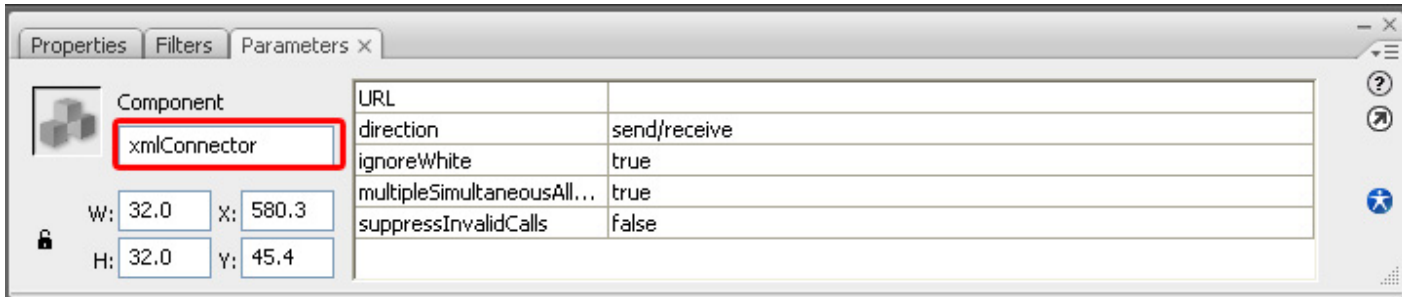
- Click the Parameters tab and specify the following parameters for the gallery instance:
  - autoFlip** -> true
  - backgroundTransparent** -> true
  - descriptionColor** -> #0063DC
  - descriptionSize** -> 16
  - imageHeight** -> 250
  - imageOffset** -> 120
  - imageSpaceBack** -> 103
  - imageSpaceMain** -> 111
  - imageWidth** -> 250
  - reflectionSize** -> 30
  - reflectionStrength** -> 30



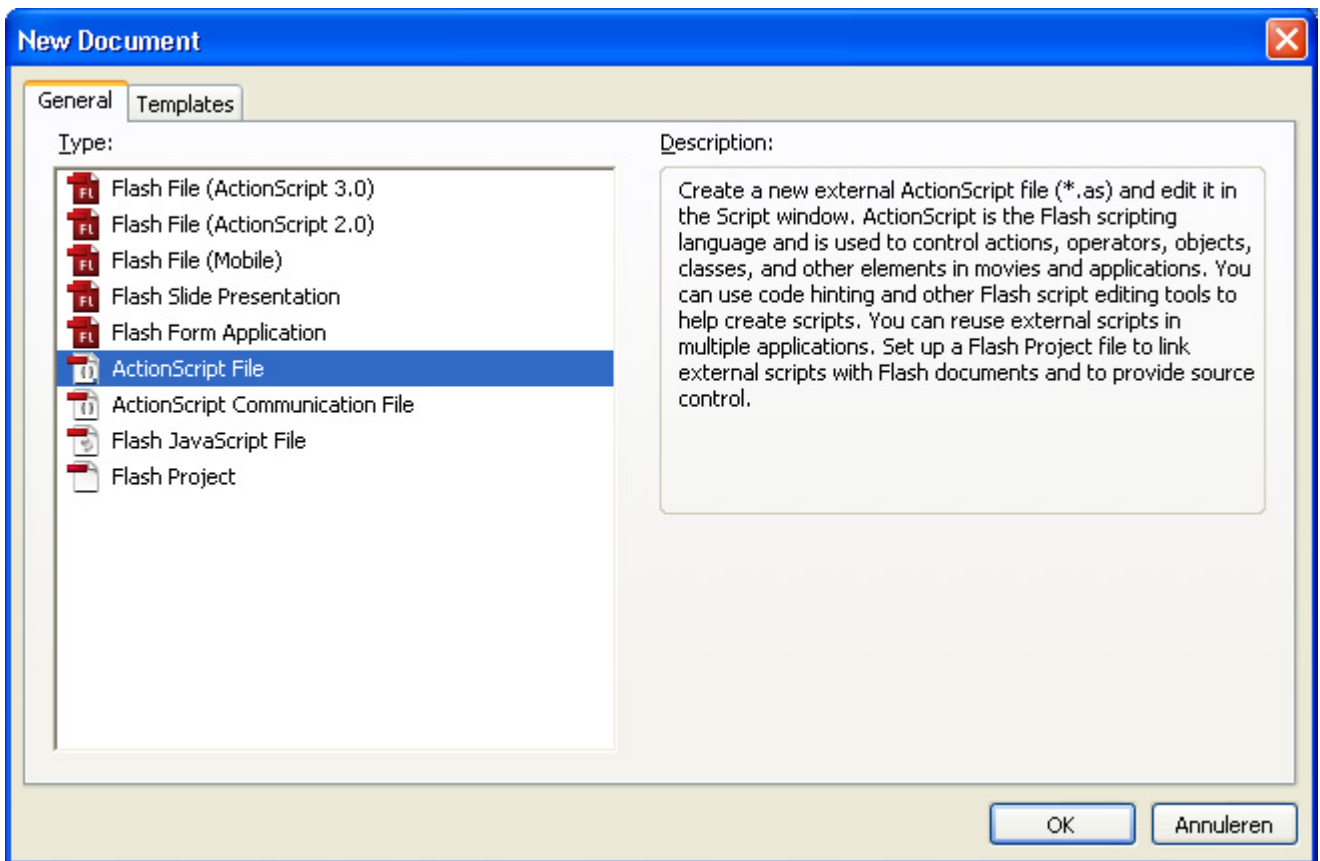
- Drag a **XMLConnector** component from the components panel to the Stage.



