

Building Your Blog With Drupal 7

A step-by-step guide

This Drupal 7 e-book is brought to you by Learn By The Drop, *a place to learn Drupal*. Visit the website at <http://learnbythedrop.com> to subscribe to video tutorials and blog posts that will help you get the most out of the Drupal platform.

Building Your Blog With Drupal 7

Table of Contents

Introduction	1
Who Will Benefit From This Guide?	1
Your Blog	1
Getting Started	2
Local Site	3
Web Based Site	3
Downloading and Installing Drupal	4
Post Installation Configuration	9
Installing Necessary Modules	12
Module Installation Basics	12
Installing Necessary Modules	15
Building Your Blog	16
Setting Up Category Tags	19
Adding A WYSIWYG Editor	23
Adjusting Your Text Formats	26
Create A Shortcut To Add Blog Posts	27
Customizing Post Paths	28
Creating Content	31
Getting Your Blogs Posts Onto The Front Page	33
Arranging Sidebar Blocks	37
Creating Your About Page	39
Creating Your Contact Form	40
Preventing Contact Form Spam	46
Checkpoint 1	48
Comment Moderation Preferences	49
Easy Content Sharing	51
Embedding YouTube Videos	53
Inserting Images Into Posts	59
Checkpoint 2	63
Managing Themes	64
Editing Enabled Themes	66

Building Your Blog With Drupal 7

Adding New Themes	69
<i>Administering Your Drupal Site</i>	<i>70</i>
Backing Up Your Site	70
Running Cron	72
The Status Report	73
Module, Theme And Core Upgrades	74
The System Log	75
Adding Additional Authors	77
<i>Final Words</i>	<i>79</i>

Building Your Blog With Drupal 7

Introduction

This guide will show you a step-by-step process for building a blog site using Drupal 7. I want to make clear up front that the approach taken in this guide represents a *recipe* rather than an absolute approach to building a blog using Drupal 7. There are several different recipes that you could use due to the large, ever growing number of contributed Drupal modules. I have chosen the approach that is most comfortable and makes the most sense for me. It's likely that you may want to conduct additional research to find a recipe that meets your specific needs. With that said if you don't know much about Drupal, or have never created a blog with Drupal I'm sure that you will learn a lot.

It's also worth noting that after the initial publication of this guide some of the contributed modules that I mention here are likely to change in different ways. Modules will gain new functionality and additional settings. Some modules may no longer be the best tool for the job. I'll do my best to keep things up to date but your assistance in identifying these issues would be most helpful. Send feedback via the form at <http://learnbythedrop.com/contact>.

Who Will Benefit From This Guide?

If you're new to Drupal then this guide will help you to get on the fast track to learning how to build and manage a Drupal 7 based website. If you're experienced with earlier versions of Drupal but haven't yet explored Drupal 7 then this guide will help you to familiarize yourself with the differences between earlier versions of Drupal and Drupal 7.

Your Blog

Before going further I think it's important to define what this blog is going to be like once we're through. Keep in mind that I have created this list of features based on my (quite extensive actually) experience with building and using blogs over the past decade. You may take issue with the approach and that's okay. Feel free to make some different

Building Your Blog With Drupal 7

choices if you're comfortable with the Drupal platform. As I was saying, here are the general features that will make up our blog site.

- A "front" (or home) page with a list of ten partial (or teaser) posts listed in descending order by published date with a pager to navigate to older posts.
- An "about" page that gives you the opportunity to tell people a little something about who you are and what the site is about.
- A "contact" page that gives people the opportunity to provide you feedback via a simple form whose results are emailed to you.
- A "blog post" content type that includes a *what you see is what you get* (WYSIWYG) editor, a field for freely tagging (or categorizing) your blog posts, an image upload tool that allows you to insert images into your posts, and the ability to easily embed YouTube videos.
- Commenting functionality with moderation so that comments can be added to posts with your approval.
- Spam protection so that your contact form and comment fields aren't deluged by offers for pharmaceuticals, gold watches and World of Warcraft goodies.
- A widget attached to each blog post that makes it easy to share the post on a variety of popular platforms.
- A side bar that includes a list of recent comment and blog post titles with links to the full posts.
- Links (or "URLs") to blog posts that include the posted date and title in the link.
- A tag "cloud" that resides in the sidebar that links to the topics that you have mentioned in your blog posts.

Getting Started

Now that we have the preamble out of the way we can get down to the business of building a blog using Drupal 7. The first thing you need to do is install Drupal 7 on either a local machine or a web server. If you're just learning Drupal and don't plan to launch a site anytime soon then running Drupal on your local machine is a good idea. That approach keeps your costs low because you won't have to purchase a domain name or

Building Your Blog With Drupal 7

a hosting account. If you plan to learn while you build a site you plan to launch soon or will be working with other people on the site then you probably want to do your building on a web server. I will cover the basics of getting started with both approaches below.

