

FLICKR HACKS™

*Tips & Tools for Sharing
Photos Online*



O'REILLY®

*Paul Bausch & Jim Bumgardner
With a foreword by Caterina Fake, cofounder of Flickr*

Flicker Hacks™

by Paul Bausch and Jim Bumgardner

Copyright © 2006 O'Reilly Media, Inc. All rights reserved.
Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North,
Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (safari.oreilly.com). For more information, contact our corporate/institutional sales department: (800) 998-9938 or corporate@oreilly.com.

Editor: Brian Sawyer

Production Editor: Darren Kelly

Copyeditor: Rachel Wheeler

Proofreader: Sada Preisch

Indexer: Ellen Troutman Zaig

Cover Designer: Michael Kohnke

Interior Designer: David Futato

Illustrators: Robert Romano, Jessamyn
Read, and Lesley Borash

Printing History:

February 2006: First Edition.

Nutshell Handbook, the Nutshell Handbook logo, and the O'Reilly logo are registered trademarks of O'Reilly Media, Inc. The *Hacks* series designations, *Flicker Hacks*, the image of a slide projector, and related trade dress are trademarks of O'Reilly Media, Inc.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and O'Reilly Media, Inc. was aware of a trademark claim, the designations have been printed in caps or initial caps.

While every precaution has been taken in the preparation of this book, the publisher and authors assume no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

Small print: The technologies discussed in this publication, the limitations on these technologies that technology and content owners seek to impose, and the laws actually limiting the use of these technologies are constantly changing. Thus, some of the hacks described in this publication may not work, may cause unintended harm to systems on which they are used, or may not be consistent with applicable user agreements. Your use of these hacks is at your own risk, and O'Reilly Media, Inc. disclaims responsibility for any damage or expense resulting from their use. In any event, you should take care that your use of these hacks does not violate any applicable laws, including copyright laws.



This book uses RepKover™, a durable and flexible lay-flat binding.

ISBN: 0-596-10245-3

[M]

This excerpt is protected by copyright law. It's your responsibility to obtain permissions necessary for any proposed use of this material. Please direct your inquiries to permissions@oreilly.com.

HACK
#50

Make a Slideshow

PHP and Flash combine to create eye candy with extreme flexibility.

In “Authenticate Users” [Hack #41], we built an app that displays a grid of images. Wouldn’t it be nice to get those same images to appear in an animated slideshow? That’s what this hack is about.

What You Need

This hack uses a modified version of the script developed in [Hack #41], so you’ll need to have successfully completed that hack before starting this one. You’ll also need a web server with PHP 4 support.

In this hack, we’ll be making a Flash movie that is compatible with Flash 7. You’ll need a version of Macromedia Flash that is at least as recent as Flash MX 2004.

The Flash Movie

If you’re a programmer who has never learned Flash, you’ll like this hack, because the movie requires virtually no authoring using the Flash UI (which is artist-friendly but hacker-unfriendly). Instead, the effects in the movie are accomplished exclusively via ActionScript, Flash’s scripting language.

Create a text file called *slideshow_script.as* and add the following ActionScript code:

```
// Basic Looping Slideshow for Flash 7
//
// - Jim Bumgardner
//
// Parameters:
// flickrSnaps - list of server/id/secret triplets (delimited by commas)
// timeBetweenSlides (milliseconds)
// fadeTime (milliseconds)
// debug (adds status msgs)

flickrSnapAry = _root.flickrSnaps.split(',');

// Create containers/loaders here...
containers = [];
curSlide = 0;
curSlot = 0;
if (!timeBetweenSlides)
    timeBetweenSlides = 5000;
if (!fadeTime)
    fadeTime = 3000;

doNextSlide = function()
{
```

```

clearInterval(this.inth);
curSlide = (curSlide + 1) % flickrSnapAry.length;
this.altSlot.doLoad(curSlide);
}

fadeInStep = function()
{
    var r = (getTimer() - this.fxStart)/this.fxDuration;
    if (r >= 1) {
        r = 1;
        clearInterval(this.inth);
        this.inth = setInterval(this, "doNextSlide", timeBetweenSlides);
    }
    // This applies an ease in/out to the fade effect
    r = r*r*(3-2*r);
    this._alpha = r*100;
    this.altSlot._alpha = (1-r)*100;
    // This next line provides the barn-door effect
    // Comment it out for a more standard-looking
    // cross-fade effect
    this._xscale = r*100;
}

doFadeIn = function(duration)
{
    this.fxStart = getTimer();
    this.fxDuration = duration;
    this.fxStep = fadeInStep;
    this.inth = setInterval(this, "fxStep", 1000/30);
}

handleLoadInit = function(target)
{
    // Center the slide
    this._x = (Stage.width - target._width)/2;
    this._y = (Stage.height - target._height)/2;
    this.doFadeIn(fadeTime);
}

handleLoadError = function(target, errorCode)
{
    _root.errMsg.text = 'got load error ' + errorCode + ' on image ' +
        this.itsURL;
}

doLoad = function(slideNbr)
{
    if (_root.debug)
        _root.errMsg.text = "loading " + slideNbr + ' ' +
flickrSnapAry[slideNbr];
    var fVars = flickrSnapAry[slideNbr].split('/');
    this._alpha = 0;
    this.itsURL = 'http://static.flickr.com/' + fVars[0] + '/' +

