

Wiki Wiki Wah Wah

It may have a strange name, but Ramin Nakisa finds the Wiki is transforming the way people collaborate at work

A friend of mine who has strong Celtic roots bemoans the fact that his ancestors shunned writing. Instead Celts had a strong oral tradition that passed myth, poetry, and song between generations. He thinks that it is a tragedy that most of our knowledge about Celtic life in Britain comes from very few writings, many of them from Roman sources. Compare this with the Greeks and Romans, whose lives we can reconstruct in incredible detail including their history, military strategies, cuisine, religion and system of medicine. These cultures encouraged writers, and because of this their knowledge outlived their culture. The Renaissance was fuelled by a rediscovery of this cache of classical thought, and if it were not for these authors, and the monks who meticulously copied their manuscripts, we could still be living in a Dark Age.

The act of preserving and sharing knowledge is just as important today. We must all have been in a situation where a critical employee leaves, taking their business knowledge with them and leaving a huge knowledge gap that is impossible to fill. Some employees actively hoard knowledge in order to make themselves indispensable. Contractors in the information technology industry are notorious for writing software that is incomprehensible to and unmaintainable by anyone but themselves. For many enterprises, particularly commercial enterprises, it is important to capture knowledge quickly and easily, and put it in a form that is easy to comprehend. This has given rise to a buzzword describing the process of capturing and storing knowledge: "knowledge management". There are expensive software packages that claim to be able to implement knowledge management, but one of the best is reducible to just 36 lines of Perl code, and is available under



the GNU General Public License at no charge. This software is called WikiWiki, from the Hawaiian word “wee kee wee kee” meaning “quickly”, and is often abbreviated to the more easily pronounceable “wiki”. Those humble 36 lines of code may well change your life and the way you work.

Wiki provides a means by which a community of users can collaboratively create and edit a web site. The name wiki refers to the software used to generate the site, and also the web documents themselves. Wiki uses a simple markup language, similar to Hypertext Markup Language (HTML), and you do not need an editor to edit a wiki because this is done through the web interface itself. Submissions are not peer-reviewed before being visible to other users so that the time taken to modify and save a wiki can be as little as ten seconds. One of the labour-saving aspects of wiki is that you do not have to use complicated link syntax. The wiki engine detects a link when the text in a wiki uses a form of capitalisation known as CamelCase. CamelCase is a word in which a capital letter is surrounded by lowercase letters, so for example these words are in CamelCase: LinuxUser, LiNUXUSER, LinuxusEr. These words are not: LINUXUSER, Linuxuser, LINUXUSEr. When wiki comes across a document containing CamelCase it automatically makes a link to a new wiki page of that name. If the wiki has not been created yet, a question mark appears after the link inviting users to define the page.

An important part of the ethos of wiki is that there are no permissions, so that any user can edit any page. If someone does not agree with the contents of one of your wikis they can change your text in just a few seconds, and this could escalate into an “edit war”. Malicious users could vandalise the wiki. In practice, vandalism is seldom a problem. Firstly, wiki has built-in version control. This means that for any page you can view the edit history, and revert to any previous pre-vandalisation version. The second reason is that vandalising a wiki is so easily done, and so easily fixed, that not many people bother to do it. If you are a cracker you will get respect from your peers for breaking into the Pentagon’s network, whereas maliciously modifying a wiki about configuring Emacs is unlikely to earn much respect.

Wiki history

WikiWiki was created by Ward Cunningham, one of the leading lights in the design pattern approach to programming. The path that led Ward to WikiWiki started when one of his friends, who worked at Apple, showed him a copy of HyperCard. This was an application that came bundled free of charge with every

Macintosh computer that allowed users to create cross-referenced documents. Ward used the program to create a HyperCard stack of design patterns. Later, when working for Tektronix, he decided to use HyperCard to try and track the flow of ideas in the company. Initially he wanted to create a stack with three categories of cards: cards for ideas, cards for people who had and held the ideas, and cards for projects where ideas were shared. He decided to use a single, simple card that could capture all three categories that had just three fields: name, description and links. The seed of WikiWiki was in the links field. Although cards were easy to edit in HyperCard, links were awkward. Ward made the links field automatically search for cards with that name, and beep if the link was not found. If the user held down the missing link button the missing card would be created for them.