Local Site

Your local site approach depends on whether your platform is a Mac or a Windows PC. Macs are generally known for their friendliness for web development. If you are running on Windows then I recommend [WampServer](#). If you are on the Mac platform you should check out [MAMP](#). Both recommended programs allow you to run Apache, MySQL and PHP, all of which are necessary to run a Drupal site.

Web Based Site

If you are taking a web based approach you will need a web host. The conversation about which web host to use would go on for ages since there are so many hosts that could possibly do the job. There are two things that i will point out with respect to web hosts. First, your host must give you access to file transfer protocol (FTP) access, Apache (the web server), MySQL (the database) and PHP (the source code interpreter). You'll need these things at the very least in order to install and manage a Drupal website. Second, you should choose between shared or virtual private server (VPS) hosting.

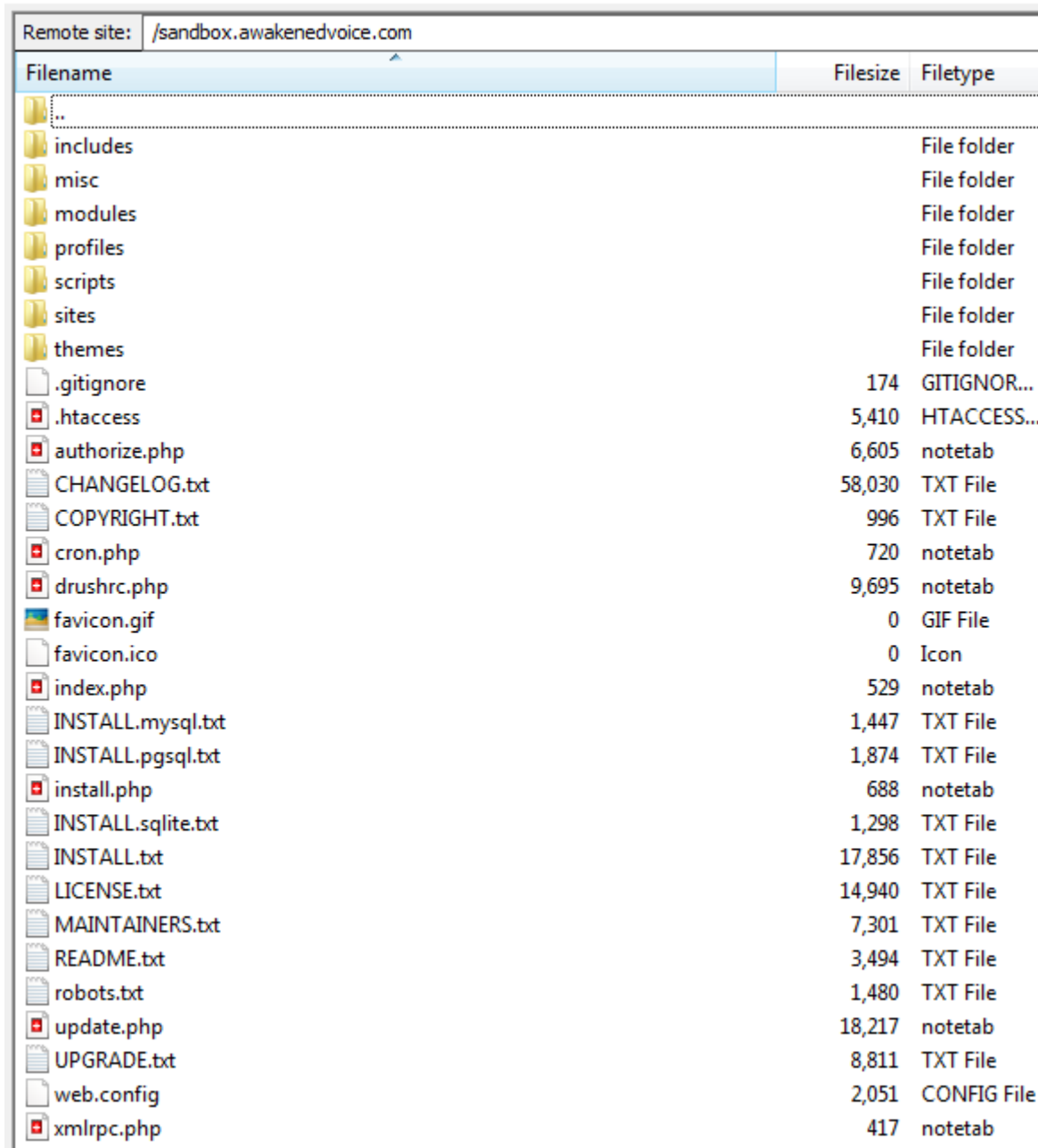
Shared hosting plans are less expensive (\$10 - \$20 per month) but they are also more limited in terms of processing power for your site because your site is *sharing* the resources of a server with many other sites. Shared is recommended only if you expect light traffic to your website. VPS (sometimes called virtual dedicated server or "VDS") is more expensive (\$30 - \$50) per month but you get the advantage of dedicated RAM and CPU resources for your website. If you are building a site that you value and hope to receive a growing amount of web traffic you should probably choose the VPS option.