```

```

        fVars[1] + '_' + fVars[2] + '.jpg';
        this.itsLoader.loadClip(this.itsURL, this.loadee);
    }

    // Initialize two movieclips for slideshow
    for (var i = 0; i < 2; ++i)
    {
        var mc = _root.createEmptyMovieClip('ctr_' + i, i + 1);
        mc.createEmptyMovieClip('loadee', 1);
        mc.idx = i;
        mc.doLoad = doLoad;
        mc.doFadeIn = doFadeIn;
        mc.doNextSlide = doNextSlide;
        mc.onLoadInit = handleLoadInit;
        mc.onLoadError = handleLoadError;
        var loader = new MovieClipLoader();
        loader.addListener(mc);
        mc.itsLoader = loader;
        containers[i] = mc;
    }
    containers[0].altSlot = containers[1];
    containers[1].altSlot = containers[0];

    _root.createTextField('errMsg', 10, 10, 10, 480, 50);
    _root.errMsg.textColor = 0xFFFFFFFF;

    // Load the first slide
    if (flickrSnapAry.length <= 0)
        _root.errMsg.text = "No snaps specified";
    else
        containers[curSlot].doLoad(curSlide);

```

Then, run Flash and create a new project. Use the Modify → Document menu and specify the following settings for the movie:

Dimensions

500 × 500 pixels

Background Color

Black

Frame Rate

30 fps

Select the first frame of the movie on the timeline, and then open the ActionScript editing window by selecting Window → Development Panels → Actions from the menu.

In the ActionScript window, enter the following one-line script, which causes the script you just created to be included:

```
#include "slideshow_script.as"
```

That's about all the Flash authoring you need to do. As you can see, most of the interesting stuff is in the ActionScript file.



If you like Flash's text editor, feel free to put the complete script inside the ActionScript window, rather than using an include file. I prefer to keep my scripts separate, so I can use a more full-featured text editor.

Save the project as *slideshow fla*, and then select File → Publish. This will produce a Flash movie file called *slideshow swf*, which should be approximately 1 KB in size. This is one of the nice things about making ActionScript-centric Flash movies: they tend to be small and don't require annoying preloaders.

Upload *slideshow swf* to your web server.

The Flash movie does not talk to the Flickr API directly. Instead, it accepts a list of Flickr photos as one of its parameters. The list uses two kinds of delimiters. Each individual photo is specified by three strings—the server number, the photo ID, and the photo secret—which are delimited using slashes. Each of these triplets is separated by a comma. Thus, a list of two photos looks like this:

```
29/44754794/29dbf5c75a,29/47126414/42ce67e060
```

The Flash movie takes this information and uses it to construct URLs that are used to pull the photos off of Flickr. The PHP web page that loads the Flash movie will build this list of photos by querying the Flickr API for information about your photostream.

The Code

The PHP script is a modified version of the script developed in “Authenticate Users” [Hack #41].

