

## The TWiki wiki and the enterprise

# TEAM BUILDER

A wiki can keep corporate groups on schedule and support collaboration. TWiki is both a leading open-source Wiki for the enterprise and a collaboration platform for building web applications.

BY PETER THOENY

**W**ikis are catching on. Perhaps this is the year of the wiki, as last year was the year of the Blog. wikis are very useful for collaboratively collecting and organizing information. The best known wiki is Wikipedia [1], the online encyclopedia, but many thriving public Wiki sites inhabit the Internet, such as the original WikiWiki-Web [2] by Ward Cunningham.

Can a Wiki be used in the Enterprise? At first, it may seem like a Wiki is too unstructured and too chaotic for a corporate enterprise environment. There is also an – at least perceived – lack of security and no audit trail. Surprisingly, as many early adopters have demonstrated, a Wiki can work very well behind a corporate firewall.

Webmaster-maintained intranets are notoriously outdated. Content is typically incomplete, and finding the content is often a challenge. Also, inconsistencies sometimes exist across departments. Special tools, knowledge, and permission are required to maintain the content. The reason for these problems is sometimes called “webmaster syndrome”: If an employee discovers a page with incorrect or insufficient information, the employee will often ignore the

problem because it takes too much time to find out who the webmaster is and to write an e-mail requesting an update.

The appearance of Wikis marks a paradigm shift from webmaster-maintained content to content maintained by domain experts and casual users. If you move some or all intranet content into a Wiki, readers can browse the content as if it were on an ordinary website, but the readers are also empowered to add information and fix problems on the spot, with no lengthy process for requesting and approving the change. There is no client-side software to install, and everyday users can make changes through an intuitive interface with minimal expense for training and troubleshooting.

Wikis also have the effect of streamlining and simplifying email. Email is a “post and reply” media: you are simply adding stuff, not changing what is already there. The email format is great for discussion, but it is hard to find a “final consensus” on a thread. E-mail is also not hyperlinked and not structured – content can’t be grouped easily into related topics. E-mail and attachments are not version controlled, making it difficult to determine the history of a document. A Wiki can help address some of the short-

### THE AUTHOR

Peter is a software developer with over 15 years experience, specializing in software architecture, user interface design and web technology. Peter is the author of the open source collaboration software, TWiki, and has managed the project over the last five years. He manages the Knowledge Engineering group at Wind River, which has a large TWiki deployment.

comings of email communication. Rather than sending a detailed document by email, users can send a simple email message with a link to a collaborative and version-controlled version of the document within a Wiki.

A wiki is a flexible tool to support workflows of teams. This can be done in the free-form Wiki way, with linked pages that are collaboratively maintained. It can also be accomplished with a structured wiki application that has forms, queries, and reports tailored to the needs of the employees who will actually be using the tool. We see a paradigm shift from programmers creating applications to content contributors with moderate skill creating their own custom applications.

Corporate Wikis have some special requirements. Before implementing an intranet Wiki, enterprise admins want to make sure the tool will provide user support, version control, security, ease of use, and an interface for integrating the tool with existing enterprise applications.

Several Wikis have made their mark in the enterprise (see the sidebar titled "Enterprise Wikis"). This article focuses on TWiki, a leading open source Wiki that also often serves as a collaboration platform for building web applications.

## What is TWiki?

TWiki is a large open source project hosted at TWiki.org [3] with many developers contributing ideas, spec, code and documentation. TWiki's mission is

### Enterprise Wikis

Enterprise-grade open source Wikis include applications such as:

- PHPWiki: A feature-rich implementation with support for various databases (PHP);
- Tiki: A CMS with Wiki, Slashdot-style forums, blogs, image galleries, chat, etc. (PHP);
- TWiki: Heavily featured Wiki variant and application platform for the enterprise, with many Plugins (Perl);
- XWiki: Feature rich Wiki implementation, compatible with some TWiki Plugins (Java);
- ZWiki: A Wiki implementation that runs on the Zope application platform (Python).



Figure 1: The TWiki Interactive environment with (1) Webs, (2) Changes, (3) Search, (4) Jump, (5) Edit, (6) Links, (7) Templates, (8) TWiki forms, (9) Attachments, and (10) Revisions.

to be a leading-edge, web-based collaboration platform targeting the corporate intranet world; to foster information flow within an organization; to let distributed teams work together seamlessly and productively; and to eliminate the webmaster syndrome of outdated intranet content.

TWiki can serve as:

- a shared notebook for projects;
- A departmental collaboration tool;
- An Intranet publishing tool;
- A knowledge base;
- A platform for creating web-based applications like news portals, inventory systems, and issue tracking systems.