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



- Enter **[http://api.flickr.com/services/feeds/photos\\_public.gne?format=rss2](http://api.flickr.com/services/feeds/photos_public.gne?format=rss2)** for the URL property.
- Change the direction parameter to **receive**.
- 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**.



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();

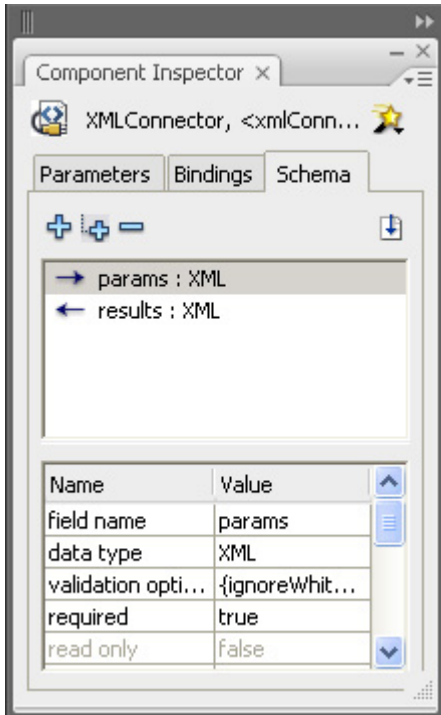
        for (var i:Number = 0; i < rawValue.length; i++)
        {
            var item:Object = new Object();
            var node:XMLNode = rawValue[i];
            item.description = XPathAPI.selectSingleNode(node,
"item/title").firstChild.nodeValue;
            item.url = XPathAPI.selectSingleNode(node,
"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.

