Chapter 2
Quick Install

The goal of this chapter is to guide you through a simple DokuWiki installation from start to finish. We'll start by getting your environment ready. Then we'll download and install on a development server, examine the user interface, and create and save your first wiki page.

One of the features that attracted me to DokuWiki is the ease of installation. This chapter is over 20 pages long, but the installation itself only takes a few minutes. I suggest that you do a 'throwaway' install first - run through all the steps in the first ten chapters of this book, using a test site that you can get rid of when you're done experimenting. DokuWiki's architecture makes it easy to do so - all you have to do is delete the folder containing the test site. No cleaning up databases or running uninstall routines that leave cruft around if they don't complete successfully.

What You'll Need

One characteristic of the experienced craftsman is that he lays out his tools and determines what else he needs before he begins working. No running to the hardware store halfway through a job because he didn't have a 5/16 hex head wrench! Similarly, here I'll explain what you need in order to get started with DokuWiki - both in terms of hardware and software as well as knowledge and experience prerequisites.

Environment

First, let me describe what my environment looks like. I've got a fairly varied environment; while it is unlikely your environment is exactly like mine, there is very probably one machine laying around here that is similar to yours. And most hardcore developers love taking a peek over the shoulder of another developer to see how they've got their stuff set up.

I have four machines on my desk - two running Linux Fedora, one running Mac OSX 10.4, and the last using Windows XP. All are running Firefox 2.x or 3.x. By the time I finish this scribbling, I might have migrated all of the boxes to Firefox 3, one of them to a flavor of *buntu, and maybe another to one of those builds of Fedora 11 that is starting to peek out from it's hiding place. All of these clients are hooked up to a Red Hat Linux file server; my development websites are all under a '/Sites' folder on that file server - no data on local machines for me!

The Web server software (Apache and IIS) running on each local machine is pointed to the root of /Sites on the file server. I can access my development folders via http://localhost/development_site_1 and http://localhost/development_site_2 from any of my workstations. Should you choose to put your development websites on your local machine, you'll see no difference - the only change I had to make to serve data from the file server was to point the Web server to the file server folder structure instead of the local machine's /var/www/html or \InetPub\wwwroot folder.

Anyways, one of the Fedora boxes is my primary workstation, and that's what I'm working on. For those of you using Windows, I'll note differences when they are relevant. Most of the time, there won't be anything significant to mention.

Requirements

In order to follow along, you'll need three things.

First, access to a box with a Web server that supports PHP installed on it. Apache is recommended. Visit www.apache.org, for more information.

Second, PHP 4.3 or higher installed on the server. PHP 5 is highly recommended due to security issues. The wiki, http://www.dokuwiki.org/install:php, has a number of notes about specific PHP configuration items. Additionally, cruise by www.php.org, for more information on PHP configuration in general.

And, third, a reasonably modern browser. I use Firefox and have had no problems. The wiki:compatibility page has a long list of browsers together with what works and what's broken, and how badly. In general Firefox, Konqueror, and the latest versions of Opera all work fine; IE has some quirks (generally appearance-oriented - buttons too wide in IE7, for example), and some of Safari's versions have some minor issues.

Unlike installation mechanisms for other wikis, as long as you have access to an FTP client or a file manager where you are going to put your DokuWiki files, you'll be fine. You don't have to run scripts or batch files or otherwise have root access. You may have to change permissions on some folders, but most shared hosts provide a mechanism to do so. If yours doesn't provide that minimal functionality, perhaps it's time to find a new host.

Do I need to be a PHP programmer?

DokuWiki is written in PHP. Do you need to know how PHP works? Heck no.

You can go a long way with DokuWiki without knowing a line of PHP code. But it's helpful if you know a little - at least enough to be dangerous. You won't necessarily write any code, but being able to look at a set of PHP files and understand generally what's going on would help you be a better administrator. Here are a few things you should know about PHP.

PHP is an add-on to a Web server. When the Web server receives a request for a page with a .php extension, the server sends that page through the PHP interpreter, which processes

it, creates output and turns the output over to the Web server which, in return, sends the output (plus, perhaps, more stuff) back to the browser of the requesting visitor.

PHP files are just text files, modifiable in any text editor. PHP statements and functions have to be enclosed with a pair of delimiters that the Web server and the PHP interpreter recognize. Generally, each statement or function is terminated with a semi-colon.

```
<?php
some_php_statement_or_command;
some_php_function();
?>
```

PHP can be embedded in an HTML code block, like so:

```
*** my_first_php_page.php
<body>
Hello World!"
<?php
some_php_statement_or_command;
=some_php_function();
?>
<b>More basic HTML, in bold</b>
<?php
another_php_statement;
=another_php_function();
?>
</body>
```

Note that these examples are just descriptive, not functional A couple of working PHP programs are listed in the next section. For files that contain only PHP code, the closing tag ("?>") is strongly discouraged. It is not required by PHP. Not including it prevents trailing whitespace from being accidentally injected into the output.

Do I need to know how to administer a Web server?

Again, no. However, knowing a couple of things about administering a Web server will make your life easier. How many times have we heard Doc McCoy exclaim to Kirk, "Dammit Jim, I'm a programmer, not a systems administrator!" Well, OK, we never did, but if he was in our shoes, he would have. There are a few concepts of how the Web server works that will help you in the basic setup and configuration of DokuWiki that I'll discuss now: home directories, default documents and virtual folders.

First, there is the concept of a home directory. A Web server is simply a piece of software that waits for requests for files from a visitor (via their browser). Once the Web server receives a request, it'll look in it's home directory ('root folder') for that content. This home

directory is identified in the Web server's configuration. On IIS, this folder is c:/inetpub/wwwroot while on Apache on Red Hat/Fedora it's /var/www/html.

Second is the concept of a default document. If the user explicitly names a file in a Web browser address bar, that file is delivered, like so:

```
http://www.example.com/somefile.htm.
```

So how does a Web server dish up a home page when you just navigate to 'www.example.com'? Enter the default document. Unbeknownst to the vast majority of Web surfers, a Web server is configured to automatically look for files with specific names if not otherwise instructed. When you install IIS, IIS's default document is automatically defined as 'default.htm' while Apache's default document out of the box is 'index.html'. These can be changed just like the root folder can be.

Furthermore, you can have multiple default documents defined - and in a specific order of priority; if one isn't found, the Web server will automatically look for the next document, and then the next, and so on. This is particularly handy when you want to have a PHP script be the default document. Setting 'index.php' as your highest-priority default document means that if you have an 'index.php' file in your document root, it'll automatically be executed. If 'index.php' doesn't exist, then the next default document, say, 'index.html', will be. DokuWiki takes advantage of this functionality, as you'll see shortly.