Building Your Blog With Drupal 7

Downloading and Installing Drupal

Once you have made your choice about how and where to host your Drupal site it's time to get down to the business of building your blog. The first step in that process involves downloading and installing Drupal. You can always find the latest version of Drupal at <http://drupal.org/download>. Go there and download the compressed version of the file to your computer. Once downloaded you should unpack the compressed version of the file into the local directory of your choice. Finally, using your favorite FTP program (I recommend [FileZilla](#)) upload the Drupal files to the root directory of your website so that the structure is something like <site root>/index.php. The image below shows you an example of what the directory should look like.

Building Your Blog With Drupal 7



The screenshot shows an FTP client window with the remote site address `/sandbox.awakenedvoice.com`. The file list is as follows:

Filename	Filesize	Filetype
..		
includes		File folder
misc		File folder
modules		File folder
profiles		File folder
scripts		File folder
sites		File folder
themes		File folder
.gitignore	174	GITIGNOR...
.htaccess	5,410	HTACCESS...
authorize.php	6,605	notetab
CHANGELOG.txt	58,030	TXT File
COPYRIGHT.txt	996	TXT File
cron.php	720	notetab
drushrc.php	9,695	notetab
favicon.gif	0	GIF File
favicon.ico	0	Icon
index.php	529	notetab
INSTALL.mysql.txt	1,447	TXT File
INSTALL.pgsql.txt	1,874	TXT File
install.php	688	notetab
INSTALL.sqlite.txt	1,298	TXT File
INSTALL.txt	17,856	TXT File
LICENSE.txt	14,940	TXT File
MAINTAINERS.txt	7,301	TXT File
README.txt	3,494	TXT File
robots.txt	1,480	TXT File
update.php	18,217	notetab
UPGRADE.txt	8,811	TXT File
web.config	2,051	CONFIG File
xmlrpc.php	417	notetab

Figure 1: FTP View

Once the files are uploaded you should navigate to the `sites/default` folder and make a copy of the `default.settings.php` file so that the name of the new file is `settings.php`.

Once you've done that you'll want to make sure that you have created a MySQL database to serve as the database for your Drupal site. The instructions for creating a

Building Your Blog With Drupal 7

new database differ by platform so I won't even attempt to instruct you on how to do it. Your web host or local server app help files should give you instructions on how to do that. What I will tell you to do is take note of the path to your MySQL server (or database host), the database name, the database username and the database password. You'll need these pieces of information when you get to the first Drupal install screens.

It's now time to take a second to confirm that you've completed a few steps before we move forward.

- Have you uploaded the Drupal 7 files to the root directory of your website?
- Have you created a copy of the default.settings.php file and renamed it to settings.php?
- Have you created a MySQL database and noted the database host, database name and database password?

If you answered yes to all of those questions it's time to move forward with the Drupal install by navigation to the root of your website. Once you navigate to the root of your website you should see the screen below. If you don't see a screen like that then you need to go back and check your setup steps.