Although this hyperlinking method sounds like a small innovation, it reduced the effort required to cross-reference cards below a critical laziness level. The stack was a runaway success. Ward would invite people to his desk to see the stack. After being shown a few people and projects they would see something and say “that’s not right”, immediately fix the problem, and create new links to ideas and projects that had been omitted. The stack was so compelling that it made people miss lunch, and in the information technology business this is high praise indeed. Later, when working for Wyatt Software, Ward wrote another stack describing their product WyCash. This gave birth to another signal feature of WikiWiki, a card called RecentChanges. Whenever any card was changed, a link was automatically added to RecentChanges recording the card that was changed, a time stamp, and a short text description of what had been done. After editing any card, Ward would find himself looking at RecentChanges to look over his changes. Some users of WikiWiki find looking at recent changes addictive, and these people are known as “recent change junkies”.

Writing a wiki engine is not particularly difficult, as wiki is, like many of the best ideas, very simple. After the success of the original WikiWikiWeb written by Ward Cunningham in Perl, many wikiclones were developed. There are now over eighty implementations in a wide variety of languages: Awk, Cheshire, C, C++, C#, Cold Fusion, Common Lisp, Erlang, Htag, Java, Perl, PHP, Prolog, Python, Rebol, Ruby, Scheme, SmallTalk, and Tcl. There are even wiki engines written using single-platform languages such as Active Server Pages, Visual Basic and Visual Foxpro. The number of implementations grew so large that it spawned a competition to see who could create a wiki producing core functionality with the least amount of

code. The minimal features included things such as automatic link generation, having content editable by anyone, having a simple markup language and back links, where clicking on the title of a page lists all pages that refer to that page. The winner so far is TinyWiki, written in 36 lines of Perl, followed closely by FinikiWiki at 38 lines of Perl, then RikiWiki at 40 lines of Ruby.

Text formatting rules

All wikis use a simple markup language, although syntax varies between implementations. In this article I will concentrate on UseMod because it is full-featured, simple to configure, and widely used. If you just want to type text into a page you do just that. Unlike HTML there are no outer tags. Extra spaces between words and line breaks are removed. Paragraphs are created by leaving a blank line between blocks of text. You can highlight text either by using HTML bold and italic tags or using the quote style where two single quotes are "italics", three single quotes are "bold" and five single quotes are bold and italic. Links to other wiki pages can either be created using CamelCase, such as LinuxUser, or by putting the title in square brackets such as [Linux User]. Links to web sites are detected automatically, or can be written [http://www.google.com/Search Engine] which would produce a link labelled "Search Engine" linked to the Google web site. Images are embedded as URLs and can point to images on your or other web sites. Image URLs are recognised by their file extension, and the extensions recognised include gif, jpg, png, bmp and jpeg.

You can have one of three different types of list. A simple list is denoted by an asterisk at the beginning of a line with different levels of indentation possible by adding more asterisks.

```
* first item
* second item
** second level item
*** third level item
```

If instead of bullets you want numbered items, use hash instead of an asterisk and UseMod wiki will replace the hashes with numbers. The final form of list is a definition, where a short phrase appears on one line followed by an indented paragraph giving a possibly lengthy definition. This is represented as a semicolon at the beginning of the line followed by the term being defined, then a colon, then the description. To add levels of indentation just add more semicolons.

```
;FIFO:First in first out.
;LIFO>Last in first out.
```

Sometimes the case of a word may mimic CamelCase when it shouldn't, so you can use the tags to switch off automatic link creation. If you want to create literal text that is not touched by the wiki engine and appears as you typed it, there is the pre-formatted tag `<pre>`. Anything within these tags appears in a fixed width font. This feature is useful for pasting code fragments into your page. If all you want to do is indent a paragraph, start it with a colon at the

beginning of the line, and to nest indentation use more than one colon.

The easiest way to learn text formatting rules is to use them yourself on the Meatball wiki SandBox at <http://www.usemod.com/cgi-bin/mb.pl?SandBox>. This is a page that has been set aside for people to try out wiki markup. Don't worry about making a fool of yourself because you don't have to actually save your edits. Just hit the preview button to see how they will appear in the final page.

