Integrating with Blogs

Blogs (also known as *weblogs*) have become lightweight, general-purpose platforms for publication, self-expression, and collaboration. Bloggers push the limits of new-media production, especially in the area of integration, because they want ultimately to discuss anything they can see or think or hear—without any effort, of course. Because you can directly tie blogs in with other systems—often without any programming on your own part—you'll now study how to combine blogs with other applications and data sources. In this chapter, I cover end-user functionality that lets you publish content to a blog from a web site or a desktop application. In Chapter 7, you'll study how you can program the relevant web APIs to read and publish blog content. I close this chapter by applying lessons from blog integration to wikis, which I believe are ripe for a similar type of remixing.

In this chapter, you will do the following:

* You'll learn how to configure your WordPress or Blogger blog to receive pictures from Flickr through Flickr's Blog This button.

* You'll study the mechanisms behind blog integration by studying how it's done with Flickr.

* You'll learn how to use a desktop blogging client to take advantage of a richer writing environment for blogging.

* You'll see how the combination of syndication feeds and blogging can be recursive (that is, how content from blogs can be refashioned into new blog entries).

* You'll experience the forward-looking social browser integration of Flock, which combines a Web browser, Flickr photos, and blogging all in one user interface.

I'll first cover the mechanics of blogging from the point of view of the user, and then I'll cover what this means in terms of the back end (specifically the use of APIs once again, this time for blogs).

Integration Scenarios for Blogs

Essentially, blogs are online journals about a topic, a theme, or a person written by one person or a small group. Here are other general patterns:

* Blogs consist of entries that are typically displayed in reverse chronological order.

* These entries are often classified into categories.

* Most blogs provide their content via RSS or Atom syndication.

In Chapter 4, I discussed how RSS/Atom syndication makes the life of a reader simpler by allowing the reader to aggregate content. In this chapter, you'll examine how the lives of blog authors can be made simpler. Wouldn't it be great to be able to do the following?

1. First write or create some piece of digital content (it could be simple text, HTML, images, video, or a word-processed document) in the tool of your choice (Microsoft Word, OpenOffice.org, the rich-text editor of WordPress, vi, Thunderbird).

2. Then easily publish that content to a blog (or any other web site). That is, you could have the piece of content you wrote show up in a blog in a way that preserves the formatting—or at least translates that format appropriately to the new environment—without having to do much (or any) of the manual work of translating that formatting.

We have not pulled off such general seamless integration yet. However, we will examine some specific and useful cases of integration in this chapter. Figuring out how integration happens in these specific scenarios enables you to build not only your own tools for supporting similar circumstances but also solutions to the general integration problems.

FREE HOSTED BLOGS AS A WAY TO START BLOGGING

If you are not already using a blog, it's useful to set up an account with which to experiment. You can download blogging software, write your own, or pay for blog hosting, but the easiest way to get started is to use one of the following free hosted blogging services:

* WordPress (http://wordpress.com)

* Blogger (http://blogger.com)

* LiveJournal (http://livejournal.com)

There are others, but these three should get you started.[1]

Sending Flickr Pictures to Blogs

As you have seen in previous chapters, Flickr provides excellent functionality to display and add narration to your photos; you can create slide shows; create sets; tack titles, descriptions, and tags to photos; and create groups to collaborate with others with similar interests. Yet, it is natural to want to present your photos outside the world of Flickr. If you have a personal blog, would you not want to display your photos on your own blog and tell stories around them?

1.   http://blogs.about.com/od/blogsoftwareandhosts/a/topfreeblogs.htm

As a Flickr user, you can automatically post a photo to your blog, provided you do the following:

1. First configure Flickr to work with your blog.

2. Hit the Blog This button for the desired photo.

The following sections are detailed instructions on the previous two steps. Before I cover how to use the automated process, I'll cover how you would manually present a photo from Flickr on your blog. You would do the following:

1. Generate the appropriate HTML for the photo in question. For this to work, you would need to know the URL for the actual image, as well as the URL for the photo page. You could grab the URL of the image from the web browser (through right-clicking the image and copying the image URL, for instance).

