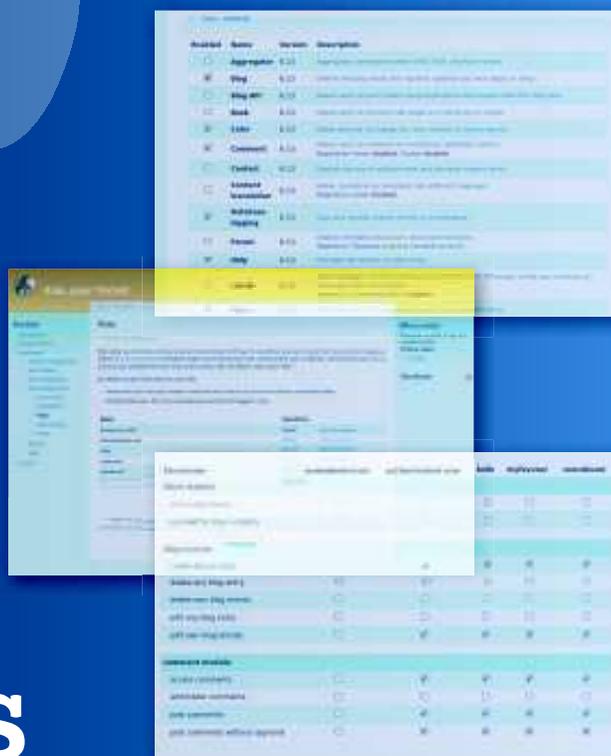


Drupal



Drupal Quick and easy websites



Want to make your mark online? **Shashank Sharma** helps you to create a Drupal website in minutes.



Our expert

Shashank Sharma

is co-author of *Beginning Fedora* and has written about free software for over four years. He's also contributed to **Linux.com**.

You probably already have a few ideas about what it takes to set up a website, but put those to one side for now, because we're going to look at a different way of doing things. With *Drupal*, all you really need is a name and an idea of the type of content you want. However, before we get going, the big question you have to answer is: with all the free hosting services available today, do you really need your own website at all? If the answer is yes, read on to discover how you can create anything from a simple blog to a complex website using *Drupal* in a matter of minutes.

Drupal is the ideal tool for all manner of site creation, especially since it's easily extensible via the many available modules. You can use it to run a simple blog, forum, wiki, or even a social network. Just like all the ingredients that go into a meal, there are many flavours to consider when preparing a website. The software is one, then there's the server and the database configuration. The colour scheme and logo are the final garnish, and we won't need them until the site's ready.



» What's this site called anyway? Don't leave visitors stumped with the default 'Drupal Powered By Bitnami' title.

However, just knowing the ingredients isn't enough to make a great dish; you also need them in just the right quantities. Could this be the reason why – despite being familiar with several different CMSs and blogging tools, *Apache*, *MySQL* and *PHP* – you've not yet managed to cook up a storm online?

But surely, I hear you ask, you can't configure *Apache*, set up *MySQL* and connect them to *Drupal* in a matter of minutes? Amazingly, you can – all thanks to the brilliance of *Bitnami* software installers. And not only will you avoid having to worry about creating databases, you won't even have to install *Drupal* yourself.

Bitnami's Drupal

Apart from *Drupal*, the *Drupal Bitnami* stack also contains an independent instance of *Apache* and *MySQL*. Naturally then, the *Drupal Bitnami* stack is 56MB, compared to *Drupal's* tarball, which takes up less than 2MB. Each stack includes everything you need to run the primary application and since it's entirely self-contained, you don't have to worry about it conflicting with software already installed on your machine. Likewise, the *Apache* and *MySQL* configuration of the *Drupal* stack won't interfere with any existing *Apache* and *MySQL* settings you have. This makes *Bitnami* stacks ideal for demoing a site, running a local instance before you move it over to production machine.

In addition to *Drupal*, *Bitnami* currently provides software stacks for many other applications, including *WordPress*, *Moodle*, *SugarCRM*, *Subversion*, *Trac* and many infrastructure stacks such as LAMP and WAMP.