TWiki is installed on many web sites, mainly behind corporate firewalls. Companies use TWiki because it is very user friendly compared to some commercial groupware systems like Lotus Notes. To get an idea how companies are using the TWiki platform, read some of the TWiki Success Stories [4] written by authors associated with major corporations such as Disney, Motorola, SAP, and Yahoo!

## Installing TWiki

Go to TWiki.org [5] and download the latest production release of TWiki, then unzip it in a temporary directory on your server. The package contains the follow-

ing directories:

- *bin*: Perl scripts, for CGI-enabled directory;
- *lib*: Perl library files;
- *pub*: Space for attached files, for httpd enabled directory;
- *data*: Page data and log files;
- *templates*: TWiki templates and skins.

The installation is a manual process. Follow the detailed TWiki Installation Guide you find in TWikiDocumentation.html, which is located in the root directory of the TWiki package. It should take an experienced system administrator less than an hour to configure Apache, set the file permissions right, enable authentication, and set up a cron job to send notifications of changes.

## TWiki Highlights

The best way to learn about TWiki is to explore the the user interface. The TWiki environment contains *webs* and *topics*. A web is like a Wiki within TWiki. Each web is a container for topics. A topic is one page that can be manipulated by the users. Figure 1 shows the topic *GnuSkinGoButton* located in the *Support* web at *TWiki.org*. Note that the web and topic names are reflected in the URL; in our example, the URL is <http://twiki.org/cgi-bin/view/Support/GnuSkinGoButton>. The numbers in Figure 1 refer to TWiki

**Advertisement**

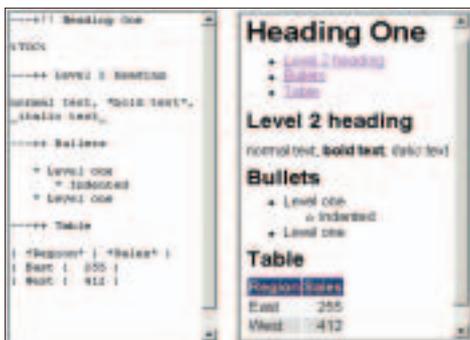


Figure 2: TWiki markup example.

features you will learn about in this section.

TWiki webs (see (1) in Figure 1) are referenced in the navigation bar on the left side of the topic. Each department or group can get its own web, each with its own namespace. The Main web and TWiki web are special. The Main web is the primary web containing all registered users and groups. The TWiki web contains all TWiki documentation and configuration settings.

Team members can see the recent changes in a web by clicking on the Changes link (see (2) in Figure 1). It is also possible to subscribe to a web in its WebNotify topic, in which case you get a daily e-mail with links to topics that changed in the web. There is also an RSS feed [6] for those who prefer to read news via RSS. All three methods allow team members to stay up-to-date, review, fix, and enhance content continuously.

TWiki has an interactive search screen that searches topics in the current web by default (see (3) in Figure 1). For power users, there is also an advanced search screen where the user can enter arbitrarily complex search strings with regular expressions. TWiki also has variables [7] to compose content dynamically. For example, a search can be canned and embedded into a topic with a `%SEARCH{...}% variable` [8]. This is typically used by web applications, where the search string is supplied by a URL parameter via a `%URLPARAM{"search"}% variable`.

The Jump box (see (4) in Figure 1) provides quick navigation within the TWiki space. If you want to jump to the subscription topic, for example, type the name of the topic `WebNotify` into the Jump box and hit the enter key. If you know there is a topic

about “goals,” but you do not remember the exact name, type goals to get a list of all topics that have “goals” in the name. If you want to quickly jump to the Marketing web, type Marketing. (including the dot) to jump to the Marketing web (assuming there is a web of that name).

As in any other Wiki, the most important link is the “Edit” link on every page (see (5) in Figure 1). The Edit link invites everyone to contribute to the topic. TWiki has a markup system that is easy to learn (see Figure 2).

Writing a Wiki page is like writing an e-mail – just do it and don’t worry too much about the formatting (the underscore renders the text in italic). There is currently a WYSIWYG editor [9] integration in the works. Topics are locked during an edit session. If someone else tries to edit a topic you are currently working on, she will get a warning that you are currently editing the topic.