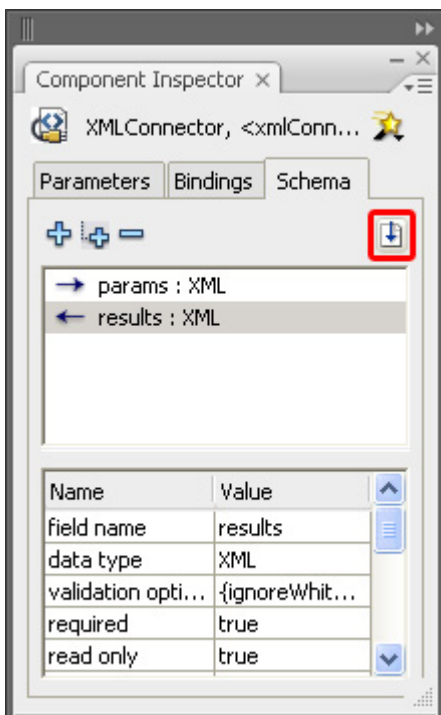


21. We need the XML Schema of the RSS feed to make our binding.

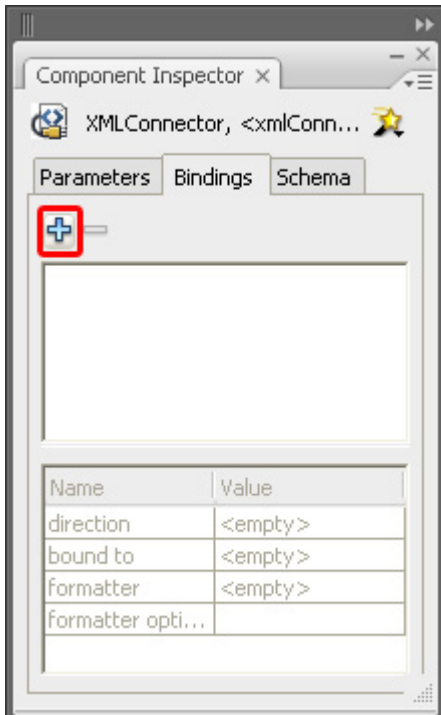
22. Open the [http://api.flickr.com/services/feeds/photos\\_public.gne?format=rss2](http://api.flickr.com/services/feeds/photos_public.gne?format=rss2) URL in your browser and save the page as xml.

23. Go back to Flash and in the Schema tab of the Component Inspector select **results : XML**.

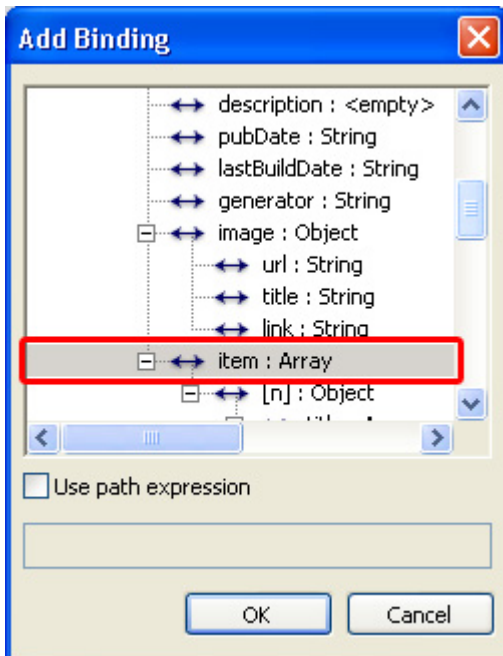
24. Click the import Schema button (icon of a page with an arrow in it).



