

# ObjectiveFlickr

Integrating a web service to your Cocoa app

Lukhnos D. Liu

2009-05-14



# Outline

- ✦ What is ObjectiveFlickr
- ✦ Who uses ObjectiveFlickr, and why
- ✦ Embedding the framework in your project
- ✦ API workflows
- ✦ Sample apps

[self introduction];

- ✦ Taiwanese Mac and iPhone software developer
- ✦ OpenVanilla
  - ✦ Brought good Cangjie (倉頡) input method to OS X
- ✦ Principal at Lithoglyph Inc.
  - ✦ TapExpense
  - ✦ Mondrianum
  - ✦ Clients include Yahoo! Taiwan and KKBOX

What, Who, Why and Where

# What is ObjectiveFlickr?

- ✦ An Objective-C framework for accessing Flickr API
  - ✦ URL (auth and static data) formation
  - ✦ REST Method invocation (GET/POST)
  - ✦ Uploading
  - ✦ Response block mapping (XML → NSDictionary)
- ✦ Builds on *all* major Apple platforms
  - ✦ Includes [REDACTED] and [REDACTED]!

# Who uses ObjectiveFlickr?

- ✦ Posterino
- ✦ HoudahGeo
- ✦ PictoDrop
- ✦ Screensavr
- ✦ FlickrBooth
- ✦ Skitch

# Why use ObjectiveFlickr?

- ✦ Flickr API has good coverage, but API signing is a hassle
- ✦ Cannot just call `-[NSData dataWithContentsOfURL:]`
  - ✦ Cocoa apps are *asynchronous* in nature

# Where do I get a copy?

- ✦ We love github!
- ✦ <http://github.com/lukhnos/objectiveflickr>
- ✦ Readme file covers most of the things you need to know

# Using ObjectiveFlickr

# Basic Ideas

- ✦ Embedding ObjectiveFlickr in your app
- ✦ Apply for a Flickr API key
- ✦ Create Flickr API context (OFFlickrAPIContext)
- ✦ Create API request object (OFFlickrAPIRequest)
- ✦ Make request
- ✦ Pick up the response

# Embed OF in your app

- ✦ Check out from github:  
<http://github.com/lukhnos/objectiveflickr>
- ✦ Use Xcode cross-project reference to link against it
- ✦ Mac project: Drag in, voilà
- ✦ iPhone project: Some manual interventions
  - ✦ Need also set include headers
- ✦ See sample app projects if you are lost

# Getting API key

- ✦ You need to apply for a Flickr API key first  
<http://www.flickr.com/services/api/keys/>
- ✦ Know your “key” and “shared secret”

# Create API context object

```
• OFFlickrAPIContext *context = [[OFFlickrAPIContext  
alloc] initWithAPIKey:YOUR_KEY  
sharedSecret:YOUR_SHARED_SECRET];
```

# Make requests

- `OFFlickrAPIRequest *request = [[OFFlickrAPIRequest alloc] initWithAPIContext:context];`
- `[request setDelegate:self];`
- `[request callAPIMethodWithGET:@"flickr.photos.getRecent" arguments:[NSDictionary dictionaryWithObjectsAndKeys:@"1", @"per_page", nil]];`

# Pick up response

- ✦ You need delegate methods to pick them up
  - ✦ - (void)flickrAPIRequest:(OFFlickrAPIRequest \*)inRequest didCompleteWithResponse:(NSDictionary \*)inResponseDictionary;
  - ✦ - (void)flickrAPIRequest:(OFFlickrAPIRequest \*)inRequest didFailWithError:(NSError \*)inError;
  - ✦ - (void)flickrAPIRequest:(OFFlickrAPIRequest \*)inRequest imageUploadSentBytes:(NSUInteger)inSentBytes totalBytes:(NSUInteger)inTotalBytes;

# Sample Apps

# RandomPublicPhotos

- ✦ Demonstrates how to make a Mac app with OF
- ✦ Uses public API only, no auth required

# SnapAndRun for iPhone

- ✦ An iPhone sample app you can use!
- ✦ Uses camera and photo library on your device
- ✦ Demonstrates how to use auth with customized URL scheme (the mainstream method now)
- ✦ Multi-state, asynchronous message flow pattern