Third is the concept of virtual folders. When entering a URL, the visitor can include a path elsewhere to specify a subdirectory under the root folder. For example, a visitor might navigate to

```
http://www.example.com/customer_one/sales_presentations/listing.htm
```

to go to "Customer One's" listing of sales web pages while

```
http://www.example.com/customer_two/financial_results/index.htm
```

would do the same for Customer Two for the financial results. Web servers can be configured with 'virtual folders' that provide shortcuts to folders deep in the directory structure.

A virtual folder of 'onesales' might be mapped to 'customer_one/sales_presentations', so a visitor could enter

```
http://www.example.com/onesales/listing.htm
```

or

```
http://www.example.com/c2fin/index.htm
```

Furthermore, virtual folders can be mapped to their own URLs. Thus, visitors to Customer One's files might actually just navigate to

```
http://www.cust_one.com
```

and visitors to Customer Two's files would enter

```
http://www.cust_two.com
```

in their browser, none of them being any the wiser that they're all ending up somewhere deep within www.example.com's Web site.

We'll take advantage of these capabilities to create multiple wikis on the main Web site.

Windows vs. Linux

If you're a Windows developer, you may be disconcerted by the great-big, beautiful world outside the grasp of Redmond. This software, and this book, are a couple of examples. Ne'er fear - most open source software is built on the Linux platform, and if you're interested in open source, you need to learn a bit about Linux. In fact, getting involved with an Open Source application like DokuWiki is an excellent way to dip your toe into the Open Source world without getting completely wet.

This book is being written on a Linux workstation, and DokuWiki is being installed and run on a Linux workstation, a Linux file server, and a Linux Web server. As a result, Windows folks might be a bit confused by some of the nomenclature and conventions they run into. Here are a couple of quick tips to help you navigate through some of the differences.

First, the forward slash indicates a folder. Why Microsoft programmers decided to use the backslash - the character used to initiate escape sequences in the UNIX world a decade before any 'softie had finished high school - to separate folders is beyond comprehension. We'll never know, and I digress. In Linux, folder hierarchies are separated with a forward slash.

Files that begin with a period are hidden from normal view. These are typically system or configuration files. I like to know what's going on so I keep the "Show Hidden Files" option in my file manager checked and that's what you'll see in screen shots in this book; the next Linux system you look at might not.

Linux doesn't rely on extensions to the extent that Windows does. The operating system knows what files are without depending on the extension. Thus, COPYING and README and VERSION (as well as .htaccess) are all understood by the file manager to be plain text files.

Preparing a Local Machine

One of the best parts of DokuWiki (and there are lots of them!) is the install process. After reading through this chapter a couple of times, if you can't install a new DokuWiki wiki and have it ready for testing in five minutes, you're just not paying attention.

Installation onto a local machine consists of eight quick steps. The first three are to prepare a location on your machine, make sure your Web server is running and make sure it's capable of running PHP. (The last five, covered in the next section, are to download, extract, copy, install, and run.) The installation topic on the DokuWiki wiki inserts another step in this process - handling security. I do basic security set up here but not in detail, because we'll deal with security in great detail soon enough, and it isn't critical to dive in with both feet now on a local machine. If you are feeling brave and installing on a production (outward facing) machine right away, then you must handle security during configuration.

Let's lay some groundwork before we jump into installation. To wit, let's prepare a place for our wiki before we start installation.

Web site files location

If you've already built Web sites, you can probably skip this step, but you may want to read it anyway, just to make sure we're all using the same language and you understand the conventions I'm following. Since Web sites are like potato chips - no one can have just one - create a place where all of your Web sites will reside. I have a folder on my file server named "Sites" (clever, huh?). The Web server's root folder points to this /Sites folder. Under the /Sites folder, there are for each Web site:

```
/Sites/AlgorithmHeaven
/Sites/Hentzenwerke
/Sites/HentzenHause
/Sites/SoftwareMuscle
/Sites/WhilHentzen
```

and so on.

Since a customer Web site goes on my consultant's Web site, www.whilhentzen.com, I create folders for each customer under my main site, like so:

```
/Sites/WhilHentzen/cust_one
/Sites/WhilHentzen/cust_two
/Sites/WhilHentzen/cust_three
```

The wiki for Customer One goes in the cust_one folder, the wiki for Customer Two goes into cust_two, and so on.

You'll see in a moment that navigating to one of the customer's Web sites is then just a matter of

```
http://localhost/WhilHentzen/cust_one
```

because localhost will be pointed to /Sites when we configure the Web server.

DokuWiki files location

Next, you'll want a place for the DokuWiki files you're going to be using. DokuWiki comes in a single compressed file, but you'll most likely want to incorporate templates that alter the appearance (and, to some extent, the functionality) of DokuWiki, and plugins that provide additional functionality. So you may find yourself with a whole slew of files before you know it.

I have a folder named 'dw_masters' that I keep in my /Sites folder, alongside of the folders for each Web site.

```
/Sites/AlgorithmHeaven
/Sites/dw_masters/dokuwiki_source
/Sites/dw_masters/plugin_one
/Sites/dw_masters/plugin_two
/Sites/dw_masters/template_one
/Sites/HentzenHause
/Sites/HentzenWerke
/Sites/SoftwareMuscle
/Sites/WhilHentzen
```

and so on. Note that dw_masters only resides on my development Web server, for ease of access. I **don't** copy it to the production Web server!

Install Web server software

Second, let's get the Web server installed and configured on your local machine. The default choices for Web server software is Apache for Linux machines and IIS (Internet Information Server) for Windows, although there is a version of Apache that runs on Windows.

Windows

If you're using Windows, you can test to see if IIS is installed and working by typing "http://localhost" in your browser. If you get a page like shown in **Figure 1**, you're all set. *\|\|\|

***Figure 1**. Localhost on IIS.*

If you get a page like shown in **Figure 2**, IIS is either (1) not installed, or (2) installed but not running.

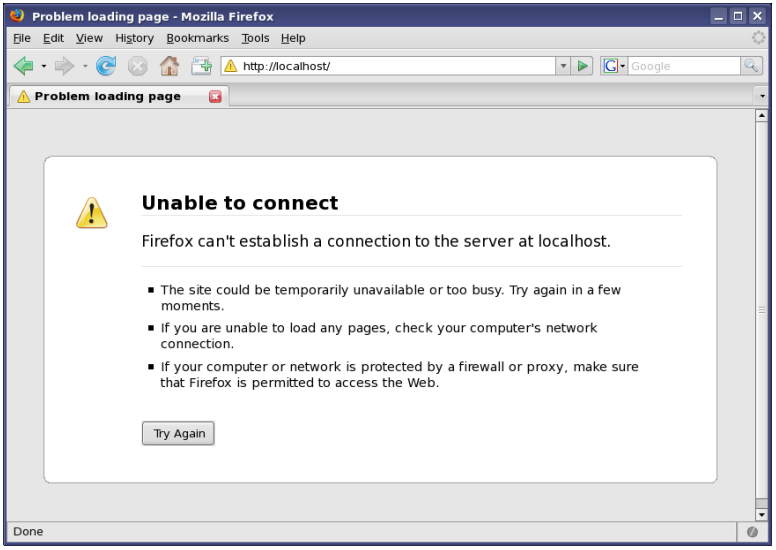

***Figure 2**. The Web server software is either not installed or not running.*

Installing IIS

If you need to install IIS, here's how. Open Control Panel, Add/Remove Programs, Windows Components, scroll down to IIS, and highlight IIS, as shown in **Figure 3**.