   If the photo in question is your own, Flickr provides some help in this department. For a given picture, hit the All Sizes button. For a given size of the photo, you can copy and paste the HTML given under the "Copy and Paste this HTML into Your Webpage" heading.

2. With the HTML now in hand, you would go to your blog to create a new post and then paste in that HTML.

Flickr helps automate this process by using blogging APIs. I'll now cover how.

Configuring Flickr for Integration with Blogs

Before you publish your photos from Flickr to a blog, you need to tell Flickr about the blogs you plan to use. Here are step-by-step instructions for configuring your blogs for access by Flickr:

1. Go to `http://flickr.com/blogs.gne`. You have to sign in to Flickr first.

2. Hit the Add Another Blog link (`http://flickr.com/blogs_add.gne`). You will see a list of weblogs that you have already configured. Note the types of blogs supported by Flickr:

   * Blogger
   * TypePad
   * Movable Type
   * LiveJournal
   * WordPress
   * Manila
   * Atom
   * Blogger API
   * MetaWeblog API
   * Vox

Depending on the type of blog you want to integrate with, the parameters you will need to fill in differ.

WHY IS THERE A LIST OF BLOG TYPES IN THE FIRST PLACE?

If all you are interested in is setting up Flickr to enable you to send a photo to your blog, you do not need to understand why there are so many blog types listed. If, however, you are interested in the mechanisms behind blogging integration, it's useful to ponder what you see here.

For instance, why does Flickr ask about the type of blog you have? It's conceivable that Flickr would not have to ask that question at all if all blogs were the same in terms of the mechanics of integration. The fact that this question is asked indicates that there is some sort of dependency on the blog type that affects how Flickr connects to the blog. But if your blog type is not on the list, what are you supposed to do? What exactly are those dependencies, and can they be formulated in terms of parameters of the system? I'll return to these questions later in this chapter.

Let's take a look at two types of blog software to understand some of the necessary parameters involved in blogging integration: WordPress and Blogger.

WordPress

To add a WordPress blog to your Flickr configuration, do the following:

1. Go to `http://flickr.com/blogs_add.gne`. Make sure you have a WordPress blog that you own for this example. You can either install your own WordPress blog on your hosting service or use the free WordPress service (see the "Free Hosted Blogs As a Way to Start Blogging" sidebar).

2. Click WordPress Blog in response to the question "What kind of blog do you have?" Note that with this choice you end up at the URL `http://flickr.com/blogs_add_metaweblogapi.gne`—which suggests that WordPress is accessible through the MetaWeblog API.[2]

3. Enter the following parameters:

    * API endpoint (for WordPress blogs, the URL is `http://{url-of-your-blog}/xmlrpc.php`, for example, `http://blog.mashupguide.net/xmlrpc.php`)
    * Username
    * Password

4. After you hit Next—and assuming you entered the correct combination of API endpoint, username, and password—you have the choice of storing the password on Flickr and changing the URL or label. After you have entered your choices, click All Done.

5. You can now choose a template for your blog and customize it (if you know HTML and CSS).

6. You can test the blog configuration by issuing a test post. To do so, go to `http://flickr.com/blogs.gne`, and click the Test Post button that corresponds to the blog. If things go well, you'll get the message "A test post to [name of your blog] has been sent. Feel free to delete it once it's gone through," and you should see a test post on your blog.

2.   http://en.wikipedia.org/wiki/MetaWeblog

Blogger Blogs

Blogger is another popular host of free blogs and is owned by Google. To add a new-style Blogger blog to Flickr, do the following:

1. Select Blogger Blog from the drop-down menu at http://flickr.com/blogs_add.gne. Make sure you have a Blogger blog, which you can sign up for at http://www2.blogger.com/create-blog.g.

2. At this point, you may be asked to head over to Google to authorize Flickr's access to your blog. If so, you will see a prompt like that in Figure 5-1. If not, skip to step 4.