Create a new text file called *testSlideshow.php* and enter the following PHP code:

```
<?php
ini_set("error_reporting ", E_ALL);

// Configuration stuff
$flickrAPIKey = "your API key goes here";
$flickrSharedSecret = "your shared secret goes here";

// Caching configuration stuff
$cachingenabled = false;
$dbUser = "your database username";
$dbPass = "your database password";
```

```
$dbAddress = "your database domain name (or localhost)";
$dbTable = "database table name";

// Create new phpFlickr object
require_once("phpFlickr.php");
$f = new phpFlickr($flickrAPIKey,$flickrSharedSecret);
if ($cachingenabled == true)
{
    $f->enableCache(
        "db",
        "mysql://$dbUser:$dbPass@$dbAddress/$dbTable");
}
$f->auth();
$token = $f->auth_checkToken();

// Parse parameters
$page = 1;
if (isset($_GET['page']))
    $page = $_GET['page'];

$per_page = 30;
if (isset($_GET['per_page']))
    $per_page = $_GET['per_page'];

$fade_time = 2000;
if (isset($_GET['fade_time']))
    $fade_time = $_GET['fade_time'];

// Find the NSID of the username inputted via the form
$nsid = $token['user']['nsid'];

// Get the friendly URL of the user's photos
$photos_url = $f->urls_getUserPhotos($nsid);

// Get a set of the user's photos
$photos = $f->photos_search(array("user_id" => $nsid, "per_page" =>
$per_page, "page" => $page));

$flickr_snaps = '';
$i = 0;
foreach ($photos['photo'] as $photo) {
    if ($i++)
        $flickr_snaps .= ',';
    $flickr_snaps .= $photo['server'] . '/' . $photo['id'] . '/' .
    $photo['secret'];
}

print <<<EOT
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.
w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
```

```
<title>slideshow</title>
</head>
<body bgcolor="#000000" color=white><?php
<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
  codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/
  swflash.cab#version=7,0,0,0"
  width="500" height="500" id="slideshow" align="middle">
<param name="allowScriptAccess" value="sameDomain" />
<param name="movie" value="slideshow.swf?fadeTime=$fade_
  time&flickrSnaps=$flickr_snaps" />
<param name="quality" value="high" />
<param name="bgcolor" value="#000000" />
<embed src="slideshow.swf?fadeTime=$fade_time&flickrSnaps=$flickr_snaps"
  quality="high" bgcolor="#000000" width="500" height="500" name="slideshow"
  align="middle" allowScriptAccess="sameDomain"
  type="application/x-shockwave-flash"
  pluginspage="http://www.macromedia.com/go/getflashplayer" />
</object>
</body>
</html>
EOT;
?>
```

You'll need to modify the configuration information at the top, providing your API key, shared secret, and database information if you want caching. For more information about this, see "Authenticate Users" [Hack #41].

Upload *testSlideshow.php* to your web server, into the same location where you uploaded the *slideshow.swf* file.

Running the Hack

To run the hack, enter the address of *testSlideshow.php* into your web browser's address bar:

```
http://www.yourdomain.com/testSlideshow.php
```

If you have not yet been authenticated, you'll see the Flickr permissions screen. Once you've granted permission, the slideshow should begin.

The slideshow provides a transition effect between photos by using both an alpha cross-fade and a scaling effect that makes each new photo appear to move like a closing door.

You can modify the slideshow by passing any of the following parameters to the app:

per_page

This controls how many photos are used in the slideshow before it loops.

page

This controls which page of photos from your photostream is loaded.

`fade_time`

This specifies, in milliseconds, how long the cross-fade effect lasts. This particular parameter gets passed into the Flash movie.

Here's an example URL that passes these parameters to the PHP script:

```
http://www.yourdomain.com/testSlideshow.php?per_page=10&page=4&fade_time=1000
```

Hacking the Hack

To keep things simple, I've written this script to pass only one adjustable parameter, `fade_time`, to the Flash movie, but I would suggest adding a whole slew of parameters so you can customize the slideshow to your liking.

If you want to pass additional parameters to the Flash movie, they can be accessed from within the ActionScript as predefined variables (which are attached to the `_root` timeline). Follow the PHP variable `fade_time`, which gets converted to the ActionScript variable `fadeTime`, to see how this works.



Because of differences between web browsers, parameters to the Flash movie need to be specified in two different locations within the `<object>` tag.

I would suggest adding additional transition effects to the slideshow ActionScript and adding additional parameters to select and control these effects.

A great number of web developers never get past learning one language or technology, but you'll find that combining multiple languages and technologies, as we have done here, can greatly enrich the kinds of end-user experience you can provide.

See Also

- “Authenticate Users” [Hack #41]
- “Make a Color Picker” [Hack #45]