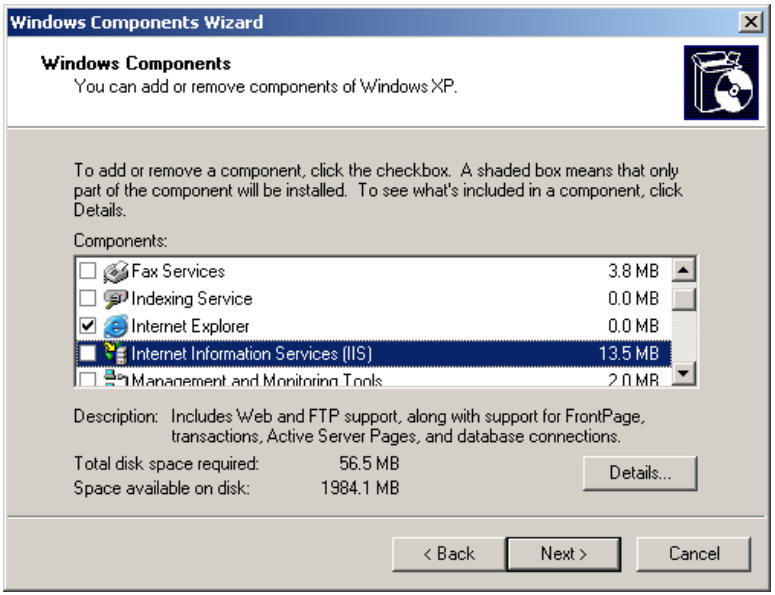

***Figure 3**. Selecting IIS in the Windows Component Wizard.*

Select the checkbox and click the Next button to install it.

Next, point IIS to the \Sites folder, wherever you've got it located. Do this via Control Panel | Administrative Tools | Internet Information Services as shown in **Figure 4**.

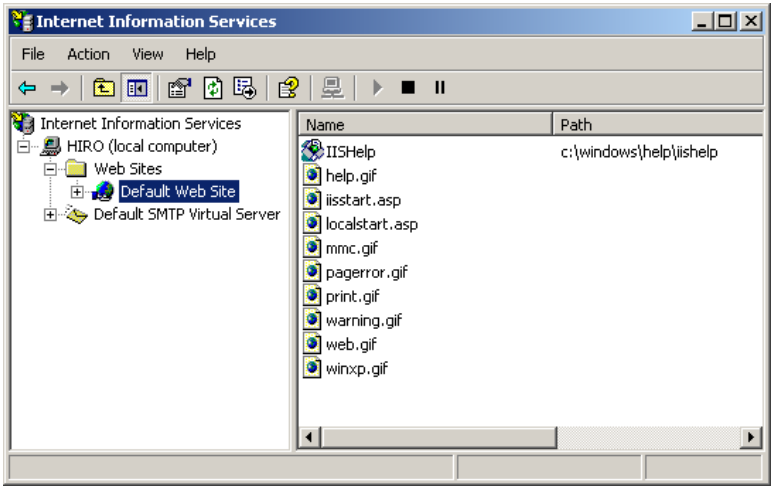

***Figure 4**. The IIS dialog.*

Right click and select Properties, then click on the Home Directory tab. Change the Local Path to the location of '\Sites', as shown in **Figure 5**.

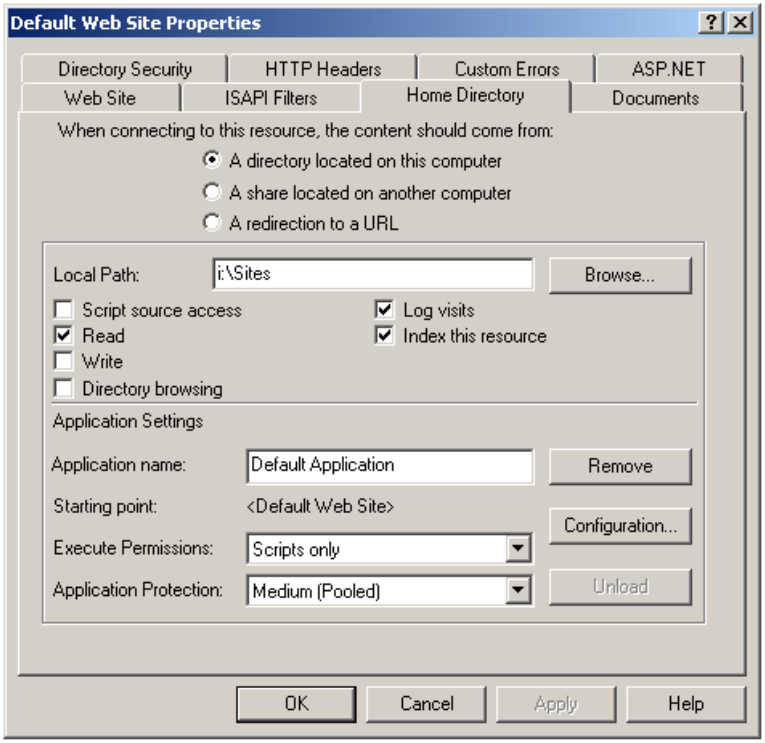

***Figure 5**. Changing the home directory in IIS.*

Finally, click on the Documents tab, add 'index.php' and 'index.htm' as the default documents, and delete the other three, as shown in **Figure 6**.