Insert 858Xf0501.tif

Figure 5-1. A prompt from Flickr explaining the authentication process required to enable you to send Flickr photos to new Blogger blogs. (Reproduced with permission of Yahoo! Inc. ® 2007 by Yahoo! Inc. YAHOO! and the YAHOO! logo are trademarks of Yahoo! Inc.)

3. At Google, if you are not already logged in to Blogger, you will be prompted to log in. (Notice that it is Google/Blogger asking for the login, not Flickr here.) Once you are logged in, you'll be asked to grant access rights to Flickr, as shown in Figure 5-2. Note the comment "Flickr.com will not have access to your password or any personal information." There is a fuller explanation of how the authentication scheme works available on the Google web site.[3]

Insert 858Xf0502.tif

Figure 5-2. A prompt from Google requesting authorization to post to your Blogger blog from Flickr

3.   https://www.google.com/support/accounts/bin/answer.py?answer=41192

4. If you grant access to Google, your browser will be redirected to a page presenting you with a drop-down list of the blogs available at Blogger. Select the one you want.

5. Verify the settings (you can change the URL and label). Hit the All Done button.

6. Test your settings by making a test entry to your blog. Do you see a test entry on your blog? If so, your parameters are correct.

Notice that you never enter your username/password for your Blogger blogs to Flickr at any time during the process.

Blogging a Flickr Picture

Once you have a blog configured for blogging from Flickr, you are now ready to write a blog post based on a photo directly from Flickr. Here's how:

1. Go to a specific Flickr picture, and hit the Blog This button located above the picture.

2. Choose the blog from the list to which you want to send the picture.

3. Fill out the title and your post; I often find it helpful to copy and paste the description of the picture into the post. Hit Return.

If everything goes according to plan, you'll see the message from Flickr saying "Your blog entry has been posted!" and a URL to your blog so that you can check out your new post.

Note   Sometimes, you will get errors (such as timeouts). Often you can just try again. Sometimes Flickr reports an error when the post actually goes through and you can end up with multiple posts should you try again.

How Does the Flickr Blog Integration Work?

After you have configured a WordPress or Blogger blog and posted a picture, I encourage you to think about what must be happening underneath the hood to make the Flickr blogging interaction happen. Here are some specific issues to consider and questions to ask:

* Note the contrast in the parameters needed for a WordPress blog and a Blogger blog. For the WordPress blog, you need to enter an API endpoint along with the user and password, whereas for Blogger, you don't enter those credentials but are redirected to Google for authorization. Here are some issues to consider:

    * What do you think is happening differently to account for the contrast in functionality?

    * Why do you not need to type in an API endpoint for Blogger?

    * Where do you send username/passwords for each case?

    * What are the advantages and disadvantages of each approach?

* Note the wide variety of classes of blogs recognized by Flickr. You can try each type to study the parameters required to make each type of blog work.

* Note that once you blog a picture in the Flickr interface, a list dynamically pops up via Ajax.

* Study the types of templates available and how you can customize them via CSS and HTML.

After I describe web APIs in detail (for Flickr and for other web sites), I'll answer the questions I just posed in Chapter 7. There I explore in greater detail the use of blogging APIs. Still, without diving into technical details about the APIs, you can make several observations:

* Once you have configured a blog for access by Flickr, the process for publishing a photo is the same regardless of the actual blog you use.

* By contrast, Flickr is unable to smooth out the differences among weblogs to make the configuration process look identical. That means the protocols for connecting to WordPress and to Blogger probably differ.

* Blogging protocol must address the important issue of authentication and authorization; the process in which you grant Flickr the power to post to your blog depends on the type of blog you use.

Desktop Blogging Tools

You have just seen how you can send HTML that encodes a photo and description from Flickr to a blog. It should not then be surprising to find out that you can send data to blogs from systems other than Flickr. Indeed, a whole genre of tools lets you compose and post blog entries in a more convenient environment (such as a desktop application) and then send those posts to your blog instead of having to use the native blog post interface. The following are examples of blogging clients:

* w.bloggar (`http://www.wbloggar.com/`) is a Windows desktop client.