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

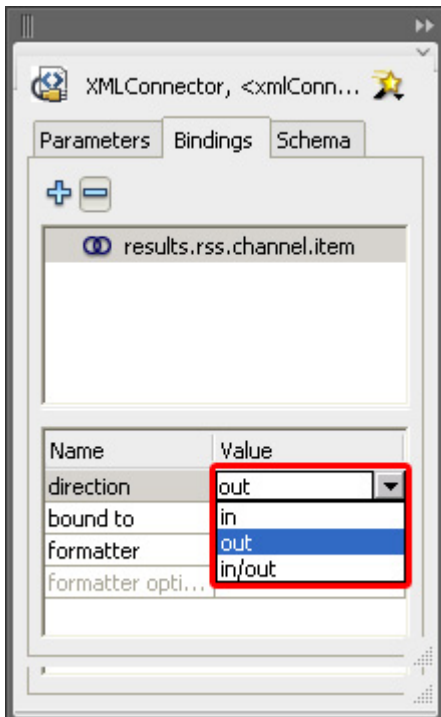


28. From the list select **results > rss > channel > item: Array** and click **OK**.



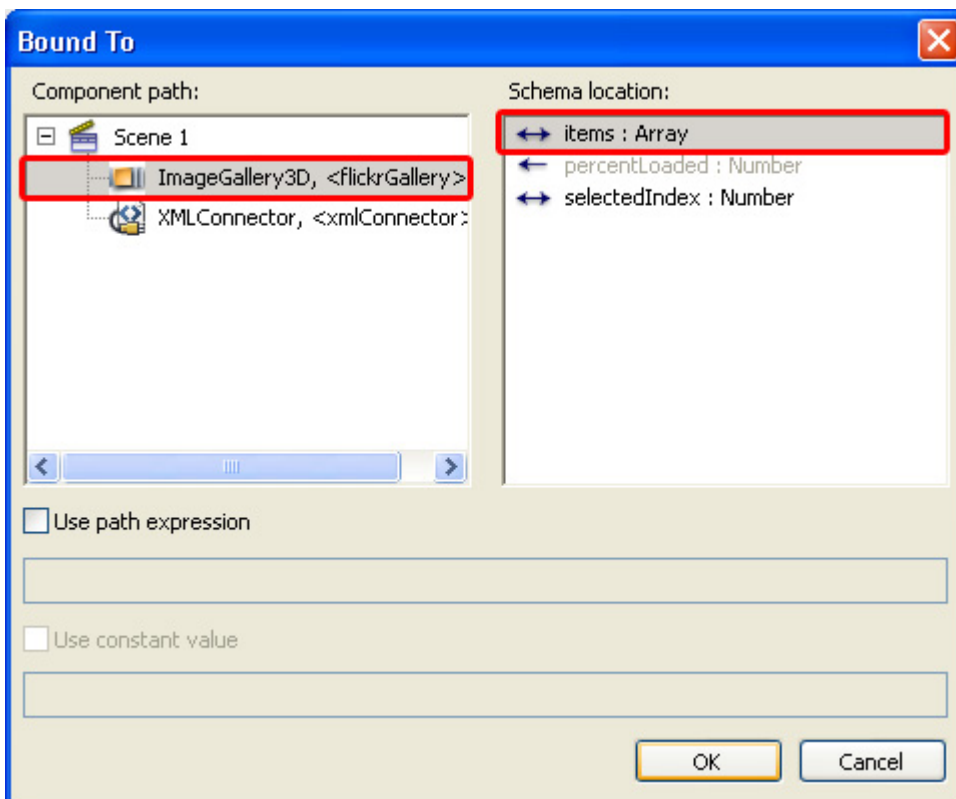


29. Back in the Component Inspector make sure that the direction is set to **out**.

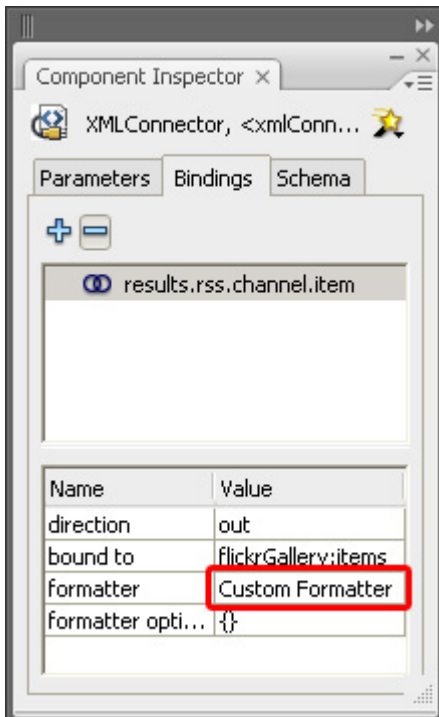