Cross-linking related content is a key feature for effective collaboration. (See (6) in Figure 1). Our brain works with associations; we often need to hear or remember a keyword in order to bring back related material. This is how a Wiki works too. People are invited to create many links to related content via WikiWords. WikiWords, also known as CamelCase words, are capitalized words, run together, such as `WebCollaboration` or `IntranetTools`. To create a link to a topic you know exists, type its WikiWord, as shown in the following example:

For more info, see `IntranetTools`

When you save the topic, the WikiWord automatically turns into a link:

For more info, see [IntranetTools](#)

Notice that, unlike HTML, you do not need to specify an anchor link. TWiki

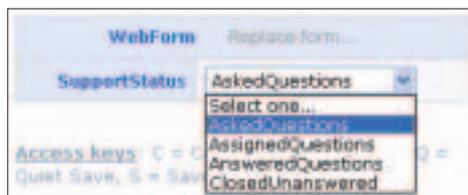


Figure 3: TWiki Form in edit mode with a pick-list.

does that automatically for you when it recognizes a WikiWord. What happens if I type a WikiWord but it is a NeverUsedWord? In other words, a topic of that name does not exist. The system recognizes the WikiWord and invites you to create the topic by placing a question mark link behind the word. To create that topic, click on this link, type in some text, and save it. If you return to the originating topic and hit Refresh, you will see `NeverUsedWord` turned into a regular link.

TWiki is template driven. (See (7) in Figure 1.) When you create a new topic by clicking on a question mark link, you will see some default text, typically just a blank screen with a dated signature at the bottom. This is defined by the system wide template topic `WebTopicEditTemplate` located in the TWiki web. To get a customized template for a web, create a topic of the same name in that web. A form-based web application needs to create topics based on a custom template. This can be done with an HTML form embedded into a TWiki topic that has an input tag named `templatetopic` supplying the name of the template topic [10].

By adding form-based input to free-form content (see (8) in Figure 1), you can structure topics with unlimited, easily searchable categories and introduce workflow-based applications.

When forms are enabled for a web and selected in a topic, a single form appears in edit mode below the big topic edit area. In other words, you get the unstructured free form Wiki content on top, and structured content with edit fields, radio buttons, checkboxes, and pick-lists at the bottom of the edit screen. The form contents are rendered as a table when viewed on the display

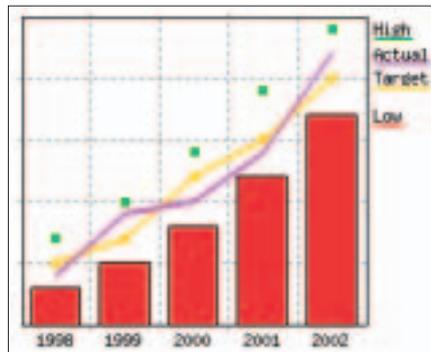


Figure 4: Resulting chart for the data in Listing 1.

## Listing 1: ChartPlugin example

```
01 %TABLE{name="exampleTable2"}%
02 | *Year* | *1998* | *1999* | *2000* | *2001* |
   | *2002* |
03 | Low    | 8    | 10   | 13   | 17   | 22   |
04 | Target | 10   | 12   | 17   | 20   | 25   |
05 | High   | 12   | 15   | 19   | 24   | 29   |
06 | Actual | 9    | 14   | 15   | 19   | 27   |
07
08 %CHART{ type="combo" subtype="bar, pline, point,
   line" name="combo2" table="exampleTable2"
   data="R2:C2..R5:C6" xaxis="R1:C2..R1:C6" legend=
   "R2:C1..R5:C1" width="225" height="200" }%
```

page. For example, the support tracker application on TWiki.org uses a TWiki Form for workflow, as follows:

1. Someone submits a form in the Support web, asking a support question [11].
2. The application creates a new topic with the form content. The topic also has a TWiki Form with a form field named *SupportStatus* with value *AskedQuestions*. This indicates that this is a new question.
3. All topics with status *AskedQuestions* are listed in the Support home. This is done with an embedded `%SEARCH{}%`.
4. Someone looks at the asked questions and decides to work on one. She edits the topic. At the bottom is the TWiki form showing a pick-list with *AskedQuestions* selected (see Figure 3). She changes that to *AssignedQuestions* and adds some content to the topic.
5. The Support web home now lists this topic in the *Assigned* list.

6. Once she has found an answer, she edits the topic again and picks *Answered Questions* for the status.

7. The Support web home now lists this topic in the *Answered* list.

For more on the process of creating an application based on TWiki Forms, see the TWiki Forms documentation [12].

Each topic can have one or more files of any type attached to it by using the Attach screen [13] to upload (or download) files from your local workstation (see (9) in Figure 1). File attachments can be used to create powerful customized groupware solutions, like file sharing and document management systems.