To begin, head on over to <http://bitnami.org/stack/drupal> and download the **bitnami-drupal-6.12-2-linux-installer.bin** file. This is the Linux installer that contains all of *Drupal's* dependencies, namely the *Apache* HTTP server, *MySQL* and *PHP*. We need this file to be executable, so open a terminal, switch to the directory where you downloaded the file, and change its permissions using the command:

```
chmod 755 bitnami-drupal-6.12-2-linux-installer.bin/
```

You can now install *Drupal* by typing:

```
./bitnami-drupal-6.12-2-linux-installer.bin
```

First, you're asked if you wish to install *PHPMyAdmin*. This enables you to easily manage *MySQL* databases, so we recommend that you do. Click Forward to continue. You'll then be asked where you wish to install *Drupal*. The default path is `~/drupal-6.12-1`, but you can specify a different directory if you require. Next, you'll be asked to create an admin account, so provide a username, password, your name and email address and click Forward.

And that's it: you've just installed *Drupal* – simple, wasn't it? But before we get started, we want to make a few tweaks, which we can do by using the **ctlscrip.sh** file in the installed **drupal-6.12** directory. The script accepts any of three arguments – **start**, **stop** or **restart** – so, to start *Apache* and *MySQL*, use the following command:

```
ctlscrip.sh start
```

You can now access the site with your browser. The **README.txt** file under the `~/drupal-6.12/` directory (or the relevant directory if you opted not to use the default location) contains the username and password you'll need for the *MySQL* admin account.

The configuration

Now the installation is complete, you can access your website by pointing a browser to <http://hostname:8080/drupal>, replacing **hostname** with whichever name you've assigned to your machine. Ours is called *Playground*, so <http://playground:8080/drupal> brings up our freshly installed *Drupal* site. You can also use the IP address of your machine if you prefer, like so: <http://192.168.2.5:8080/drupal>. When you access your site for the first time, you'll be greeted with configuration instructions and a login interface on the left bar.

Using the administrator username and password you chose when installing *Drupal*, log in using the left sidebar. We can now begin configuring our site by clicking the *Administer* button on the left sidebar. We'll begin by specifying a site name and then we'll configure other aspects, such as layout, users and comments.

On the left sidebar, click *Administer* > *Site Configuration* > *Site Information*. From this page you can type in a name for your website, your email address, tagline and so on. You can also provide a footer message, be it a simple copyright notice or the licence information for the content on your website. Finally, you can specify a custom default page if you'd rather people didn't arrive straight at the main page – an *About Me* page, for instance – and use that as the landing page.

The other items listed under *Site Configuration* enable you to tweak various settings, including the formatting of the date and enabling clean URLs. Once your site is running, depending on the traffic and the site's content, you may wish to investigate *Drupal's* caching options. With caching enabled, visitors to your site will notice a significant performance



boost. You can enable caching by clicking *Administer* > *Site Information* > *Performance*. Apart from the database and CSS caching options, you may also wish to look at the *Performance* modules.

▶ **Don't use the Primary Links to create your blogroll: all the links will show up in the header.**

Start typing

When you're ready to create some content for your site, click *Create Content* on the left sidebar. *Page*, *Blog Entry* and *Story* are the three available options. Create a *Page* if you want to record any static information – note that data such as a list of your favourite movies belong to a *Page*, as does all other personal information. *Pages* don't show up on the main page of your site and visitor and registered site members can't comment on them. This is where *Story* and *Blog Entries* differ, because they're displayed on the main page and enable comments.

You're probably wondering what you need to do to give visitors access to your pages if they don't appear on the main page. Well, to make pages visible to others, go to *Administer* > *Site Building* > *Menus*. Now select *Primary Links* and click the *Add Item* button. Type in the path to your existing page under

Creating a blogroll

A blogroll is a list of blogs or websites that you often read, and would like to recommend to your website visitors. To create your own blogroll, click *Add Menu* under *Administer* > *Site Building* > *Menus*. Type in a menu name, which is used internally to create a URL for the menu overview page, and then *Blogroll* in the *Title* text field, or whatever you wish to call the collection of links. Click *Save* when you're done.