30. Click on the **Bound to** parameter.

31. Select the **ImageFlow3D component** and choose the **items** parameter.



32. As formatter choose **Custom Formatter**.



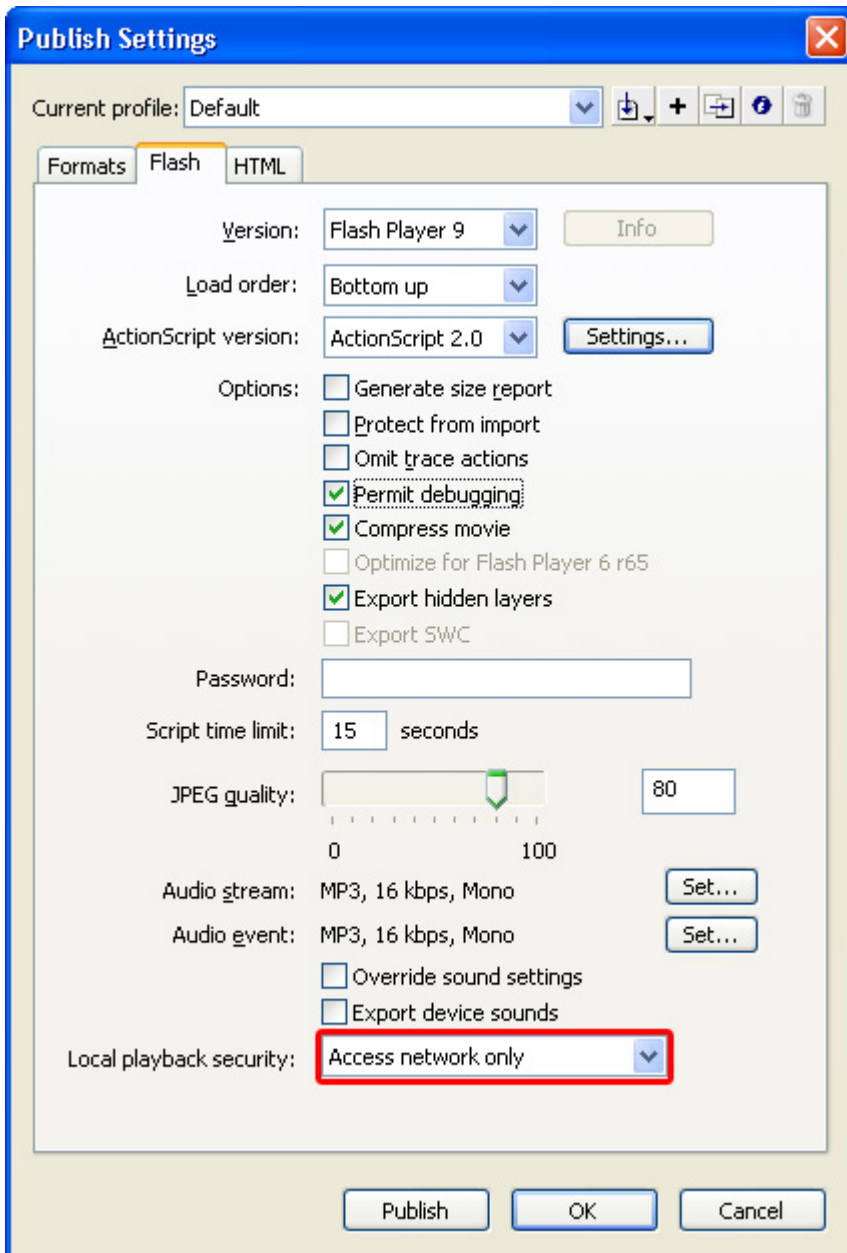
33. Enter **MyFormatter** in the formatter options field to link it to the ActionScript File with the **MyFormatter** class we created earlier.