Building Your Blog With Drupal 7

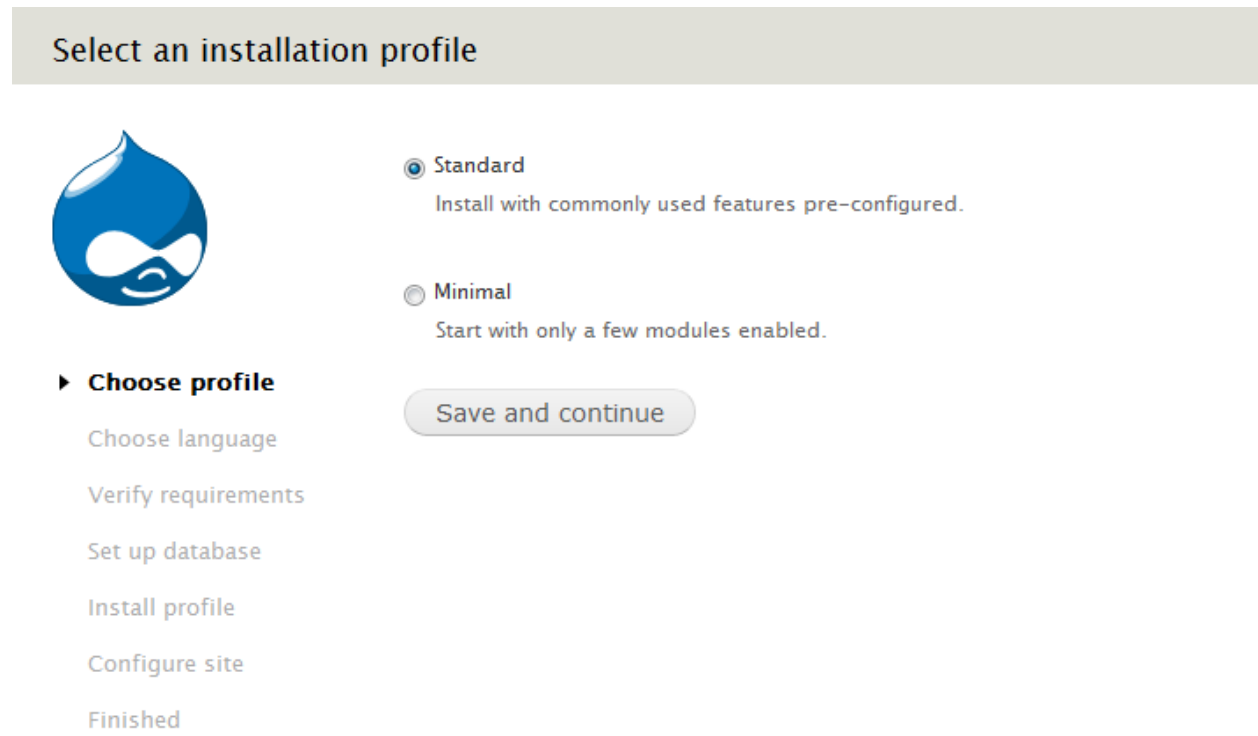



Figure 2: Install Screen

Click **Save and continue** to move to the next option which will be to choose your language. Click **Save and continue** again to move to the *Set up database* window which will look like the image below.

Building Your Blog With Drupal 7

Database configuration



✓ Choose profile

✓ Choose language

✓ Verify requirements

► Set up database

Install profile

Configure site

Finished

Database type *

☒ MySQL, MariaDB, or equivalent

☐ SQLite

The type of database your Drupal data will be stored in.

Database name *

The name of the database your Drupal data will be stored in. It must exist on your server before Drupal can be installed.

Database username *

Database password

▼ ADVANCED OPTIONS

These options are only necessary for some sites. If you're not sure what you should enter here, leave the default settings or check with your hosting provider.

Database host *

If your database is located on a different server, change this.

Database port

If your database server is listening to a non-standard port, enter its number.

Table prefix

If more than one application will be sharing this database, enter a table prefix such as *drupal_* for your Drupal site here.

Save and continue

Figure 3: Database Configuration

Enter the values that you copied after creating your database then click the **Save and continue** button. Drupal will take a few minutes to go through the setup process if everything completes properly you will be prompted for three things:

1. Site Information;

Building Your Blog With Drupal 7

2. Site Maintenance Account¹;
3. Server Settings.

Once you have filled out the relevant information and clicked Save and continue you should see the following window.