Your own Wiki

First you will need to download UseMod from:

```
www.usemod.com/cgi-bin/
wiki.pl?UseModWiki/Download
```

The package comes as a 55 kilobyte gzipped tar file. The archive contains just 11 files, and the main one is `wiki.pl` that contains all the code required to run a wiki. When you unpack the archive with:

```
tar xvzf usemod092.tar.gz
```

You will find a directory `usemod092` is created with a sub-directory `usemod092/misc`. As with all standard GNU programs you will find installation instructions in the `INSTALL` file. First, copy the `wiki.pl` file into your web server's `cgi-bin` directory. Your web server may require the script to have the file extension `.cgi` to be recognised as a CGI script so you might need to rename the file to `wiki.cgi`. You will need to make the file executable:

```
chmod +x wiki.pl
```

And you may need to change the file permissions so that your web server has sufficient permissions to work properly:

```
chmod 755 wiki.pl
```

UseMod wiki stores all its pages as files stored on disk. You will need to create the data directory for this page store (I will use `/usr/local/wikidb` as an example). Make sure that the directory has the correct permissions so that the `wiki.pl` script can read from and write to this directory. Apache runs as user "apache" group "apache" on my box, so I recursively set apache as owner of this directory.

```
chown -R apache.apache /usr/local/wikidb
```

Now you will need to configure the `wiki.pl` script itself. Open the file in a text editor, and at line 58 you will find:

```
$DataDir = "/tmp/mywikidb";
```

You should modify this to point to the data directory you just created. Using the default `/tmp/mywikidb` is a bad idea because the `/tmp` file system may be purged automatically by your system. The rest of the configuration should be done outside the `wiki.pl` script. The configuration information is stored in a separate text file called "config".

Copy this file from the place you unpacked your UseMod archive file to the data directory, then edit the file. There are many settings in this file, but you will probably want to set some of them straight away. \$CookieName allows you to name the cookie that UseMod uses to keep track of users, so if you have a name for your wiki use it for this variable. \$SiteName is the title used on all pages by browsers viewing your wiki. \$HomePage specifies the page that browsers will be moved to by default if they do not specify a page or they click on your wiki's logo. \$LogoUrl points at the logo found on every page of your wiki that acts as a shortcut to the home page. You can quickly generate a good-looking logo in a few seconds in a good graphical editing package such as The GIMP (try menu Xtms, Script-Fu, Logos, Frosty, select a font and type your wiki's name into the dialog box).

Start here...

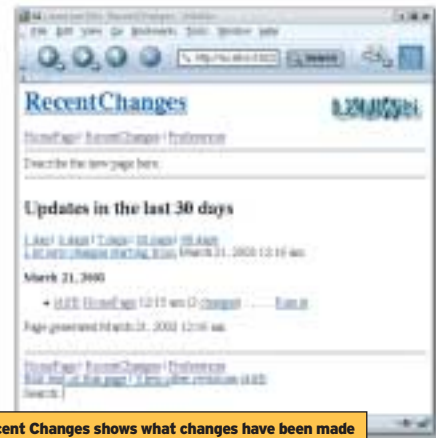


The entire configuration process, including the download, should not take more than about ten minutes, assuming you have a web server already configured to run Perl CGI scripts. Once you've completed the configuration steps you can take a look at your work by pointing your browser at <http://localhost/cgi-perl/wiki.pl> (or whatever path your server uses to access the wiki.pl script). You should see a page like that above.

Your text will appear between the horizontal rules. The links above and below the rules appear on every wiki page. The HomePage link takes you to this page from any other page, or alternatively you can click on the image in the top right-hand corner. RecentChanges shows globally all edits that have been made recently on the site. Preferences allows you to create a username, so that any changes you

make will appear under your name. If you do not do this entries will appear as an IP address. If you want to edit a page just click on "Edit text of this page" and the wiki will be replaced by a text entry box with two buttons beneath it: "Preview" and "Save". The preview is very useful for checking your markup does what you intended. "View other revisions" gives access to the version information for this page and "diff" gives side-by-side comparisons of

Gathering content



Recent Changes shows what changes have been made

your page versus previous versions. Finally, the search box allows you to search the entire contents of the wiki for Perl-type regular expressions that you specify. Now we can enter some content into the wiki, as shown below left.

