

Lightweight wikis without databases

QUICKIE WIKIS

If you are thinking about starting a wiki, and you don't want the overhead of a full-scale, database-driven system, try these light and lean wiki wonders. **BY FRANK WIEDUWILT**

Wikis are at the center of the world we know as Web 2.0. In just a few short years, the wiki has become one of the most popular tools on the Internet, thanks to a few very high profile projects and many smaller sites that are helping millions of users communicate and stay organized.

The wiki's open approach, where anyone can post and edit, makes it the perfect counterpart for another famous tool of the new Internet known as the blog. Wikis offer an easy and intuitive means for collecting and editing content.

If you are planning a large wiki with hundreds of users and thousands of pages, you'll need a full-featured, database driven wiki tool such as Mediawiki. But a community the size of the Wikipedia [1] community is atypical. For more moderate tasks, a full-fledged wiki system with an underlying database may have more disadvantages than benefits. Installing and managing the wiki is just too complicated. And for free webspace, where a database may not be available,

a wiki is only feasible if it uses flat files to store content.

Luckily, a number of smaller and simpler wikis are also available for Linux users. These lean wikis have some restrictions of course: projects should not have more than a few hundred articles. If the number of pages keeps growing, you can expect major performance

hits. However, if you're thinking about getting started with your own wiki and you don't need the horsepower of a high-end, database-driven system, a lean wiki may be exactly what you need.

The Candidates

We will be looking at five wikis, Wikepage, PWP, MoinMoin, PmWiki, and

For Bigger Projects

The wikis discussed in this article are not suitable for large-scale projects with several thousand pages. If you expect a large volume of content, and would appreciate more sophisticated user management features, you need a wiki with database support.

Mediawiki: Definitely the most popular wiki of them all, and the platform for the Wikipedia online encyclopedia and its local derivatives. Mediawiki [2] uses MySQL as its database back-end.

Wikka Wiki: This project [3] also uses MySQL to store text. The software is also extensible by means of plugins, and this makes it suitable for projects where the scope is not known right from the outset.

ErfurtWiki: This wiki [4] supports various databases, and can even use flat files as its back-end. The advantage is that you can use plugins, skins, and extension modules to modify the wiki's look and feel to match your requirements.

The Wikimatrix [5] page lists a number of wikis with and without database back-ends, describing each system's capabilities in a comprehensive overview.

Dokuwiki, all of which are under active community development. All of these projects promise simple installation, even for non-DBAs. In the simplest of cases, you just need to unpack the source code archive on the web server to run the wiki. Some systems support a customized look and feel based on templates, macros, or plugin-based extensions.

Access controls are an important issue. If you are looking to create a wiki for public access, the danger is that unfriendly web users or robots might stumble across your page and misuse it for propaganda or ads. The Tipiwiki [6] homepage is just one example of wiki misuse. In August 2006, the wiki was more or less completely inundated with ad links, leaving the pages unusable except for the welcome page.

While wikis like Mediawiki have sophisticated user management facilities that rely on a database back-end and allow administrators to define distinct user groups with distinct privileges, the smaller wikis often have nothing of the kind. The wiki dialect used for markups for tagging or linking is also an important aspect. If you need to move to a large system some time later, it would be nice to know whether you can do so without having to redo the formatting.

Wikepage

The motto of the Wikepage [7] wiki is “unpack and go.” The package weighs in at just 38 Kbytes, and its author, Sefer Bora Lisesivdin, explains the reason for the small size: in the good old days of the Commodore 64, developers just had 38911 bytes of user space to work with, but they still managed to come up with functional programs. He promises to keep the Wikepage to less than 38911 bytes, although he is working on a plugin interface.

Wikepage is based on the Tipiwiki [6] code and requires a web server running PHP 5.1.4 or better. To install the wiki, you just unpack the program archive, and you’re up and running. Wikepage supports themes, and you can even change the interface language by copying a language file to the wiki’s root directory.

Themes let you change the look, and there are a number of examples that you can use for your experiments at [8].



Figure 1: The simple structure of Wikepage lends itself to quick deployment, though the general lack of documentation slows down the creative process.

In Wiki mode, editing is open to any user. In Personal mode, users need to supply a password before editing a page. The password applies to all pages, however, and there is no support for distinct user privileges.