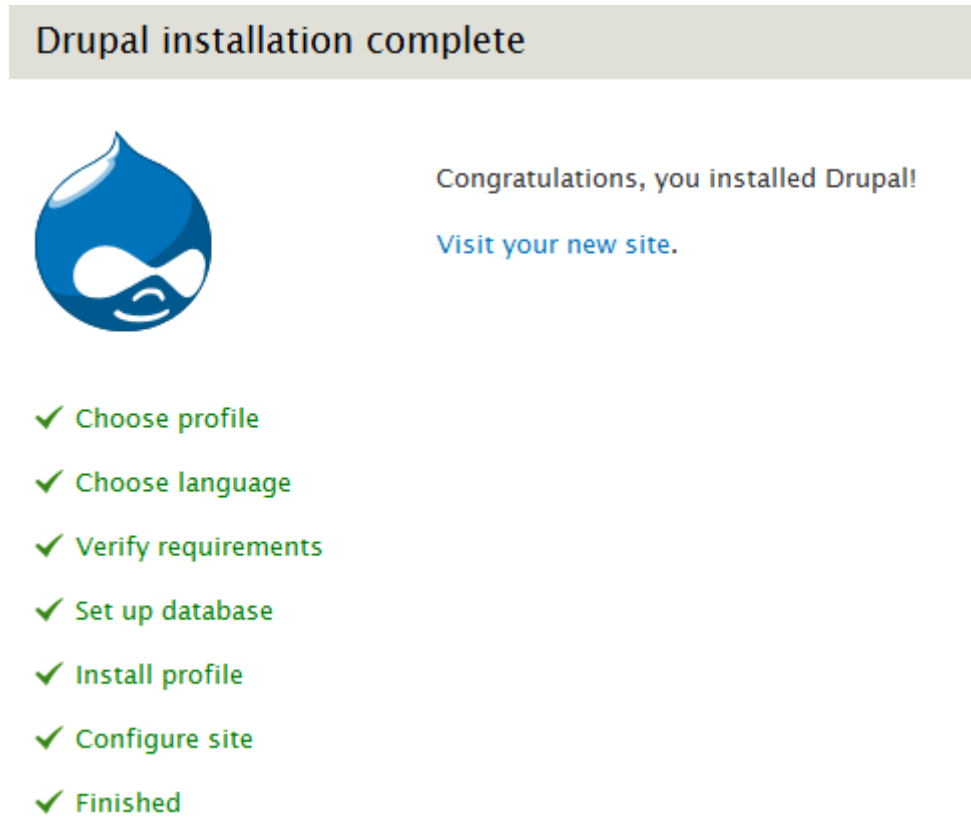


Figure 4: Install Complete Window

Click on the **Visit your new site** link and you'll be ready to move to the next phase.

Post Installation Configuration

If your installation was successful then you will be ready to perform some basic configuration on your new Drupal site. In order to do that you will need to get comfortable with the administration menu bar (pictured below) at the top of the screen.

¹ The first account that you create is very important because this user has *permissions* to do everything.

Building Your Blog With Drupal 7

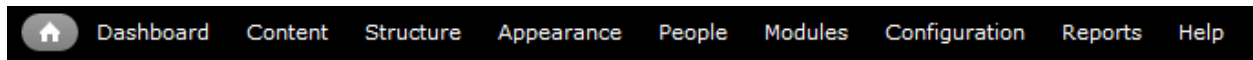


Figure 5: Administration Menu Bar

If you look at the rest of the front page you'll see that the site name you entered earlier now populates the header of your site, along with the standard Drupal logo.

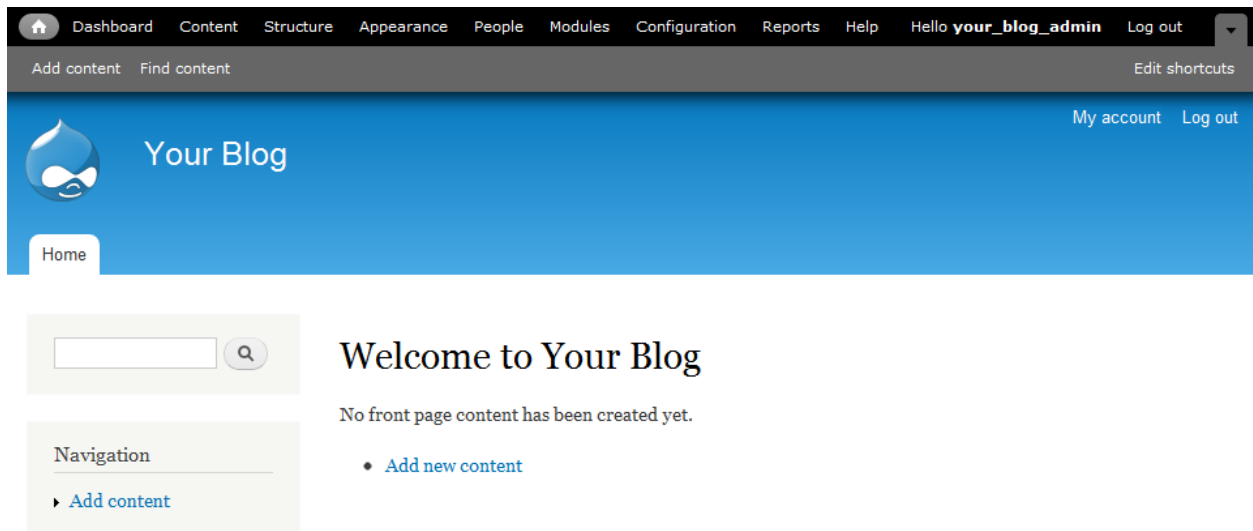


Figure 6: Front Page

Since this is your site you'll probably want to get rid of the Drupal logo. Here's how you do that. Click on the **Appearance** menu link. You'll see a window that looks like the image below.