* ecto (`http://ecto.kung-foo.tv/`) for Windows and Mac OS X.

* MarsEdit (`http://www.red-sweater.com/marsedit/`) is for Mac OS X.

* Windows Live Writer (`http://windowslivewriter.spaces.live.com/`) is a desktop client for Windows.

* BlogDesk (`http://www.blogdesk.org/en/index.htm`) is a Windows desktop client.

* ScribeFire (`http://www.scribefire.com/`) is a client right within Firefox.

* mo:Blog (`http:www.moblogworld.net/`) is a client for Palm OS devices.

Some brave souls such as Jon Udell are even doing cutting-edge experiments of blogging from Microsoft Word 2007.[4] Figure 5-3 shows how it looks to write a blog post in one of these clients.

Insert 858Xf0503.tif

Figure 5-3. Writing to a WordPress blog from the Windows w.bloggar client. Note that the post already exists on the blog and that w.bloggar is being used to post it for editing.

It is instructive to ponder why there are so many tools in this area, what exactly is being integrated by the tools, and the exact list of functionality in these tools. Answers to these questions shed light on how users actually write blogs. For instance, Brent Simmons' description of MarsEdit, which he created, gives some insight into the genre:[5]

> *MarsEdit is weblog posting and editing software. It makes writing for the web like writing email: you open a window and write something, then send it to your weblog. It has many of the same features that email applications have: drafts, text editing commands, even AppleScript support. It also has*

features specific to weblogs: categories, text filters, trackbacks, pings, and so on. People that have more than one weblog find it especially useful because they have just one place to write and edit all their weblog posts, even if their weblogs are on different systems.

4. http://blog.jonudell.net/2007/02/19/blogging-from-word-2007-crossing-the-chasm/#comments

5. http://www.newsgator.com/NGOLProduct.aspx?ProdID=MarsEdit

Combining Feeds and Blogging to Generate Feedback Flows

In blogging there is often tight coupling between reading other people's blogs and writing your own blog entries. If you happen to be reading other blogs through a feed reader, you might even be able to easily drop pieces of other people's blogs (that are coming in as RSS or Atom items) into your own blog editor.

For example, on Windows, using SharpReader[6] combined with w.bloggar[7] and the w.bloggar SharpReader plug-in,[8] you can directly write blog entries based on items coming into your SharpReader news feeds (in a process that has been called *reblogging*), as shown in Figure 5-4.

Insert 858Xf0504.tif

Figure 5-4. On a Windows desktop, SharpReader is looking at one of Udell's posts, along with a right-click invocation of w.bloggar to send this entry to a blog.

Since reblogging often produces nothing more than trivial republication of other people's words, it's easy to forget that this flow of content is actually undergird by a feedback loop of reading and writing. When you use Flickr's blog functionality, content goes from Flickr to a blog, but there's no easy flow of content from blogs back into Flickr. By contrast, the combination of weblogs that syndicate their contents through feeds and feed aggregators that are also blog clients means that what you read can flow easily into what you write. In the next section, I'll discuss Flock, a web browser that facilitates this flow between reading and writing by building in greater integration with blogging and various social media web sites.

6. http://www.sharpreader.net/

7. http://wbloggar.com/

8. http://www.sharpreader.net/plugins.html

Flock: Bringing Together Blogs and Flickr

Flock (`http://flock.com`) is advertised by its creators as the "social web browser." Built upon the Firefox code base, Flock incorporates the following in its own interface:

* Flickr, Photobucket, and YouTube integration

* Blogging integration (including Blogger, LiveJournal, TypePad, WordPress, and various self-hosted blogs), as shown in Figure 5-5

* Integration with your social bookmarks at del.icio.us and ma.gnolia.com

    *   Drag-and-drop functionality that allows you to drag Flickr photos into a writing toolbar that then connects to your blogs

Experimenting with the Flickr and blogging integration[9] in the Flock browser is a useful way to see the flow of data between systems that are starting to be built into service composition frameworks (see Chapter 11).