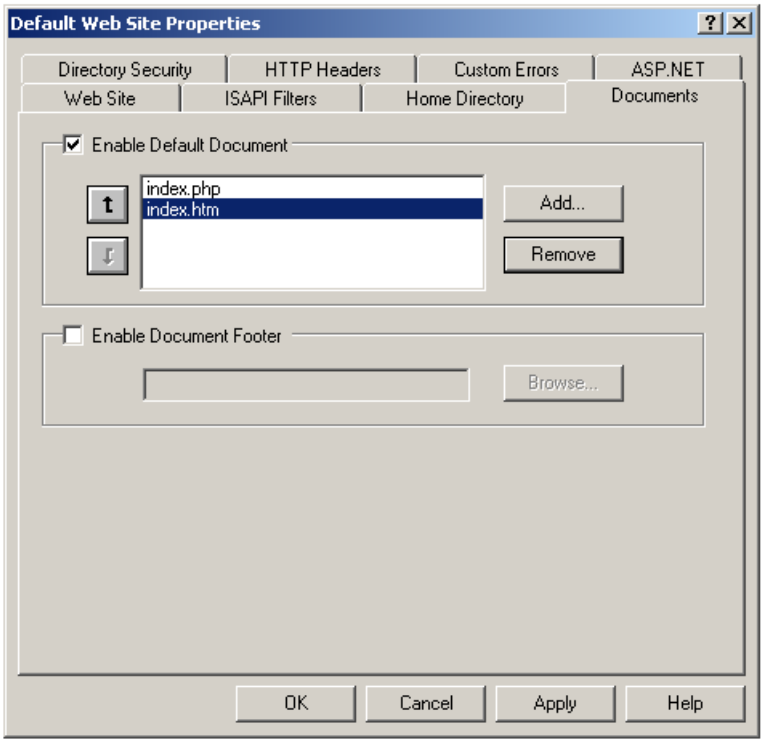

***Figure 6**. Setting the default documents in IIS.*

Click 'Apply' and you're all set.

Linux

Similar to Windows, if you're using Linux, you can test to see if Apache is installed and working by typing "http://localhost" in your browser. If you get a page as shown in **Figure 7**, you're in business.

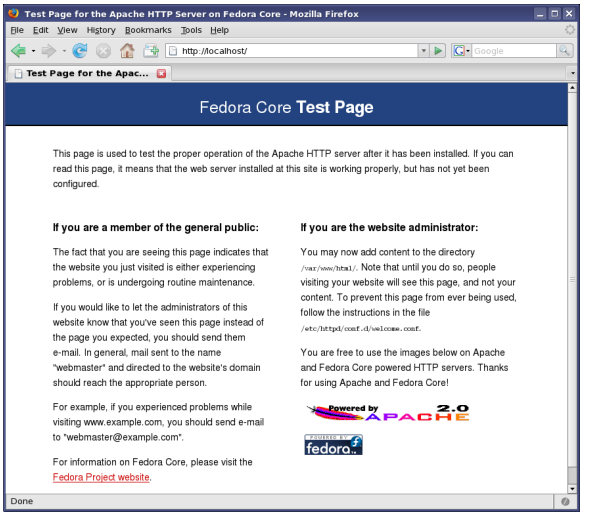

***Figure 7**. Localhost on Apache.*

If you get a page like Figure 2, Apache is either (1) not installed, or (2) installed but not running. To see if it's installed and/or running depends on the Linux distribution you're using. Most distributions have a "Services" applet (in Fedora, it's found under Administration | Server Settings | Services) that looks like **Figure 8**.

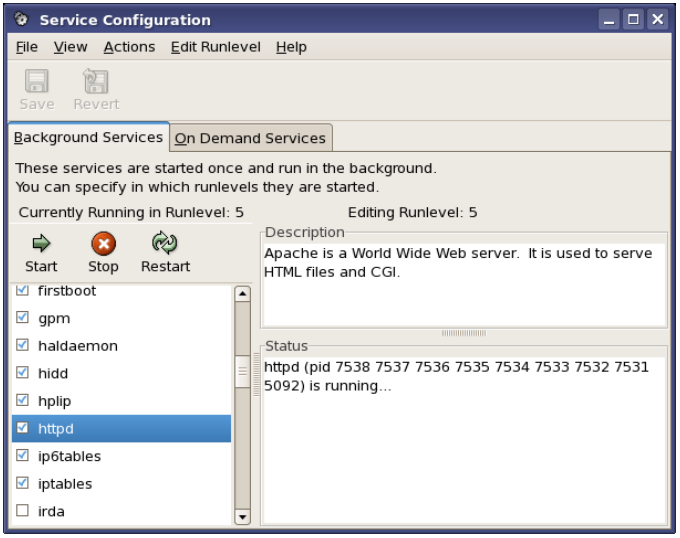

***Figure 8**. Configuring Apache on Linux.*

If Apache is not installed, there will not be a 'httpd' item in the list on the left. If Apache is installed but not running, the 'httpd' item on the left will be displayed but the entry in the edit

box on the right will say "httpd is stopped". (The checkbox in front of the 'httpd' name indicates whether the daemon is configured to automatically start when the system is booted. The httpd daemon may be configured to start automatically but may currently be stopped for one reason or another.)

Depending on the distribution of Linux you're using, installing Apache may be as simple as running the System | Add/Remove Software menu option and selecting the Web server, as shown in **Figure 9**, or as complicated as downloading a tarball and doing a manual install.

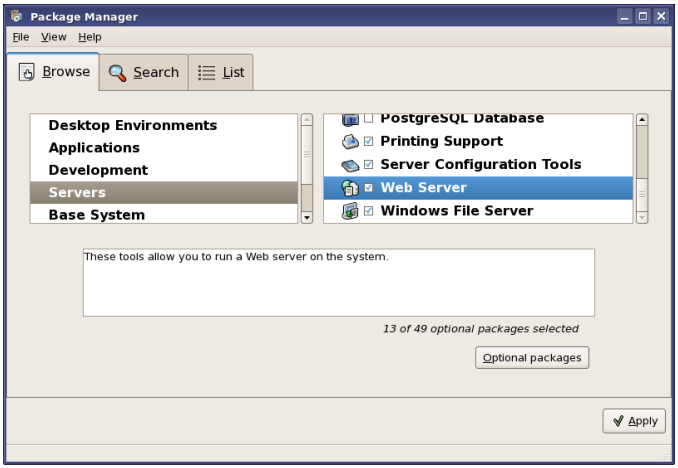

***Figure 9**. Installing Apache on Linux using the Fedora Package Manager.*

Fedora, SuSe, Ubuntu, Mandriva, and Mepis are popular desktop distribution that all provide fast and easy access to installing Apache.

Once installed, you'll need to turn it on using the 'Services' daemon. See Figure 8 again. Select the 'httpd' entry in the list, then click the "Start" button on the top to start the httpd daemon. If you want the Apache server to start automatically each time the box is booted, click the checkbox to the left of the 'httpd' name - and then click the Save button at the very top of the dialog (or select File | Save from the menu) in order to save the new configuration.

Next, point Apache to the /Sites folder, wherever you've got it located. Do this via an edit of the /etc/httpd/conf/httpd.conf text file on your local machine (the box running Apache, not the box hosting the files.)

```
#DocumentRoot "/var/www/html"
DocumentRoot "/home/whil/fileserver/Sites"
```

and

```
#<Directory "/var/www/html">
```

```
<Directory "/home/whil/fileserver/Sites">
```

Finally, make sure that 'index.html' and 'index.php' are included in the list of default documents. To do this with Apache, edit /etc/httpd/conf/httpd.conf file again and look for a line that looks like:

```
DirectoryIndex index.htm index.html index.html.var index.php
```

Add entries for "index.html" and "index.php" if they're not already there.