Building Your Blog With Drupal 7

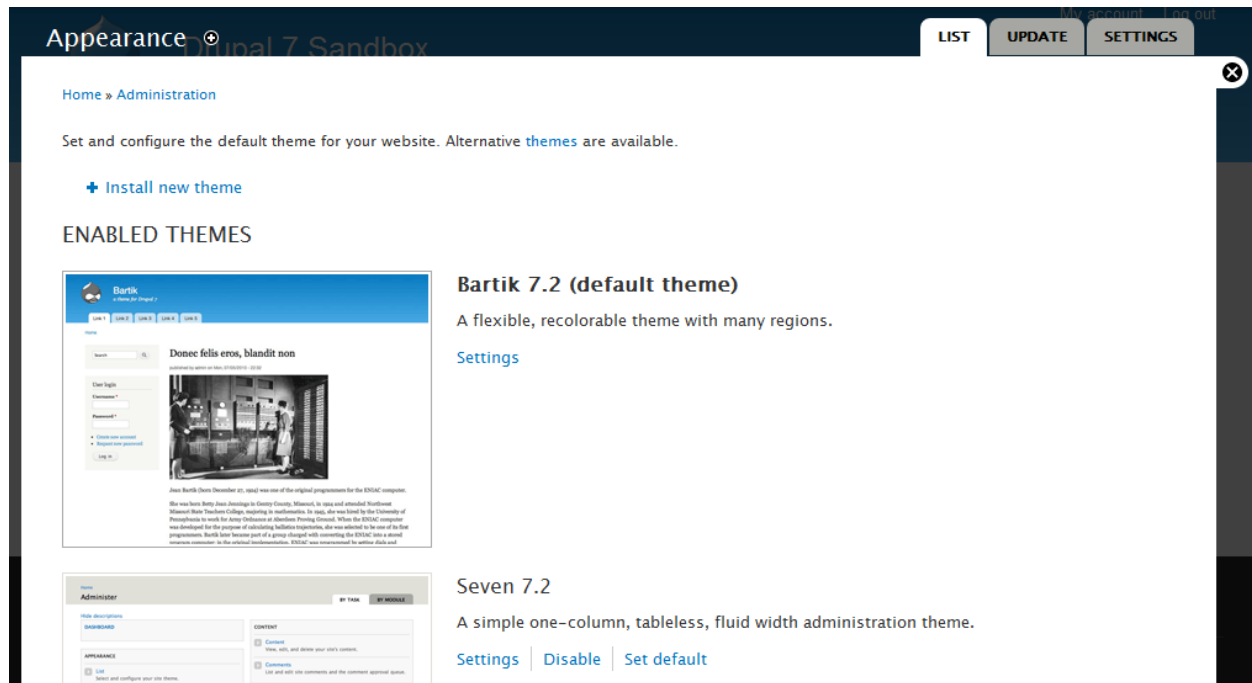


Figure 7: Appearance

Click on the Settings link on the upper right side of the window. At the top of that next screen there will be a *Toggle Display* box. The first checkbox inside the toggle display box will either disable or enable showing the logo. Uncheck the *Logo* box then click the **Save configuration** button at the bottom of the window. Click the home page icon once again. Voila! No more logo.

We're almost ready to start building our blog in earnest but before we do there's one more thing to change. By default your Drupal install allows users to create accounts on the site. Since the goal of this effort is to build a blog site and not a membership community we want to turn off this capability. In order to do that you first want to click on the **Configuration** menu link. The top left box on the page that appears is called *People*. Click on the very first link in that box called *Account settings*. The window that appears will look like the image below.

Building Your Blog With Drupal 7

Account settings

SETTINGS MANAGE FIELDS MANAGE DISPLAY

Home » Administration » Configuration » People

ANONYMOUS USERS

Name *

Anonymous

The name used to indicate anonymous users.

ADMINISTRATOR ROLE

Administrator role

administrator

This role will be automatically assigned new permissions whenever a module is enabled. Changing this setting will not affect existing permissions.

REGISTRATION AND CANCELLATION

Who can register accounts?

☒ Administrators only

☐ Visitors

☐ Visitors, but administrator approval is required

☒ Require e-mail verification when a visitor creates an account.

New users will be required to validate their e-mail address prior to logging into the site, and will be assigned a system-generated password. With this setting disabled, users will be logged in immediately upon registering, and may select their own passwords during registration.

Figure 8: Account Settings

Under the Registration and Cancellation section you will see the “Who can register account?” options. Select the *Administrators only* option then scroll to the bottom of the page and click the **Save configuration** button.

Installing Necessary Modules

While it’s true that Drupal can do a lot with just the core modules we will need to install number of *contributed* (as in contributed by the Drupal community) modules so that we can create a full-featured blog. This section will tell you which modules are needed and how to install them on your Drupal site.

Module Installation Basics

Thanks to recent changes in Drupal 7 module installation for novice users has become much easier. Rather than having to use FTP to upload modules you can now just link to

Building Your Blog With Drupal 7

the source file and Drupal will do the rest. If you want to install a new module the first thing you should do is click on the **Modules** link in the administration menu. At the top of the modules screen you will see a link called *Install new module*. If you click on that link you will see the window in the image below.

