

Weblog 2.0

Jonathan Eisenzopf
Copyright © 2001 by internet.com
April 30, 2001

Weblog 2.0 is released.

Overview

While I'm still working on the VoiceXML adventure game, I have managed to finish another release of Weblog for Internet.com. It's basically a major rewrite. Other than departmentalizing the Weblog functionality, the major addition is the integration and rewrite of Xhoo. In short, this version of Weblog is a move towards a well-defined environment for creating, editing, syndicating, and organizing channels. See the full feature set below.

Integration with Xhoo

If you haven't heard of it before, Xhoo is a Web catalog that I developed as an article a while back. You can still read about it off the Mother of Perl home page. The difference now is that Xhoo itself is built using RSS files, which allows you to leverage all of the same tools that are now available for manipulating RSS. In the future, Xhoo will probably support the RSS 1.0 threads module as soon as it matures a bit.

Weblog channels link to Xhoo categories through the taxonomy RSS 1.0 module, which is available at http://groups.yahoo.com/group/rss-dev/files/Modules/Proposed/mod_taxonomy-1-2.html [http://groups.yahoo.com/group/rss-dev/files/Modules/Proposed/mod_taxonomy-1-2.html].

Here's how it actually works with Weblog. When you add a new headline, the screen lists all of the Xhoo categories. So, in addition to adding the headline to the channels that you've selected, Weblog will also add the headline title, link, and description to the selected Xhoo categories. This will make it easier to preserve important headlines that go out of print in your news channels. It also provides a simple archive and categorization mechanism for the items in your channels, so that you can reference them later.

Better support for RSS 1.0 modules

As mentioned above, Weblog channels link to Xhoo categories by utilizing the taxonomy module. Hence, the items that have Xhoo entries will include extra information. In addition, Weblog 2.0 uses the Dublin Core Metadata module quite extensively to include dates, authors, and all that other neat stuff people have been wanting in RSS for a

while. For example, below is an item that was created by David Christian. It's also linked to the David/ category in Xhoo as indicated by the *rdf:li* element inside the *taxo:topics* element. One nice thing about this particular module is that it allows for multiple entries. So you could link to another category in Xhoo or even to another catalog.

```
<item rdf:about="http://www.cnn.com">
<title>Test Title</title>
<link>http://www.cnn.com</link>
<description>This is a test.</description>
<dc:subject>Test title</dc:subject>
<dc:creator>David Christian</dc:creator>
<dc:identifier>987428479:841313774230</dc:identifier>
<dc:date>2001-04-16T14:41</dc:date>
<taxo:topics>
  <rdf:Bag>
    <rdf:li resource="http://www.textant.com/xhoo/David"
  />
  </rdf:Bag>
</taxo:topics>
</item>
```

The Dublin Core *dc:identifier* element is used as a unique record id for each item. Dublin Core elements are also supported at channel, image, and input levels. This id is used internally in the application to track unique items. It also makes it easier to import the RSS content into and out of a database.

Better User Interface

With the integration of Xhoo, the Weblog interface underwent a major overhaul. Below are some screenshots to give you an idea of what it looks like now.

- Channel Home [channel.gif]
- Adding a Headline [add_headline.gif]
- Channel Settings [channel_settings.gif]
- Image Settings [image_settings.gif]
- Input Settings [input_settings.gif]
- Xhoo Admin Screen [xhoo_admin.gif]

- Xhoo Sub-Category [xhoo_admin_sub.gif]

New Modules

Weblog Module

I've rewritten the Weblog functionality into a discrete set of modules that can be reused to develop other applications. I did this to make it easier for me to extend the code for use in an aggregator and for being able to post to remote Weblogs via XML-RPC. The modules will probably change in the future, but it's a good set of code where you need to manipulate multiple RSS channels in a production environment. For example, below is a script that loads a channel, adds an item, saves the channel, and outputs the channel to several different formats:

```
use strict;
use Weblog;
my $config_file = '../config/config.xml';
# create new instance of the Weblog module
my $weblog = new Weblog($config_file);
# get a list of the channels that are available
my @channels = $weblog->list_channels(local => 1);
foreach my $channel (@channels) {
    print $channel->{'name'}, "\n";
}
# load the channel into memory
my $channel = new Weblog::Channel($weblog, "webnews");
$channel->creator('Andy King (aking@internet.com)');
$channel->publisher("Webreference.com");
$channel->subject("Internet/Webdev");
$channel->contributor("Jonathan Eisenzopf");
$channel->type("News Channel");
$channel->format("RSS 1.0");
$channel->identifier("11111");
$channel->source("Author");
$channel->relation("father");
$channel->coverage("eastern");
$channel->rights("2001 internet.com");
$channel ->taxo(['http://www.webref.com',
               'http://www.news.com']);
# add a new item in the channel
my $item1 = new Weblog::Item(title => 'dddd',
                             link => 'aaaaaa',
                             description => 'eeeeeeee',
                             taxo => [
                                 'http://www.da.com',
```

```

        'http://www.ba.com'
    ]
);
my $item2 = new Weblog::Item(title => 'dddd',
                             link => 'aaaaa',
                             description => 'eeeeeee'
);
# if you want to add one item to multiple channels
# $weblog->add_item($itemref, "append", $channel1, $channel2, $channel3);
# add the items to a channel
$channel->add_item($item1, "insert");
$channel->add_item($item2, "append");
# save the channel
$channel->save;
# output
my $output = new Weblog::Output($channel, "Template");
#$output->outputall("all");
$output->output("vxml");
$output->output("html");
$output->output("palm");
$output->output("wml");
$output->output("avantgo");

```

There are several example scripts that I used for testing in addition to the modules themselves that are sitting in the lib/ directory. Feel free to play around with them. This module has not been finalized or fully documented, so expect it to change. When it becomes more stable, I will probably release it to CPAN.

New Modules (Cont.)

Khoo Module

In addition to the Weblog module, I was able to modularize Khoo itself to make it easier to read and write to Khoo from an external program, like Weblog. Below is a sample script that demonstrates how it works:

```

use strict;
use Khoo;
# initialize local variables
my $dir = '/home/eisen/work/weblog/weblog-1.8/directory';
my $datafile = 'khoo.xml';
my $identifier = '986086673:382596210649';
my $category = 'Computers/Perl';
my $url = 'http://www.webdeveloper.com/directory';
my $khoo = new Khoo (rootdir => $dir,

```

```
                                baseurl => $url);
my %link = (
    name           => 'Jonathan Eisenzopf',
    email          => 'eisen@xif.com',
    description    => 'asdasdas',
    url            => 'http://www.news.com',
    title         => 'CNet News'
);
foreach my $category (@{$xhoo->categories}) {
    print "$category\n";
}
$xhoo->add_link($category, $datafile, $identifier,%link);
$xhoo->edit_link($category,$datafile,$identifier,%link);
$xhoo->delete_link($category,$datafile,$identifier);
```

The code

The source code is still licensed under the GNU license and is available as part of the tutorial download, which is linked to on the first page of the article. Enjoy.