The formatting options range from various text markups, through lists, to simple tables. The developers also let you integrate images into wiki pages. The wiki dialect differs considerably from that of Mediawiki; you can expect difficulties with the process of exporting content to another wiki.

Unfortunately, Wikepage does not give users much in the line of howtos or manuals: neither the homepage nor the source code archive provide descriptions of the markups or a guide to creating your own themes.

PWP

PWP [9] stores wiki pages in flat files and requires a web server with PHP version 4.2 or newer. If you are using PHP5, the software requires PHP4 compatibility mode. Again, the basic configuration is easily done with PWP: just unpack the source code archive and open the index

page to get the wiki up and running. You can use any text editor to modify the configuration. PWP uses CSS to format screen output, and this makes it highly customizable.

PWP does not support user management, although the developers do provide password protection for a few functions, such as exporting wiki pages to static HTML. Lars Ackermann, PWP’s author, recommends using the web server’s access controls and restricting access to the wiki via the `.htaccess` file.

PWP supports a variety of formatting options: text markups, indentation, tables, and images. Before you can use an image file, you first have to upload it to a special *Uploaded Files* area of the wiki. The software will then let you link to the image from any wiki page. The *Uploaded Files* area can also store files in other formats for public access by any wiki user.

The wiki has a version control system in which it stores all previous revisions of the wiki pages. A diff tool shows the differences between two versions of a page. A full text search against all the pages in the wiki, including text pages uploaded to *Uploaded Files*, gives users searching ability.

PWP will generate reports for wiki pages. You can choose between hierarchical lists, tables, indexes of keywords, or lists with the title and teaser for each wiki page. PWP can create calendars at the push of a button with a page for each month and an overview of the

GLOSSARY

.htaccess: *The .htaccess file gives administrators the ability to control access to data on a web server. You’ll find several tutorials on .htaccess files on the web (e.g., <http://httpd.apache.org/docs/1.3/howto/htaccess.html>).*

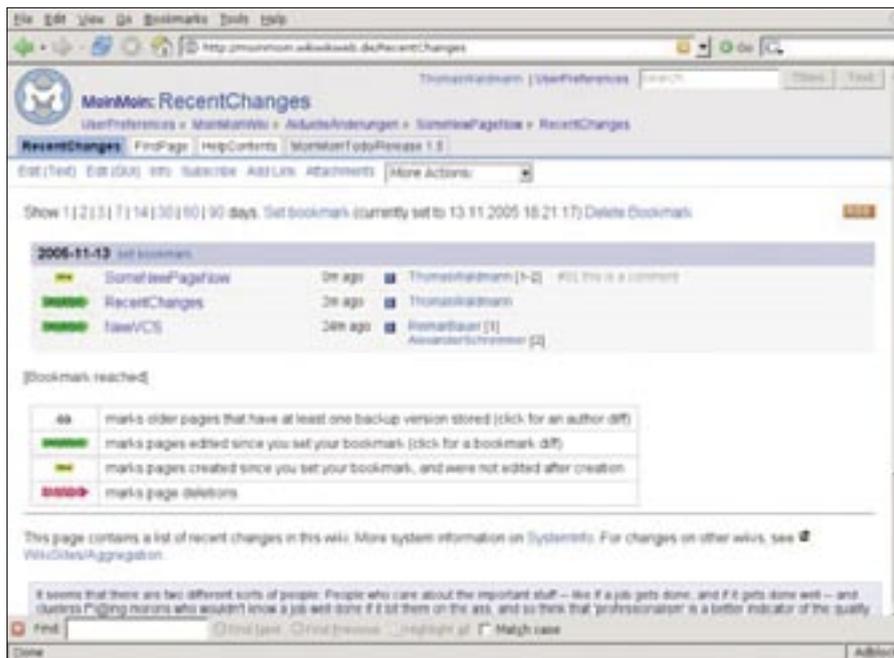


Figure 2: MoinMoin lets you track recent changes.

whole year. In a style similar to Mediawiki, PWP generates a table of contents from primary and secondary headings at the top of the page.