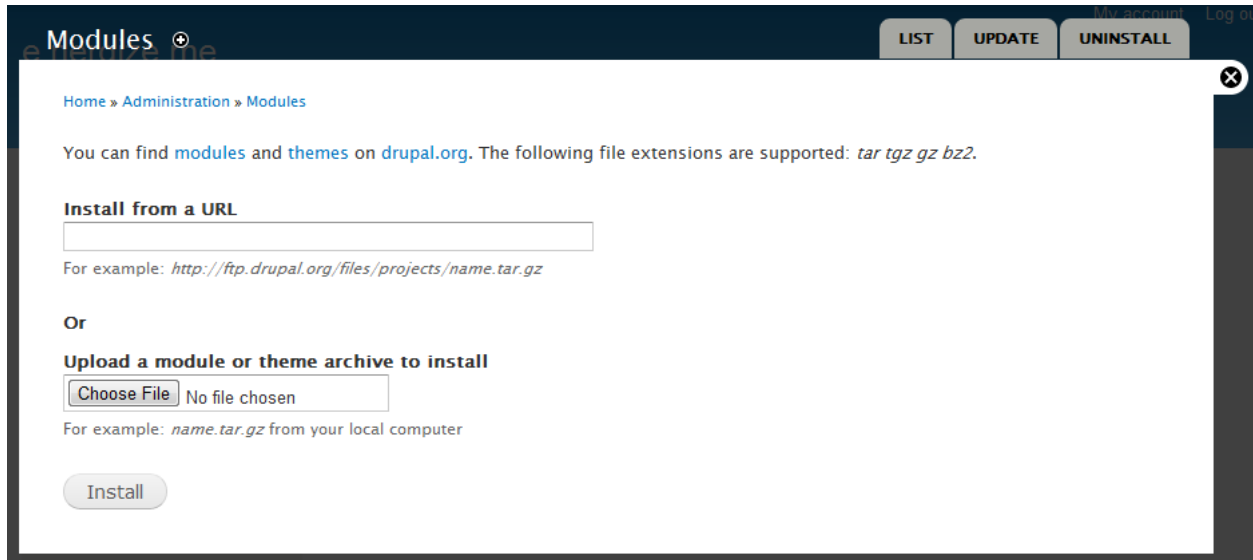


Figure 9: Install Module

Once you have navigated to this screen you need the link to a Drupal contributed module file in order to do anything useful. We will install Drupal's [Token module](#) (it's also happens to be one that we need) to help to familiarize you with the process of installation.

1. Navigate to the module page (in this case <http://drupal.org/project/token>) on Drupal.org;
2. Copy the direct link to the appropriate version² as identified by the version code in the *Recommended releases* section of the page. For example, in the picture below you want to copy the link to the 7.x-1.0-beta2 release download. You would do that by right clicking the tar.gz link (to the right of 7.x-1.0-beta2) and then copying the link;

² Drupal version numbers start with the Drupal release version so modules that are compatible with Drupal 7 will have a version code like 7.x-1.x.

Building Your Blog With Drupal 7

Recommended releases

Version	Downloads	Date	Links
7.x-1.0-beta2	tar.gz (31.91 KB) zip (35.71 KB)	2011-May-19	Notes
6.x-1.16	tar.gz (34.85 KB) zip (42.56 KB)	2011-May-18	Notes

Development releases


Version	Downloads	Date	Links
7.x-1.x-dev	tar.gz (33.12 KB) zip (37 KB)	2011-May-31	Notes
6.x-1.x-dev	tar.gz (35.06 KB) zip (42.79 KB)	2011-Jun-01	Notes

[View all releases](#) 

Figure 10: Module Releases

3. Paste the copied link into the *Install from a URL* field on your Drupal site and then click the **Install** button;
4. If everything went well (it should have!) you will see an “install success” message like the one in the image below;

Update manager

 Installation was completed successfully.

token

- Installed *token* successfully

Next steps

- [Enable newly added modules](#)
- [Administration pages](#)

Figure 11: Install Success

5. Click on the **Enable newly added modules** link, which will take you back to the module administration page;
6. Find the Token module, click the checkbox next to the module, and click the **Save configuration** button.

Building Your Blog With Drupal 7

With that final step you have completed the installation of the Token module. The next thing you need to do is enable the module. Go back to the modules window and find the *Token module* in the list. (*Hint: If you collapse the core module list you can find Token in a group called “Other”*) Select the *Enabled* checkbox next to Token and click the **Save configuration** button. You should see a message like the one in the image below.

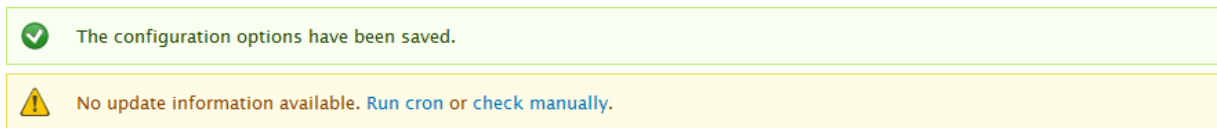


Figure 12: Enabled Module Message

You'll notice that there is a yellow warning sign and message below the green check indicating that the module was enabled. That's part of Drupal's built in update system that will alert you when either modules and themes are out of date or haven't been checked in awhile. You don't need to click anything now but if you click check manually the message should go away.