In our Support web example, a patch file attached to a support topic might be the answer to a question. That is, you can organize content in a logical way, with files attached to a topic. You do not need to worry about broken links to an FTP site when content gets restructured.

## Advertisement

## Listing 2: Status board iteration 2 raw text

```
01 | *Start* | *End* | *Primary* | *Backup* |
02 | 07:00am | 11:00am | RichardDonkin | |
03 | 11:00am | 03:00pm | PeterThoeny | |
04 | 03:00pm | 07:00pm | SamHasler | |
```

All topic content and all attachments are under revision control (see (10) in Figure 1). You always know who changed what, and you can retrieve older content at any time. Revision control is transparent to the user. It happens automatically; you do not see a check-in dialog. A special twist to the version control feature is that you can edit and save a topic as many times as you like, and no new revision is created if your work falls within a one hour window. This is a handy feature to fix typos and make more changes.

TWiki also offers access control [14]. You can restrict access to single topics and entire webs, by individual user and by groups, in three areas: view, edit, and rename. Access control lets you easily create and manage an flexible, fine-grained privilege system.

Open, free-form editing is the essence of Wiki culture. You should make decisions to restrict read or write access to a web or a topic with care – the more restrictions, the less Wiki in the mix. Experience shows that unrestricted write access works very well because:

- Peer influence is enough to ensure that only relevant content is posted.
- Peer editing – the ability for anyone to rearrange all content on a page – keeps topics focused.
- Content is transparently preserved under revision control which provides a "soft security". Edits can be undone if needed. Users are encouraged to edit and refactor (condense a long topic).

Plan the roll-out and content before you announce your TWiki:

- Select an area where TWiki can help most by using it as a pain killer (users feel relieved), not as a vitamin (good to have for prevention).
- Build the initial structure and populate some initial content with help from early adopters.
- Do the initial roll-out for a smaller group. Other teams will see how it works and try it out for their needs.
- Train and coach the users. Point new users to WelcomeGuest [15] and give the ATasteOfTWiki presentation [16].

## TWiki Plugins

TWiki Plugins enhance the functionality of TWiki. Users do not need to install anything; plugins are installed by the administrator on the server where TWiki is installed. You'll find a growing Plugin repository [17] on TWiki.org – over 150 Plugins are available for download. Plugins are a great way for administrators and web developers to tailor TWiki to their needs. Some popular plugins include:

- *ActionTrackerPlugin*: lets you keep track of action items in meeting minutes and notify assignees by e-mail;
- *CalendarPlugin*: creates monthly calendars with events;
- *ChartPlugin*: lets you create charts visualizing TWiki tables;
- *DatabasePlugin*: provides access to a relational database;
- *EditTablePlugin*: lets you edit TWiki tables using edit fields, date pickers, and dropdown boxes;
- *HeadlinesPlugin*: shows news based on RSS feeds;

- *TablePlugin*: controls attributes of tables and supports sorting of table columns;
- *TWikiDrawPlugin*: lets you use a Java Applet to create editable drawings embedded in topics;
- *SlideShowPlugin*: turns a TWiki page into a web-based presentation.

*SpreadsheetPlugin* is a pre-installed plugin with spreadsheet capabilities. Over 60 formulae are available, such as \$AVERAGE(), \$IF(), \$REPLACE(), \$TIME(), \$SET(), \$GET(). A formula can be anywhere in a topic, not just in a table.

## INFO

- [1] Wikipedia: [http://en.wikipedia.org/wiki/Main\\_Page](http://en.wikipedia.org/wiki/Main_Page)
- [2] WikiWikiWeb: <http://c2.com/cgi/wiki?WelcomeVisitors>
- [3] TWiki.org: <http://twiki.org/>
- [4] TWiki Success Stories: <http://TWiki.org/cgi-bin/view/Main/TWikiSuccessStories>
- [5] TWiki Download: <http://twiki.org/download.html>
- [6] RSS feed documentation: <http://twiki.org/cgi-bin/view/TWiki/WebRssBase>
- [7] TWiki variables: <http://twiki.org/cgi-bin/view/TWiki/TWikiVariables>
- [8] TWiki %SEARCH% variable: <http://twiki.org/cgi-bin/view/TWiki/TWikiSearch>
- [9] Kupu WYSIWYG editor integration: <http://twiki.org/cgi-bin/view/Plugins/KupuEditorAddOn>
- [10] TWiki Templates: <http://twiki.org/cgi-bin/view/TWiki/TWikiTemplates>
- [11] TWiki.org Support web: <http://twiki.org/cgi-bin/view/Support/WebHome>
- [12] TWiki Forms : <http://twiki.org/cgi-bin/view/TWiki/TWikiForms>
- [13] File attachments: <http://twiki.org/cgi-bin/view/TWiki/FileAttachment>
- [14] Access control: <http://twiki.org/cgi-bin/view/TWiki/TWikiAccessControl>
- [15] Welcome Guest: <http://twiki.org/cgi-bin/view/TWiki/WelcomeGuest>
- [16] A Taste of TWiki presentation: <http://twiki.org/cgi-bin/view/TWiki/ATasteOfTWiki>
- [17] TWiki.org's Plugins repository: <http://twiki.org/cgi-bin/view/Plugins/WebHome>
- [18] TWiki installations directory: <http://twiki.org/cgi-bin/view/Main/TWikiInstallation>