You'll now see *Blogroll* listed in the left sidebar under *Menus*, along with *Navigation*, *Primary* and *Secondary Links*. Click *Blogroll* in the left sidebar and then click the *Add Item* button.

This is just like creating *Primary Links* for your static *Pages*. The *Enabled* checkbox is already marked and *Blogroll* is selected by default on the *Parent Item* drop-down list, so just click *Save* at the bottom of the page. Repeat these steps to add more items to the *Blogroll* menu.

When you're done, click *Administer* > *Site Building* > *Blocks*. Look for the *blogroll* entry on this page and select a position for it from the *Region* drop-down list. Click the *Save Blocks* button when you're done. Visitors to your site will now see a *Blogroll* block at the position you specified.



› **Nothing sneaks into those sidebars without getting clearance from the Administer > Site Building > Blocks page.**

› the Path field. You can also point to external websites if you'd like. For static pages, the path is something like **node/4**. Now write a name for the link under the Link Title field. Finally, click the Save button, making sure the Enabled checkbox is ticked.

If you'd rather post your thoughts to the world in blog form, click the Blog Entry Item under Create Content. The Create Blog Entry page appears and this is where you can type your blog posts. In addition to text fields for the post title and body, you can also configure some settings for individual blog posts, such as using HTML and controlling whether visitors can read and write comments about particular entries. The default configuration is for single-user blogs, but if you'd rather have a multiuser blog, you'll have to enable the Blog module with Administer > Site Building > Module > Blog.

Quick tip

You can define the length of the teaser text for lengthy posts by clicking Administer > Content Management > Post Setting.

Your point of view

Drupal's Views module enables you to create custom queries that control how certain data is presented to users. You can, for instance, create a monthly archive for your posts, or a Latest Comments block. To create a custom view, you'll need to install and enable the Views module, so download **views-6.x-2.6.tar.gz** from <http://drupal.org/project/Views>.

“If you want to enjoy a fresh look, updating your themes is easy.”



› **The default theme of Drupal is all right, but it's easy to change if you'd like something more colourful.**

You're now ready to begin, but before you dive into the confusing and overwhelming world of Views, we recommend you install the Advanced Help module. This way, you'll have a helpful guide to refer to when creating complex new Views for your site. After installing the Views module, click Administer > Site Building > Views to access the Views page. The default Views for recent comments, monthly archives and so on are listed here, and you can start any of them by clicking the Enable button. If you enable any of the default Views, make sure you use the Blocks page to assign it a position, otherwise the View won't be visible to your site's visitors.

To get you used to using the Views module, let's create a new View that displays a list of the newest registered members on your site as an example. To begin, click the Add button at the top of the

Views page. You'll be asked to provide a name, description, tag and so on. The most important task here is to assign the right View Type to your new View, since this describes which arguments, fields, sort criteria and filters are available. Getting this wrong could cost you time later, so it pays to be cautious.

Because we want to create a list of newly registered users, we'll choose User as the View Type. If you wanted to create a list of recent blog posts, you'd have to choose Node instead. When you click Next at the bottom of the page, you'll be taken to the Views User Interface page. From here, you can define



Step by step: Set up a multiuser blog



1 Enable a module

Click Administer > Site Building > Modules. The core modules are already installed, so click the checkbox to enable them. Once done, click the Save Configuration button at the bottom of the page. All enabled modules are listed in a section of the left-hand sidebar.



2 Who can blog?

Click Administer > User Management > Roles. There are two user roles: Anonymous and Authenticated. Permissions for Authenticated users apply to all registered users. To restrict blogging to certain users, type in their username and click Add Role.



3 Make users bloggers

Click Administer > User Management > Permissions. Permissions are split into modules. To enable blogging for a user you've defined in the Roles page, click Create Blog Entries under their name and Save Permissions. The user can now start blogging.