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");
// get the XML from flickr
xmlConnector.trigger();
// Make the gallery active
flickrGallery.setFocus();
```

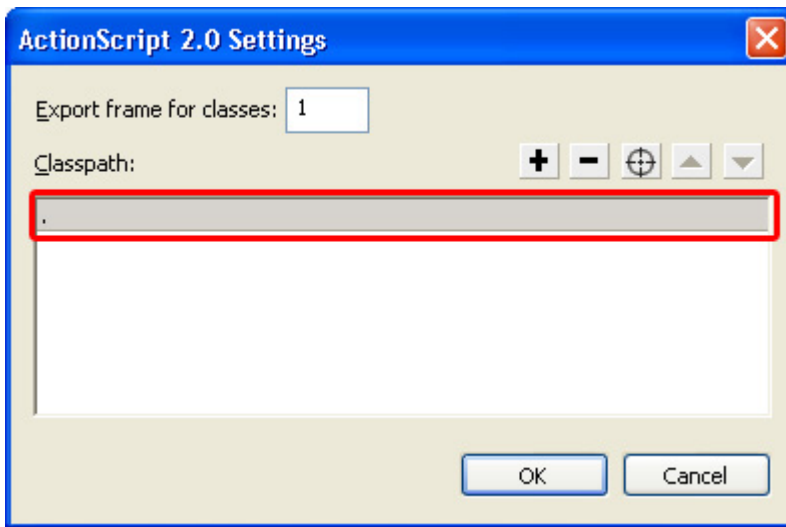
35. Open the Publish Settings and go to the Flash tab.

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

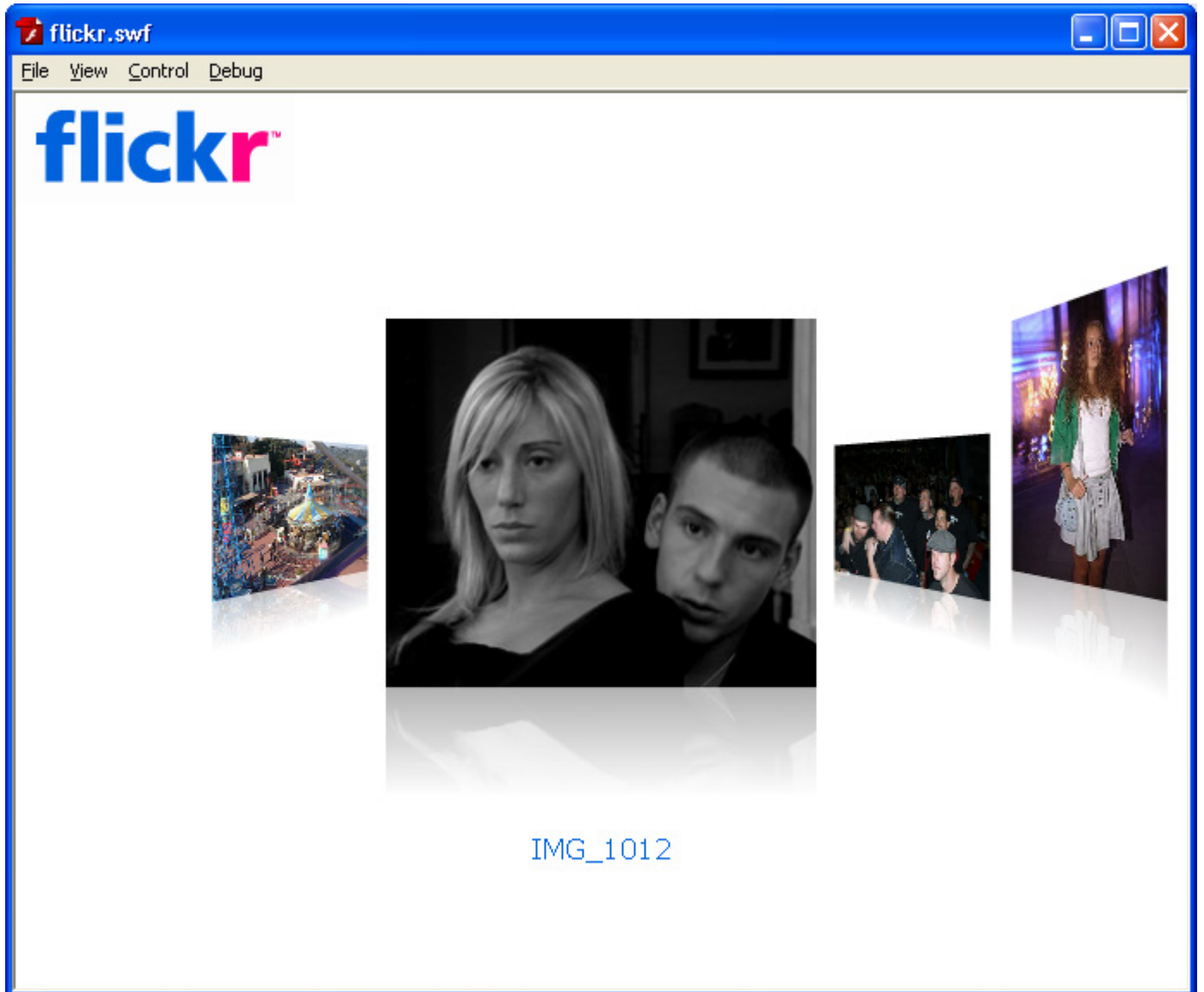


37. Click the **Settings...** button next to the ActionScript version.