### Listing 3: Status board, raw text iteration 3

```
01 %EDITTABLE{ header="on" format="| select, 1, , 07:00am, 07:30am,
08:00am, 08:30am, 09:00am, 09:30am, 10:00am, 10:30am, 11:00am,
11:30am, 12:00pm, 12:30pm, 01:00pm, 01:30pm, 02:00pm, 02:30pm,
03:00pm, 03:30pm, 04:00pm, 04:30pm, 05:00pm, 05:30pm, 06:00pm,
06:30pm, 07:00pm | select, 1, , 07:00am, 07:30am, 08:00am, 08:30am,
09:00am, 09:30am, 10:00am, 10:30am, 11:00am, 11:30am, 12:00pm,
12:30pm, 01:00pm, 01:30pm, 02:00pm, 02:30pm, 03:00pm, 03:30pm,
04:00pm, 04:30pm, 05:00pm, 05:30pm, 06:00pm, 06:30pm, 07:00pm |
select, 1, , ArthurClemens, ColasNahaboo, PeterThoeny, RichardDonkin,
SamHasler, SvenDowideit | text, 20 | text, 40 |" changerows="off" }%
02 | *Start* | *End* | *Primary* | *Backup* | *Comments* |
03 | 07:00am | 11:00am | Main.RichardDonkin | | |
04 | 11:00am | 03:00pm | Main.PeterThoeny | | |
05 | 03:00pm | 07:00pm | Main.SamHasler | | |
```

| Start   | End     | Primary            | Backup | Comments |
|---------|---------|--------------------|--------|----------|
| 07:00am | 11:00am | Man Richard Boskin |        |          |
| 11:00am | 03:00pm | Man Peter Poonny   |        |          |
| 03:00pm | 07:00pm | Man Sam Heeler     |        |          |

Save table    Clear table    Cancel

Figure 5: Status board iteration 3 in edit mode.

For example, `%CALC{"$PROPER SPACE(LinuxMagazine)"}%` returns Linux Magazine. Web applications typically use the `$IF()` formula to output text conditionally.

The ChartPlugin creates PNG or GIF charts to visualize TWiki tables using a default linear scale or an optional semi-logarithmic scale. Listing 1 results in Figure 4.

### Simple Web Application

Picture this scenario: The manager of the customer support call-center keeps track of the allocated times of the support engineers with a spreadsheet. A printout gets posted on the wall for all to see. He asks you if you can build an electronic version of the call-center status board. Everyone

should be able to see who is on call at what time, and it should be easy to change the status board.

A Wiki application is typically done in iterations. Start simple and enhance incrementally. In the first iteration, we create a topic with simple bullets:

```
* 07:00am - 11:00am: Richard
* 11:00am - 03:00pm: Peter
* 03:00pm - 07:00pm: Sam
```

The topic fulfills the requirements; everyone can see the status board and maintain it. Let's make it more useful by converting the bullets into a table, and using WikiWord links to team member's home pages for easy reference. Let's also add a column for a backup person (see Listing 2).

Now let's make it more user friendly. We use the EditTablePlugin so that the users can select the times and names from a list (see Listing 3). The view

mode and edit mode results for the raw text in Listing 3 are shown in Figure 5 respectively.

The call-center status board is a very simple application that does exactly what the customer needs. You can create more sophisticated applications with TWiki forms, embedded search, and some HTML to glue everything together. One example is the TWiki installations directory [18]. All installations are listed in a big table. A filter can be applied to list, for example, only corporate installations behind a firewall. You can borrow the source for your own applications. Hint: To view the raw text of a TWiki topic, append `?raw=on` to the URL.

### Conclusion

TWiki is a powerful platform for web collaboration and web development. The TWiki platform supports free form content sharing in Wiki style. Plan your Wiki rollout carefully, and provide training for users. You will see an explosion of interest after people "get it." ■

## Advertisement