Mac OS

Apache is installed by default on Mac OS X and simply needs to be turned on from within System Preferences. Select the Sharing panel, and then check the box for Web Sharing.

Create default Web HTML page

Regardless of which OS (and Web server) you're using, it's a good idea to put a simple, but custom, HTML page in the root of /Sites. If you give it the name of your Web server default document configuration (e.g. 'index.html'), your Web server will automatically load it as soon as you navigate to 'localhost', even if there is not page name specified. I have a simple "Hello World" page like shown in **Figure 10**.

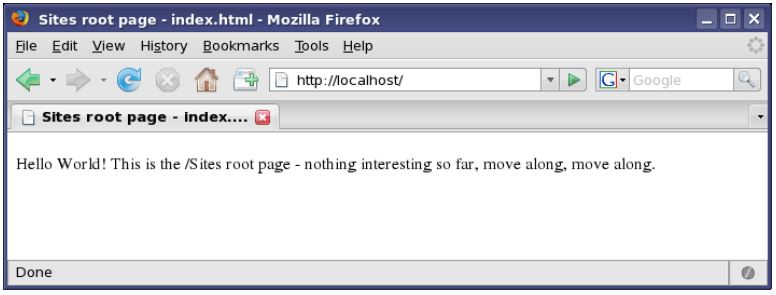

***Figure 10**. Localhost serving up a new default document.*

This tells me that my Web server is running and dishing out pages properly. (Note that the address in the address bar says 'http://" and not "file:///". If the latter is displayed in your browser address bar, it means that your browser is dishing out a file from your local hard disk simply as a file, not serving it up after being processed by your local Web server.) The content of the index.html page (found in the CH02.ZIP file) looks like this:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
```

```
<meta name="keywords" content="site root">
<title>Sites root page - index.html</title>
</head>
<body>
<p class=heading>Hello World! This is the /Sites root page - nothing interesting so
far, move along, move along.</p>
</body></html>
```

Again, the advantage to having this page is that when you navigate to http://localhost, you will see it instead of the Web server's default page, and thus verify that your Web server is running properly and configured so that localhost points to your /Sites folder.

Another configuration scenario is to do development completely on a local machine, and use source code control software to keep the master copy on the file server or a development server that you share with other developers. And a third scenario is to do all development locally and deploy to a hosting server. Whatever works best for you!

Verify PHP is installed and running

Taking this concept one step further, you could (and should) also create a simple PHP page (named index.php) to make sure that PHP is installed and running. I have a test PHP page (index.php, found in CH02.ZIP) in the root of /Sites with the following contents:

```
<html>
<head>
<title>/Sites root page, now displayed in living color, via PHP: Hello
World</title>
</head>
</body>
<h1>Hello World</h1>
<?php echo "Golly, I'm a PHP script (index.php) running on the Sites root page.";
?>
<br><br>
<?php
echo('Today is... ' . date('l, F dS Y. ') );
?>
<br><br>
<?php
phpinfo()
?>
</body>
</html>
```

Running this page via http://localhost/index.php will display a page like shown in **Figure 11**.

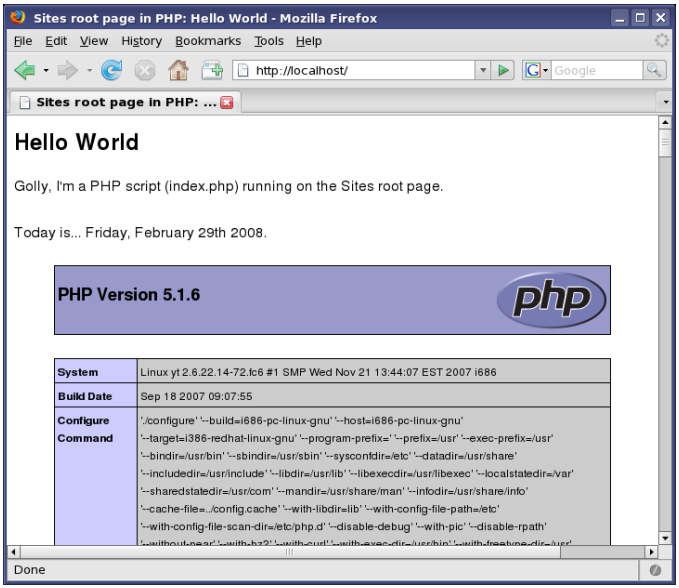

***Figure 11**. Localhost serving up the default PHP file.*

This is a great tool to verify that PHP is installed and running properly. (A significant percentage of DokuWiki installation problems end up being a result of an incorrect PHP installation.) Note that the PHP code tells you both the current date and time, and lists the configuration of your PHP installation, which is very handy for troubleshooting. I recommend you do this if you haven't already.

As a Best Practice, it's good to have a script with phpinfo() available when you're troubleshooting, but you don't want to leave that script on a page accessible by the outside world, as it will tell the Bad Guys too much about how your site is configured, and thus give them information that would be useful in hacking your site.

Dealing with multiple Web sites

Like I said, you can't just have one. Web sites, that is. So I point my Web server root to the main folder, /Sites, and then access a specific site by including that folder's name, like so:

```
http://localhost/Hentzenwerke
http://localhost/WhilHentzen/cust_one
```

The Web server will look for the default page in that directory. Taking the suggestion of having a custom "index.html" file one step, further, you might want to create custom default documents for each subfolder that will contain a separate wiki to make sure you are where you think you are.

Installing DokuWiki on a Local Machine

Now we get to the last five steps - download, extract, copy, install, and run. Navigate to

```
http://www.splitbrain.org/projects/dokuwiki
```

The link for the current version as well as the current version under development and older releases are all available on this page. The DokuWiki Website,

```
http://www.dokuwiki.org/DokuWiki
```

contains all you ever wanted to know about DokuWiki.

As of this writing, the latest release is in the file

```
dokuwiki-2009-02-14.tgz
```

If you're a Windows developer, the 'tgz' extension may unnerve you a bit. Not to worry - it's simply a ZIP file produced by a Linux archiving program. Most popular Windows compression programs will be able to handle this file without pause. The details are here:

```
http://www.dokuwiki.org/install:unpacking
```

The DokuWiki wiki

The DokuWiki wiki is a wonderful resource. Instead of repeating the entire URL throughout the book, I'll just use the name of the specific page. So a reference to

```
install:unpacking
```

can be accessed through the URL:

```
http://www.dokuwiki.org/install:unpacking
```

Download

Download the file and pop it somewhere on your drive where you want to work with it. I have a folder on my local machine called "zips" which is where all my downloaded files land. Once I decide a file is worth keeping, I send a copy off to an archive folder on my file server so if I need it again, I don't have to rely on being able to connect to the Web site that was hosting the original.

In this case, however, I immediately put the dokuwiki-2009-02.14.tgz file in the dw_masters folder (described earlier in the "DokuWiki files location" section) so that I can begin working with it.

Extract files

Once downloaded, the next step is to unzip the file. Depending on what software and mechanism you use, you will eventually end up with a folder named 'dokuwiki-2009-02-14', with the following contents:

```
/bin
/conf
/data
/inc
/lib
.htaccess.dist
COPYING
doku.php
feed.php
index.php
install.php
README
VERSION
```

(You're safe using an "Extract here" menu option from your file extraction software, as opposed to 'Extract to dokuwiki-2009-02-14' because the files in the .tgz file are already wrapped in a dokuwiki-2009-02-14 folder. 'Extract here' will create the dokuwiki... folder for you.)

A view of this structure in Konqueror, the file manager I use on my Linux Fedora machines, is shown in **Figure 12**.

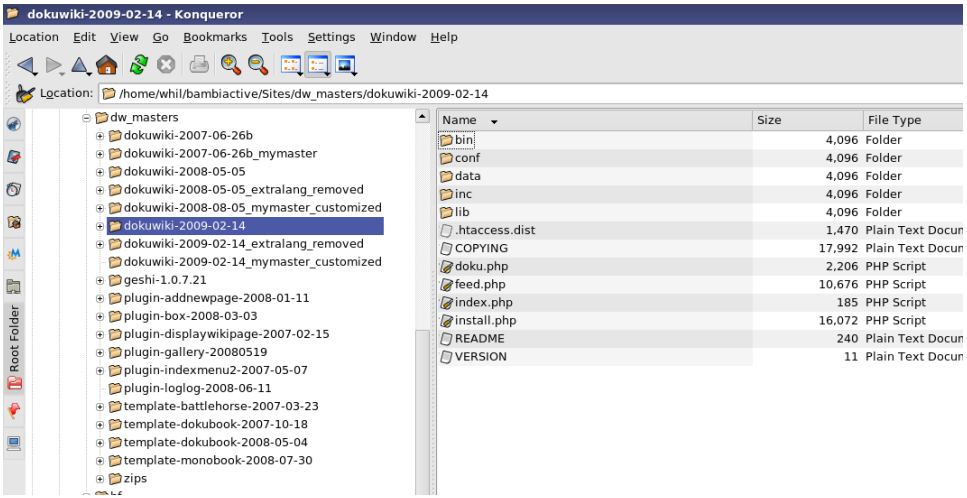

***Figure 12**. The Dokuwiki folder structure.*

This looks like a lot of stuff all of a sudden, but I'll explain it all in short order. Right now, let's look at the contents of the dokuwiki-2009-02-14 folder.

The index.php and install.php files are the main programs that we'll need. Thus, you can see that the dokuwiki-2009-02-14 folder represents the root of the DokuWiki file system.

If you take a look around inside this folder structure, you'll find that there are over 2480 files scattered among 350 folders. Before you freak out, let me reassure you that a great many are there simply to provide internationalization for DokuWiki. The core engine has been translated into about 50 languages, each of which has its own set of folders for a variety of purposes. Additional components have also been translated into a number of languages, each of which also requires a bucketful of files. I'll show you how to get rid of extraneous languages and other files if you don't need them in Chapter 15.

Copy to destination

We're simply going to copy the entire dokuwiki-2009-02-14 folder structure to our Web server folder, and then rename the dokuwiki-2009-02-14 folder to a name more to our liking. For example, suppose I want to create a wiki for 'cust_one' on my 'whilhentzen.com' Web site. I'll copy the DokuWiki folder so it lands here:

```
/Sites/whilhentzen/dokuwiki-2009-02-14
```

and then I'll rename it, like so:

```
/Sites/whilhentzen/cust_one
```

and I'll navigate to it with my browser like so:

```
http://localhost/whilhentzen/cust_one
```

When I create a second wiki for another customer, I'll copy and rename so the wiki files end up in

```
/Sites/whilhentzen/cust_two
```

and I'll navigate to that wiki with my browser like so:

```
http://localhost/whilhentzen/cust_two
```

(You may be asking yourself if this means that each wiki is using its own copy of the software, and the answer is yes. This is not a concern because the entire system (without your content) only comprises a couple of megabytes (about 7 MB if you keep all of the internationalization files) and it's extremely easy to update the source code to a newer version. Additionally, there's a technique that allows you to use one set of DokuWiki code to run multiple, separate wikis, described at **tips:farm**.

Up to this point, I've used generic examples of 'cust_one' and 'cust_two' for names of wikis. From now on, I'll use a more palatable example - a wiki for the specification of a software program that allows sales reps of my favorite customer, "Dynamite Construction Machines", to develop customized configurations of their products for their customers.

Run the installer

As I mentioned earlier, there are two files of immediate interest in the DokuWiki root: index.php and install.php. These are text files that contain PHP programs. You navigate to them in your browser just like an HTML page. The Apache Web server, if properly configured, will take the PHP code in a PHP program (or 'script', as they are more commonly referred to), interpret it, and return results.

We're most interested in "install.php" at the moment. To execute it, enter

```
http://localhost/path-to-dokuwiki/install.php
```

in your browser. For me, this URL is