We have entered some text and a summary of the change we have made in the "Summary" text box. When we hit preview we notice that a blue question mark appears after the word LinuxUser because it is coincidentally in CamelCase. So we put tags around the word, hit preview, and the question mark has vanished. To add a logo we enter the URL <http://www.linuxuser.co.uk/shared/newlogolarge.png> on the first line. This is recognised as an image because of the "png" file extension and wiki takes the image straight from the LinuxUser web site. To see what we have been doing we can click on the RecentChanges link and the output should look like that shown above.

RecentChanges shows us which page was changed, who changed it, and the summary of their changes, if they entered one. It also provides a link to a page showing differences between versions. The finished page, that should only have taken a minute to create, is shown below.



The finished page

Wiki applications

To see what is possible with wiki, take a look at Wikipedia, which is a collaboratively edited encyclopedia where articles are released as "open content" under the GNU free documentation license, or GFDL. GFDL ensures that any Wikipedia text or media are licensed by the copyright

Key links

Wikipedia:
www.wikipedia.org
 Ward Cunningham's
 Portland Pattern
 Repository:
www.c2.com/cgi/wiki/
 Flash coders wiki:
[http://chattyfig.
 figleaf.com/
 flashcoders-wiki/
 index.php](http://chattyfig.figleaf.com/flashcoders-wiki/index.php)
 Permaculture wiki:
[www.ibiblio.org/
 ecolandtech/pcwiki/
 index.php/HomePage](http://www.ibiblio.org/ecolandtech/pcwiki/index.php/HomePage)
 LinuxStation wiki for
 linux information:
[www.linuxstation.org/
 phpwiki/](http://www.linuxstation.org/phpwiki/)

holder to the general public, giving them the freedom to copy and redistribute the content for the benefit of other people. It also allows others to amend and improve the content and ensures that those freedoms pass on to future edited versions. You may think, as I did, that it would be impossible to edit an encyclopedia without strict editorial control. And yet Wikipedia has no conventional editorial control. Anyone can edit any page of the encyclopedia, even if they do not have an account on the system. Authors tend to write only on topics about which they are very knowledgeable, and because Wikipedia has so many users the breadth of knowledge it now encompasses is vast. Anyone can proofread articles and quickly correct errors of spelling and grammar. Wikipedia has become a spectacular success story for both wiki and the Free Documentation License. Although Wikipedia first came online just over two years ago, on January 15, 2001, it currently has 110,573 articles in the English version and over 37,000 articles in other languages.

Nupedia was the first Internet encyclopedia to be distributed under the GFDL and serves to illustrate an important wiki principle: people are lazy, and if something takes too much effort it will not be done. Nupedia's editor in chief, Larry Sanger, and many other editors, worked tirelessly to find people qualified in many disciplines who were willing to contribute articles, and also to review articles by other people. If you do not have a doctorate in a relevant field you are unlikely to be allowed to submit articles to Nupedia. So Nupedia, unlike Wikipedia, is exclusive. Nupedia has an elaborate multi-stage review process, similar to publishing a scientific paper, and it can take a long time for an article to see the light of day after it has been submitted. Rather than

quickly enter a short description of search algorithms into a wiki a Nupedia author has to submit an entire article, then wait for feedback from reviewers, then debate with those reviewers point by point, then amend the article after some sort of consensus is reached. Finally, after a few weeks or months, the article may be put on the Nupedia site. Currently Nupedia has little activity and very few articles, although those articles are of very high quality. In contrast Wikipedia is flourishing, even though it was set up after Nupedia, almost as an afterthought. Wikipedia is so successful that the servers running the site are barely coping with the number of user searches and the flood of new content.

Wiki comes into its own when used for IT support. IT systems have a tendency to grow more and more complex until no single human can understand all their intricacies. This is fine, as long as the system is documented. But the documentation has to be richly cross-referenced and easy to edit as the system changes, otherwise it rapidly becomes obsolete. And it is important to have a lightweight interface that is accessible through a 56k modem connection for overnight support. The documentation must also be globally searchable by keyword even across documents. Wiki satisfies all of these criteria, and has proved to be very successful as a repository of support information in my workplace. Apart from documenting our software we also use wiki to store things such as contact details, guides for our users, coding conventions, configuration files, lists of compiler bugs, and code fragments.

Ultimately there will be as many different applications of wiki as there are communities of users interested in sharing information. See the Key links box, for some publicly accessible wikis.