38. Click the plus button and add . as Classpath.

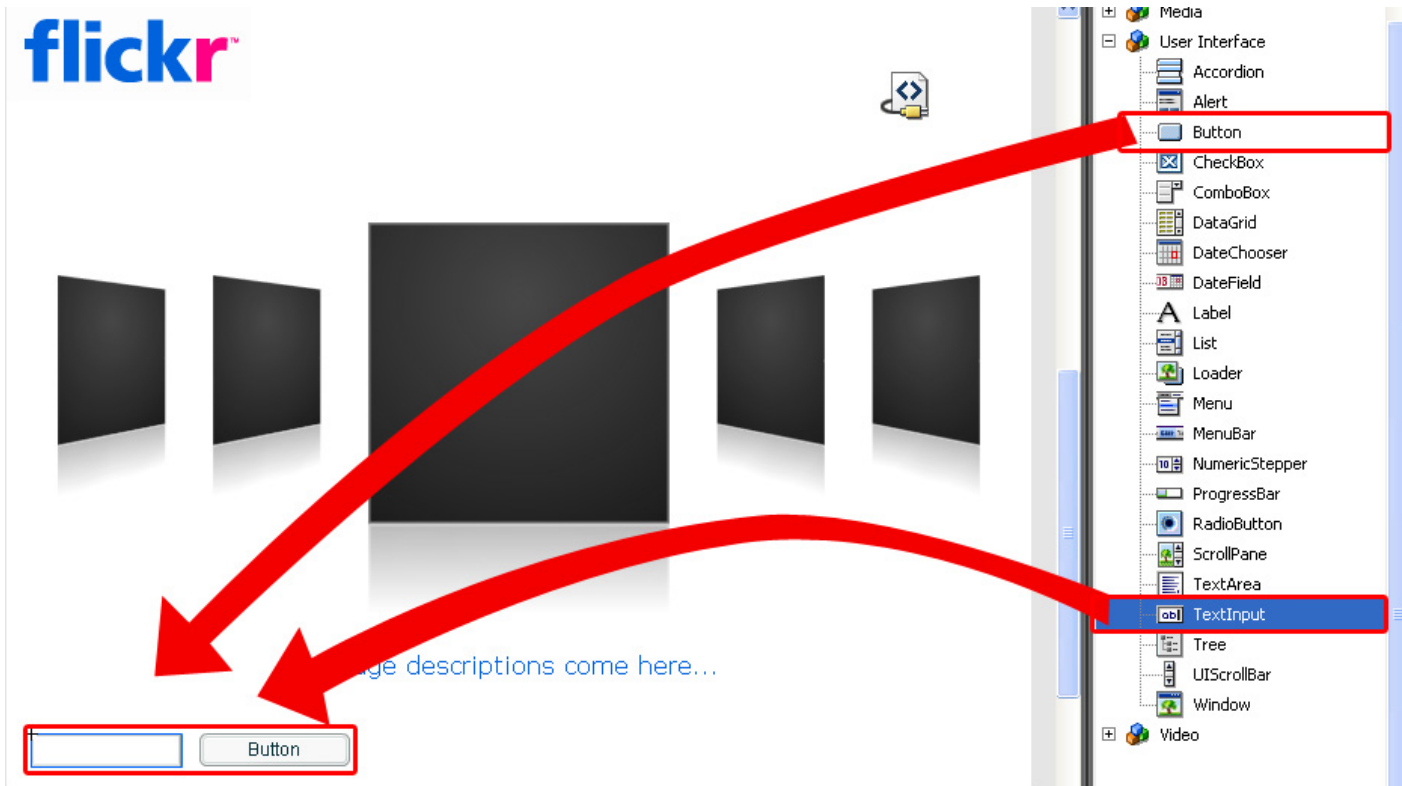


39. Save and test your Movie. You should see a gallery that loads its photo's from Flickr.

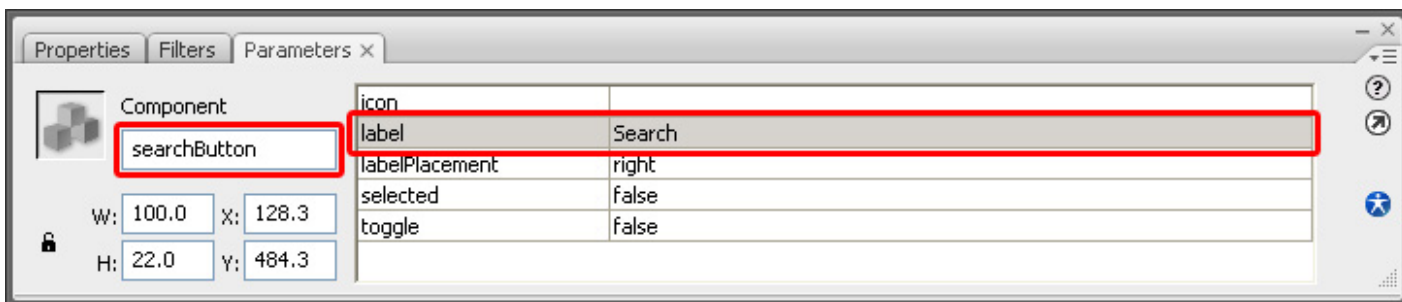


## Adding a Search Option

- 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.



- 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**.



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)