```
http://localhost/WhilHentzen/cust_one/install.php
```

The Apache Web server will execute the program and return the result - the DokuWiki Installer Web page, as shown in **Figure 13**.

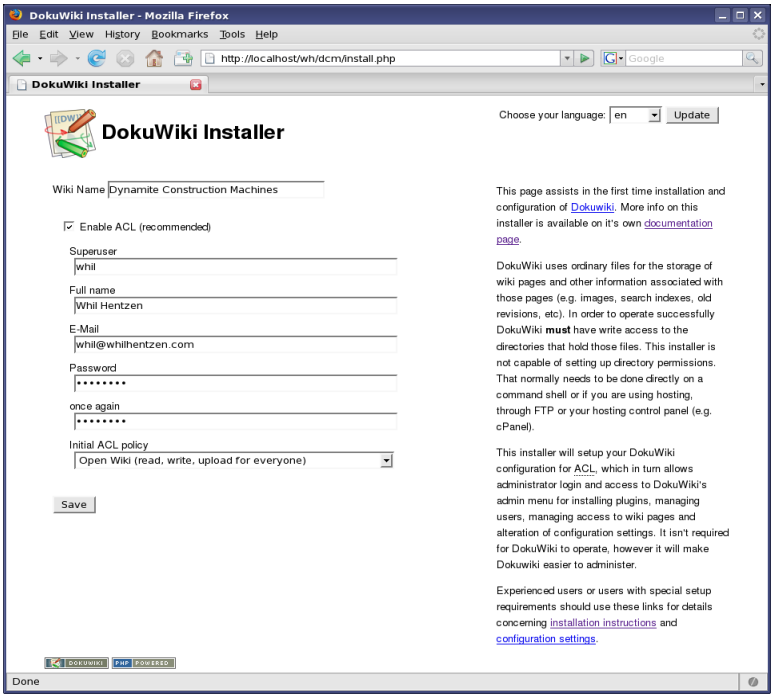

***Figure 13**. The Dokuwiki Installer page.*

If you're looking at this page, you can continue to the next section. However, it's possible that you've run into problems at this point. The most common problem displays a page that looks like **Figure 14**.

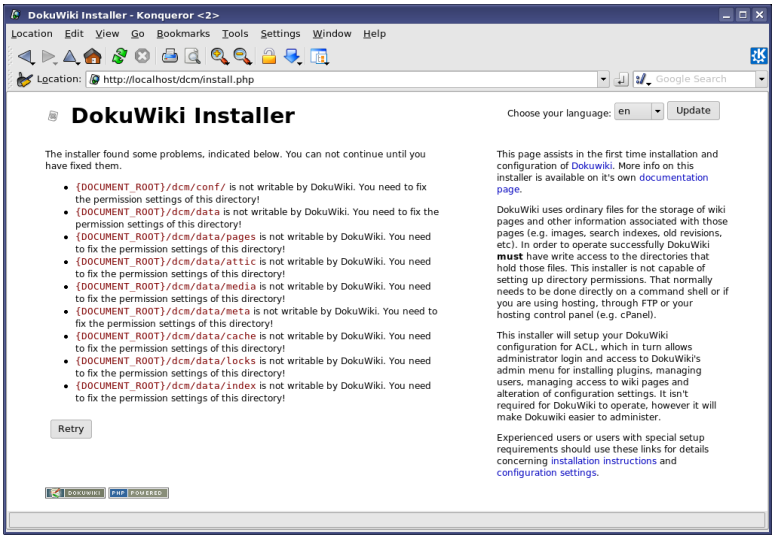

***Figure 14**. The manifestation of a common installation problem.*

Let's take a look at what's going on here.

File permissions errors

Whoa. At this point, some folks run into file permission problems, both locally and when installing it on a shared host. In short, the /data and /conf folders - and all of the subfolders in /data (there aren't any subfolders in /conf) - must be writable by the Web server user in order to work. After all, the Web server will be writing to all of those folders.

You can try

```
chmod -R httpd conf/ data/
```

The result should be permissions for **data** changed from **rwxrwxr-x** to **rwxrwxrwx** and **conf** from **rw-r--r--** to **rw-rw-rw-.**

Given the wide variety of possible configurations, this example may not work precisely for your situation. There are a number of very useful topics on the DokuWiki wiki. There's an excellent, excellent screencast that discusses how to install on a hosted server, and it covers permission problems in some detail. Additionally, the **install:permissions** help topic has further info about handling permissions. I suggest you read through the topic several times in addition to the discussion in this book.

The DokuWiki Installer

The purpose of the DokuWiki installer is four-fold. First, it'll verify that all files and functions exist. Next, it'll ensure that file permissions have been set correctly. (Figure 14 showed you the result of an Installer routine that detected incorrect permissions.) Third, it'll allow you to set a couple of initial configuration options. Finally, it'll (optionally) create an administrator account and an initial ACL ("Access Control List") policy.

The following steps show you how to set up a simple wiki by selecting one set of options. In Chapter 3, I'll explain each of the permutations of the other available options.

Assuming that you got to the Web page shown in Figure 13, the installer has already passed the first two steps. So let's move directly to the third step and fill in the info.

Choose your language combo box: First, easy to miss - the combo box in the upper right corner defines what language messages and prompts will be displayed in the wiki. By default, it's set to English, but if you open the combo, you'll see quite a list of options.

Wiki Name text box: The name will show up in the top banner of every wiki page, so choose it with some thought. You can change it later, but still, an off-the-cuff shot like "Some Dorki Wiki" is probably not the best choice.

Enable ACL (recommended) checkbox: The "Enable ACL (recommended)" checkbox is checked by default (thus the 'recommended' admonition.) ACL stands for "Access Control Lists" and they are just what they sound like - lists that control access to the wiki. Since we're interested in creating a private wiki just for our software development group, we'll need ACLs enabled. Keep it checked.

Superuser textbox: Enter a username for the administrator. This will typically be you. Note that the name of a user who edits a page is displayed at the bottom of the page as well as in the page history, so choose your superuser name with care as well. Slang or off-color names may appear funny now but they'll come back to haunt you - and they're a huge pain to fix.

Full name textbox: The value you enter here is used in Email notifications and RSS feeds.

Email textbox: The value you enter here is the address used when you request to have password resent to you ("password reset") and if you choose to be notified via email when pages are modified (namespace and page subscriptions.)

Password and **once again** textboxes: Passwords are.... passwords. They're encrypted in the system. The type of encryption can be changed in the configuration later on.

Initial ACL Policy combo box: Finally, the "Initial ACL policy" combo box determines how non-admin users will be able to access pages. Eventually we'll be selecting "Closed wiki" since this is a private wiki, but for the time being, leave it at the default "Open Wiki".

Hit Save and you're done. You should see a screen as shown in **Figure 15**.

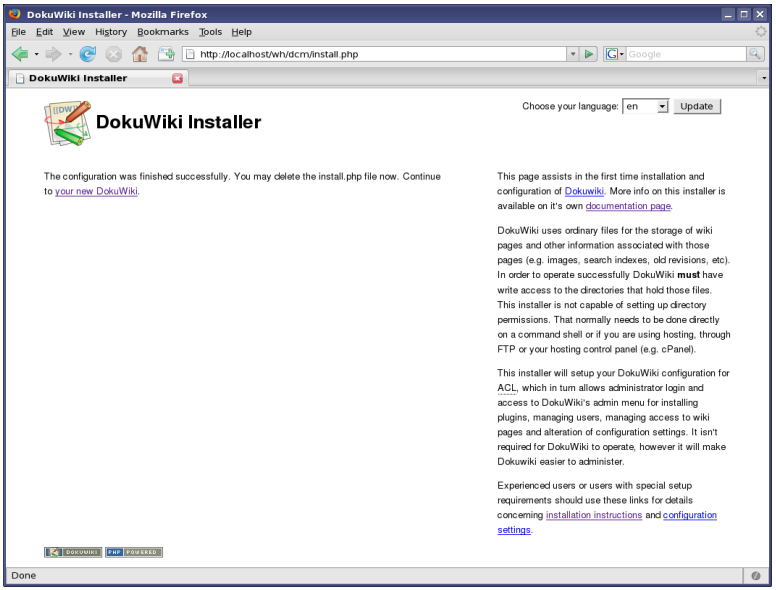

***Figure 15**. Initial installation is complete.*

The link that says "Your new DokuWiki" will take you to the home page of your brand new wiki. Before you click that and we get going, it'd be handy to know how to get to your wiki without using this link, since you probably won't ever see this page again.

Run the program

In the future, you can just run the index.php program, like so:

```
http://localhost/path-to-dokuwiki/index.php
```

For me, this is

```
http://localhost/whilhentzen/cust_one/index.php
```

Actually, if you've configured Apache to recognize index.php as a default home page, you can just enter the directory and index.php will be executed automatically:

```
http://localhost/whilhentzen/cust_one
```

Click on the link "Your new DokuWiki" and you'll be presented with your new wiki, as shown in **Figure 16**.

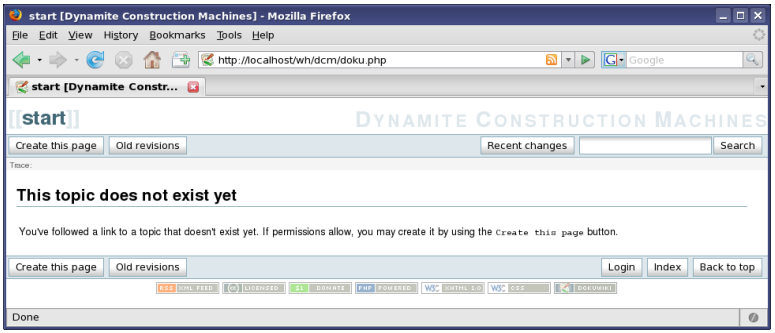

***Figure 16**. The start page of your brand new wiki.*

You're now ready to get to work - with a start page that contains no content. Let's take a look around and then create a page.

What's on a Wiki page

Before we create a page, we should see what we've got to work with. Starting from the upper left, here's what you see.

[[start]]

This is the name of the page. By default it's named 'start' but you can change it to something else via the Configuration Manager (covered in Chapter 7). This page doesn't actually exist yet; it's a placeholder for a page that you can create.

Dynamite Construction Machines

This is the title of the site, entered during installation. This, too, can be changed using the Configuration Manager. The title of the site is also a hyperlink that points back to the main page of the wiki.

The next row contains four buttons and a text box.

Create this page button

Clicking the "Create this page" button allows you to create a new page. Once a page has been created, the caption on the button changes to "Edit this page". The next section will cover this process.

Old revisions button

Once a page has been edited, old versions of the page - previous copies of this page that were saved - are stored and available for viewing via the "Old revisions" button. Clicking the button

will display a list of older versions; clicking on an entry in the list will display that page. See Chapter 5 for a detailed explanation of how this works.

Recent changes button

Clicking this button will display a list of up to the 20 most recently changed pages in the wiki. (Note: the listed changes are for the entire wiki, not just the current page.) If there are more than 20 pages that have been changed, a "less recent >>" button will display under the list. See Chapter 6 for a detailed explanation of how this works.

Search textbox and button

Entering a word or phrase will generate two lists of matching results. The first is a list of page names that match contain the search term. The second is a list of pages where the search term was found in the contents of the page, together with the number of times the term was found in the page.

Trace: label

The light grey band under the row of buttons, with the word "Trace:" aligned to the left will display your history of navigation through the wiki, much like a browser history (a.k.a. breadcrumbs). Settings in the Configuration Manager control how many levels will be shown or allow the band to be turned off completely.

This topic does not exist yet

Now we come to the interesting part of the page - the page header and the contents of the page. I'll address the basic process of creating a page in the next section; Chapter 4 covers the creation and editing of a page in detail.

Under the page contents is a second row of buttons; Create this page and Old revisions are just duplicates of the ones at the top of the page. Let's look at the three buttons in the lower right:

Login button

If ACLs are enabled, the Login button will be displayed. Clicking this button will display a login page. Since this was configured as an open wiki, you may think that the Login button is unnecessary. When ACLs are enabled on an open wiki, the Login provides access to the Administration functions for the superuser who was identified during installation (refer back to Figure 13.)

Index button

Index will display a list of all namespaces and pages in the wiki. See Chapter 6 for details.

Back to top button

The "Back to top" button is useful when the page gets long - clicking it will simply reposition focus to the top of the window.

Footer images

The 7 images at the bottom of the page identify various attributes about the wiki and provide additional functions, such as creating an RSS feed, donating to the DokuWiki author and links to the W3C, CSS and DokuWiki Web sites. These can be customized as well. See Chapter ___. (*\\\ WH find which chapter))

Now that we know our way around, let's do something.

Create and save a page

Now that we know our way around, let's create a page called Hello World. Click the "Create this page" button. You'll get a page with an edit box in the middle as shown in **Figure 17**.

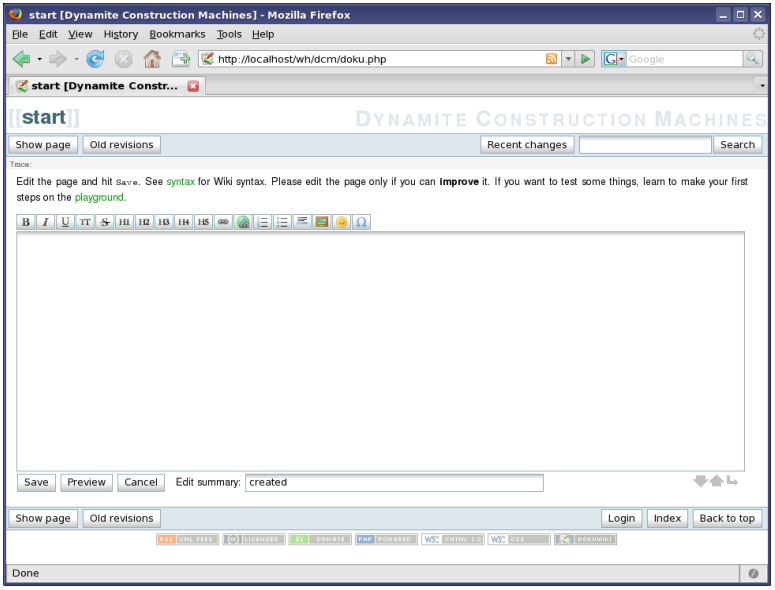

***Figure 17**. Creating your first page in DokuWiki.*

Type your content (for example, "Hellow World!") into the edit box and click Save. The result is displayed as a new page in the wiki. End of story. Now you've got your own wiki up and running.

I'll cover details on how to use the various controls and otherwise format your text in Chapter 4.

Oh, and since you're a programmer, you probably want to know what's going on under the hood. There's now a new text file called "start.txt" in the data/pages folder under your DokuWiki installation. I'll cover the details of folder and file structures in Chapter 11.

Conclusion

In this chapter, I've covered the basic steps to your first DokuWiki installation. While enough to get you started, there's more to installation than this. In the next chapter, I'll explore the various installations options in detail, and in Chapter 15, I'll provide a roadmap for speeding up the installation process.

> Updates and corrections to this chapter can be found on Hentzenwerke's Web site, www.hentzenwerke.com. Click "Catalog" and navigate to the page for this book.