While one author is working on a page, PWP will prevent access by other authors. If needed, PWP will export the whole wiki content to a special directory as static HTML. The online help has comprehensive descriptions of the markups, which again differ from Mediawiki, as well as for installing and configuring PWP.

MoinMoin

MoinMoin [10] is the only Python-based wiki that we will be looking at in this article. As such, it requires a web server with Python support. If you are interested in trying MoinMoin out, there is a desktop version of the wiki, aptly known as Desktop MoinMoin, which runs without a web server at [11]. Another local installation option is to run MoinMoin with the integrated server. This assumes a working Python installation.

A web server is recommended for an open wiki that will need to support simultaneous access by multiple users. The MoinMoin developers describe various deployment scenarios at [12]. The wiki stores articles in directories named after the corresponding wiki pages.

MoinMoin has language support, and an internal feature for changing the language. Internally, the wiki uses UTF 8 to

store texts. Macros and themes let you customize the look; the MoinMoin homepage has a collection of ready-to-run macros.

MoinMoin has integrated user management that supports granular access controls. The program locks pages during editing to prevent competitive editing. The administrator has the ability to rename texts, and in turn to prevent editing of help pages or other texts.

Markups include character and paragraph formatting, colored highlighting of the Python source code, internal and external links (with help texts if needed),

lists, and icons. MoinMoin also supports tables. MoinMoin is the only test candidate to have a WYSIWYG style editor for composing wiki pages. The wiki also supports subpages. One thing worthy of note is that the system supports text entries in DocBook syntax and can even export wiki pages to DocBook.

MoinMoin stores previous versions of a text as backups. The version control system supports access to the last 100 versions, and URLs let you access older versions. You can diff two versions and display the most recent edits.

MoinMoin also provides an RSS feed of the latest edits. The developers give users powerful searching abilities that support page title or full-text searches, multiple operator-linked search keys, and Boolean expressions. The wiki also generate indexes of titles, words, and categories.

The selection of help texts and tutorials is exemplary, with comprehensive, easy to understand howtos for any deployment scenario.

PmWiki

PmWiki [13] stores articles in flat text files; it requires a web server with PHP version 4.1 or newer. The basic setup is just as easy to install as any of the wikis we have looked at thus far: just unpack the source code archive in a directory on the web server and load the index page to launch the wiki. Besides the server-based version, there is also a standalone

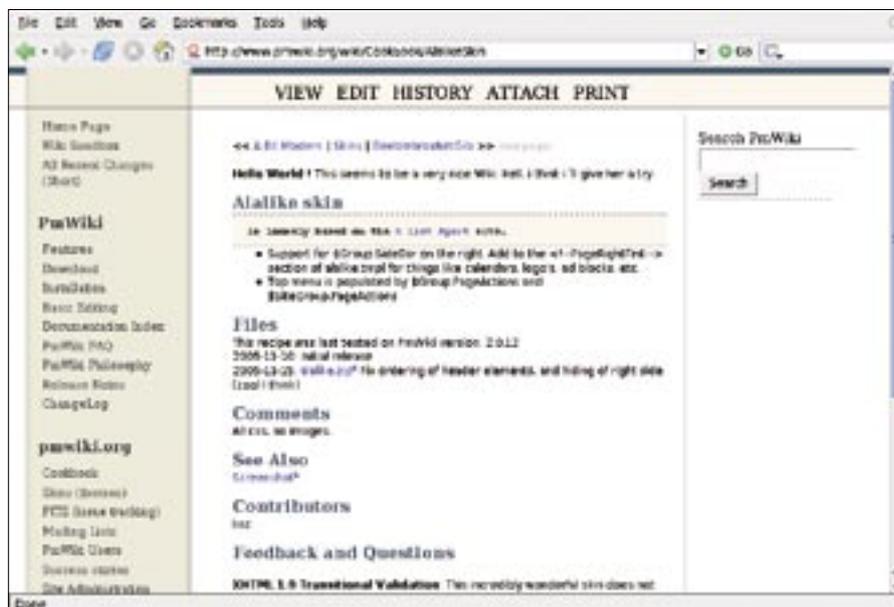


Figure 3: PmWiki is feature-rich and nicely documented.



Figure 4: Dokuwiki is well-suited to documenting technical processes.

variant [14], a useful tool for getting to know the wiki, if installed locally.