Insert 858Xf0505.tif

Figure 5-5. Blogging Flickr photos from Flock by dragging and dropping multiple Flickr photos into an editing window and then posting that entry into a configured blog. (Reproduced with permission of Yahoo! Inc. ® 2007 by Yahoo! Inc. YAHOO! and the YAHOO! logo are trademarks of Yahoo! Inc.)

9.   http://www.flock.com/faq/show/29#q_8369

RSD: Discoverability of Blog APIs

If you configure Flock for blogging, you might wonder why some blogs can be configured by simply entering the URL of the blog only, while in Flickr, you sometimes need to enter the URL to the specific API endpoint. How is Flock able to find the URL endpoint from the URL of the blog? Finding the URL of the API endpoint is similar to the problem described in Chapter 4 of locating the URLs of feeds based on the URL of the web site. You won't be surprised then to discover that someone invented an autodiscovery mechanism for the existence of blogging APIs:

```
http://en.wikipedia.org/wiki/Really_Simple_Discovery
```

For detailed technical information on the mechanism, read the RSD specification:

```
http://cyber.law.harvard.edu/blogs/gems/tech/rsd.html
```

Here I point out how RSD has been implemented by at least two major blog publishing services: WordPress and Blogger. You can go to any of the blogs run by WordPress, such as the one for WordPress news:

```
http://wordpress.com/blog/
```

in which you will find the following link:

```
<link rel="EditURI" type="application/rsd+xml" title="RSD"
     href="http://wordpress.com/xmlrpc.php?rsd" />
```

From looking at http://wordpress.com/xmlrpc.php?rsd , which is as follows:

```
<?xml version="1.0" encoding="UTF-8"?><rsd version="1.0"
    xmlns="http://archipelago.phrasewise.com/rsd">
  <service>
    <engineName>WordPress</engineName>
    <engineLink>http://wordpress.org/</engineLink>
    <homePageLink>http://wordpress.com</homePageLink>
    <apis>
      <api name="WordPress" blogID="1" preferred="true"
           apiLink="http://wordpress.com/xmlrpc.php" />
      <api name="Movable Type" blogID="1" preferred="false"
```

```
        apiLink="http://wordpress.com/xmlrpc.php" />
    <api name="MetaWeblog" blogID="1" preferred="false"
        apiLink="http://wordpress.com/xmlrpc.php" />
    <api name="Blogger" blogID="1" preferred="false"
        apiLink="http://wordpress.com/xmlrpc.php" />
  </apis>
 </service>
</rsd>
```

you can see how the WordPress blog is advertising itself as having support for four blog APIs: WordPress, Movable Type, Metablog, and Blogger.

Similarly, for Blogger blogs such as http://googleblog.blogspot.com/, you'll get the following:

```
<link rel="EditURI" type="application/rsd+xml" title="RSD"
    href="http://www.blogger.com/rsd.g?blogID=10861780" />
```

And http://www.blogger.com/rsd.g?blogID=10861780 shows support for one API, the Blogger API:

```
<api name="Blogger" preferred="true" apiLink="http://www.blogger.com/api"
    blogID="10861780"/>
```

Like feed autodiscovery, RSD functions as a reasonably well-implemented de facto standard without much formalization.

Linkbacks

I'll now explain a type of communication flow that you might notice from studying blogs (though not directly from how Flickr interacts with blogs). Among comments listed for a given blog post are often entries that come from other web sites. How is a blog able to track links that come from the outside? Weblogs use *linkbacks*, a family of methods for receiving notifications of inbound links to a web site.

As documented at http://en.wikipedia.org/wiki/Linkback, there are three major protocols for linkbacks: refback, trackback, and pingback. It's useful to know which of the protocols are supported by various blogging software so that you know which of the protocols to support if you set out to use linkbacks. Why might linkbacks be useful for mashups? You may want your mashup to either notify web sites that it links to or receive notifications of being linked to.

Note that Flickr doesn't support linkbacks, although it notifies you when someone else adds a comment to your picture or makes it a favorite.

Wiki Integration at an Early Stage

Wikis are web sites for bringing together user contributions, though they are designed to be more radically collaborative than blogs. According to the Wikipedia, a wiki is as follows:

> *A website that allows the visitors themselves to easily add, remove, and otherwise edit and change available content, typically without the need for registration.*10

The ideal scenario for wikis is allowing anyone to edit pages, combined with a lack of broken links. That is, when a user follows a link to a page that doesn't exist, the user is not given a 404 error but rather the opportunity to create that page.

I mention blogs and wikis together in this chapter because they are siblings. Indeed, there are hybrid blogs/wikis—or at least experimentation to bring them into hybrid structures.[11] And there are other similarities between blogs and wikis: both are used to publish web sites, both can have APIs that facilitate integration, and both tend to have plug-in infrastructures that make them more like platforms than simple software. This combination of APIs and plug-ins increases the mashup opportunities.

10.   http://en.wikipedia.org/wiki/Wiki—accessed as http://en.wikipedia.org/w/index.php?title=Wiki&oldid=109882004

11.   http://www.docuverse.com/blog/donpark/2003/09/05/wiki-based-web sites

You have seen some complicated ways in which the tools and data involved in blogs aren't being mashed up. Although the potential for wiki mashups is great, there are a lot fewer examples of such mashups. Much of the technical foundation is in place (for instance, many wikis have APIs and plug-in frameworks[12]), but the uptake of wikis is less than that for blogs.

The closest thing to a mass phenomenon we have in the world of wikis is Wikipedia. It's not surprising then to see some mashing up of Wikipedia, though not as much as you might expect. Let's look at one example of a remix of Wikipedia, FUTEF, which is a custom search engine that draws content from Wikipedia (`http://futef.com/`):

1. Go to `http://futef.com/`, and type **Bach** into the search engine.

2. Study the search results that come back, their order, and the categories listed.

3. Compare what you see in FUTEF with what you get from the same search in Wikipedia. In Wikipedia, you get an immediate redirection to the article on Johann Sebastian Bach. For other Bach-related terms, study the Bach disambiguation page.[13]

Curiously, FUTEF has built its own API that it has invited others to use.[14] Why, for instance, would anyone use FUTEF's API to access Wikipedia when Wikipedia provides its own? Well, once FUTEF fulfills its plans to offer content other than Wikipedia, I can see a good reason for trying the FUTEF API. At this point, I'd say FUTEF is useful primarily as a demonstration of how you can repackage Wikipedia.

Other places to look in terms of integration with Wikipedia is in authoring tools akin to blogging clients and in bots that have been written to support the editing of Wikipedia. You can find a list of such editors here:

`http://en.wikipedia.org/wiki/Wikipedia:Text_editor_support`

And you can find a discussion of Wikipedia bots here:

`http://meta.wikimedia.org/wiki/Bot`

Summary

Here are a few points to remember from this chapter:

* Flickr lets you blog a single picture. From this function, you can see a specific instance of data being sent to blogs.

* There are many types of blogs, and they require different type of configuration schemes.

* Flock tries to envision a future in which a whole bunch of tools are integrated: a web browser, Flickr, blogs, and social bookmarking.

* You can generate a feedback loop using RSS, news aggregators, and blogging, and most blogs automatically generate RSS.

* Blogs represent a type of remixing in a narrative, in contrast with the data-oriented remixing via tags and straight RSS so far discussed.

Now that you have studied how these tools work, you are in a good position in the coming chapters to start building your own tool. You will create some mashups step by step, remembering what you have seen as an end user of these tools.

12.   http://en.wikipedia.org/w/api.php for MediaWiki and http://api.pbwiki.com/ for PBWiki, which is a popular free wiki host provider

13.   http://en.wikipedia.org/wiki/Bach_%28disambiguation%29

14.   http://www.programmableweb.com/api/FUTEFWikipedia