PmWiki supports password protection for various areas of the site. You can either prevent access to wiki pages or restrict editing to a specific user group. Granular access controls for the wiki allow the administrator to assign users to groups.

The program has local support. To install it, just download the required language file from [15] to two directories within the wiki, and edit the configuration file. PmWiki supports UTF 8.

You can use CSS or Skins to customize the look. An extension adds support for interactive skin development; in fact, you can use a separate skin for each user. Guest book and forum extensions add value to the user experience.

Formating capabilities for the wiki pages are well up to the task: besides the typical character and paragraph formats, PmWiki supports lists, tables, and image integration. A special extension supports image scaling. PmWiki provides a printed mode preview of articles, in which it hides menus, headers, and footers.

The documentation for the project is very good, with guides and manuals on installing and running the wiki. Incidentally, the guides are referred to as cook-books.

Dokuwiki

Dokuwiki [16] is designed as an open wiki right from scratch. Simply unpack

the source code archive in a directory on your web server, and off you go. If you intend to use the *Access Control List* feature to restrict access to your wiki, modify the configuration files after the install (*conf/dokuwiki.php* or *conf/local.php*), and then define an administrator and corresponding password.

Templates let you customize the look. Configuration is file-based – don't bother looking for a front-end. CSS helps you modify the interface – a reflection of the flexibility of the CSS technology.

Dokuwiki can give you a status report on recent edits via RSS or Atomfeed. And Dokuwiki supports multiple interface languages. To enable a language, just add the language directive to the configuration file – the Dokuwiki documentation provides a comprehensive howto. If you wish to contribute a translation, the Dokuwiki pages give you instructions on doing so. The wiki also has full UTF 8 support for content.

Dokuwiki has excellent formatting support, with text and paragraph formats, tables, and image integration. Like Mediawiki, Dokuwiki supports image scaling with preview thumbnails that users can click to display the larger version. The *Section Editing* function allows users to edit smaller sections of a larger text. The *diff* function displays two versions of a page and highlights the differences.

Full text searching is available, and breadcrumb navigation at the top of the page shows a user the path they have taken through the wiki. Dokuwiki auto-

matically generates tables of content, and the system locks files against editing, once they have achieved a specific status.

Besides comprehensive documentation, the Dokuwiki homepage has a collection of tutorials, covering topics such as syntax extensions and advanced design.

Conclusions

The wikis in this article facilitate the first steps towards setting up and managing your own project. You can use your lean wiki as a simple content management system or a tool for organizing discussions and documents.

PmWiki, MoinMoin, and Dokuwiki all provide user management features that let administrators create groups of authors with distinct privileges, which makes these tools suitable for larger and more structured user communities. The only way to protect the other systems is to use the web server's access controls to protect the directory where the wiki is installed. ■

INFO

- [1] Wikipedia: <http://en.wikipedia.org>
- [2] Mediawiki: <http://www.mediawiki.org>
- [3] Wikkawiki: <http://wikkawiki.org>
- [4] ErfurtWiki: <http://erfurtwiki.sourceforge.net/>
- [5] WikiMatrix: <http://www.wikimatrix.org/>
- [6] Tipiwiki <http://tipiwiki.sourceforge.net>
- [7] Wikepage <http://www.wikepage.org/index.php?wiki=Wikepage>
- [8] Themes for Wikepage: <http://www.wikepage.org/index.php?wiki=Themes>
- [9] PWP: <http://www.lars-ackermann.de/wiki/static/StartPage.html>
- [10] MoinMoin: <http://moinmoin.wikiwikiweb.de>
- [11] Desktop MoinMoin: <http://moinmoin.wikiwikiweb.de/DesktopEdition>
- [12] MoinMoin installation help: <http://moinmoin.wikiwikiweb.de/HelpOnInstalling/>
- [13] PmWiki: <http://www.pmwiki.org/>
- [14] Standalone version of PmWiki: <http://www.pmwiki.org/wiki/Cookbook/Standalone>
- [15] Language files for PmWiki: <http://www.pmwiki.org/wiki/PmWiki/Download>
- [16] Dokuwiki: <http://www.splitbrain.org/projects/dokuwiki>