The typical next steps to take after installing and enabling a module involve adjusting the module *configuration* and *permissions*. Some modules don't require any adjusting but it's always good to check. In the case of the Token module neither of those things requires adjusting. What we'll do next is install all the modules we need first and then deal with configuration and permissions as we set up the features that those modules make possible.

Installing Necessary Modules

Now it's time to go shopping for the rest of the modules we're going to need to create our full-featured blog with Drupal 7. Follow the instructions in the previous section to install and enable the following list of modules. Once that is complete we will get down to the business of building your blog.

Building Your Blog With Drupal 7

Module Name/Link	Purpose
Pathauto	Allows you to choose the structure of the path to blog posts.
WYSIWYG	Adds the ability to create and edit posts in a <i>what you see is what you get</i> editor.
Media	For easy insertion of images and other media into your posts.
Media: YouTube	Adds media module support for embedded YouTube videos.
Webform	Used to create a contact form whose results are sent via email.
Mollom	For protection against spammers.
ShareThis	Used for easy sharing of your blog posts.
Views	Lets you create custom lists (pages, blocks, RSS feeds) of your blog posts.
Chaos tool suite	This is required in order to use the Views module.
Tagadelic	Used to create the tag cloud that points users to blog topic pages.

Table 1: Necessary Contributed Modules

If you have installed and enabled all of the modules listed above you are now ready to get down to business building your blog in earnest.

Building Your Blog

There are several steps that we will take to build your blog. The first thing we need to do is create a special “blog” content type. To do that you need to select the **Structure** option in the Administration menu. Then you want to select the second link from the top which is the **Content types** link. Finally you want to click the **Add content type** link at the top of the next window. The next window you see should look like the image below.

Building Your Blog With Drupal 7

Content types +

Home » Administration » Structure » Content types

Individual content types can have different fields, behaviors, and permissions assigned to them.

Name *

The human-readable name of this content type. This text will be displayed as part of the list on the *Add new content* page. It is recommended that this name begin with a capital letter and contain only letters, numbers, and spaces. This name must be unique.

Description

Describe this content type. The text will be displayed on the *Add new content* page.

Submission form settings

Title

Publishing options
Published , Promoted to front page

Display settings
Display author and date information.

Comment settings
Open, Threading , 50 comments per page

Menu settings

Title field label *

Title

Preview before submitting

☐ Disabled

☒ Optional

☐ Required

Explanation or submission guidelines

This text will be displayed at the top of the page when creating or editing content of this type.

Save content type Save and add fields

Figure 13: New Content Type

Fill out the Name field with the value “Blog Post”. We won’t use the name “Blog” because name could potentially conflict with the core blog module if it is ever enabled.³ It’s not necessary to fill out the *Description* field although you could enter whatever you

³ You may want to know why we’re not using the core Blog module to create the blog. The reason is that the core blog module offers less flexibility than a custom created content type. So I think that the custom content type is a better choice for a single site-wide blog.

Building Your Blog With Drupal 7

like there. The next step to take is to configure the settings in the bottom section which has a tabbed interface with *Submission form settings* at the top.

You can leave the *Submission form settings* as they are and move to *Publishing options* by clicking on the tab. Once there uncheck the *Published* and *Promoted to front page* checkboxes. I like to have the save action not publish any post by default so that's why *Published* is unchecked. We won't "promote" the posts to the front page automatically because we're going to create our own front page using the Views module later.

The next three settings below publishing options, *Display settings*, *Comment settings*, and *Menu settings* don't require any changes. I think the options are pretty self explanatory there so take a look and make changes if you feel they are necessary. The Menu settings will create a menu option for every post if you check it so you probably don't want to check that option. You'll be ready to move on to the next steps once you click the **Save content type** button.

If you click the **Content** admin menu option and then select the **Add content** link at the top of the page you should see a link that says "Blog post" on the window that opens up.



Figure 14: Add Content Window

Feel free to click through to see what your blog post content type looks like. The content editing screen is still missing a few things that we need to add in order to start creating blog posts. The two things we'll tackle next are the categories, which will be used to create a tag cloud and the WYSIWYG editor.

Building Your Blog With Drupal 7

Setting Up Category Tags

Category tags allow you to freely categorize your content by typing words in a field on content editing screen. We'll use the tags to create a tag "cloud" that your visitors can click on to go directly to the topics that interest them most. This feature is enabled via the combination of the core Taxonomy module and the contributed Tagadelic module which we added in the previous section.

Click on **Structure** in the admin menu and then click the **Taxonomy** link. The next screen you see will have an *Add vocabulary* link at the top of the page. Click that link and give it the name "Tags" and the description "Blog post topics." Then click the **Save** button. You'll be returned to the taxonomy window but now you will see your new vocabulary listed as shown in the image below.

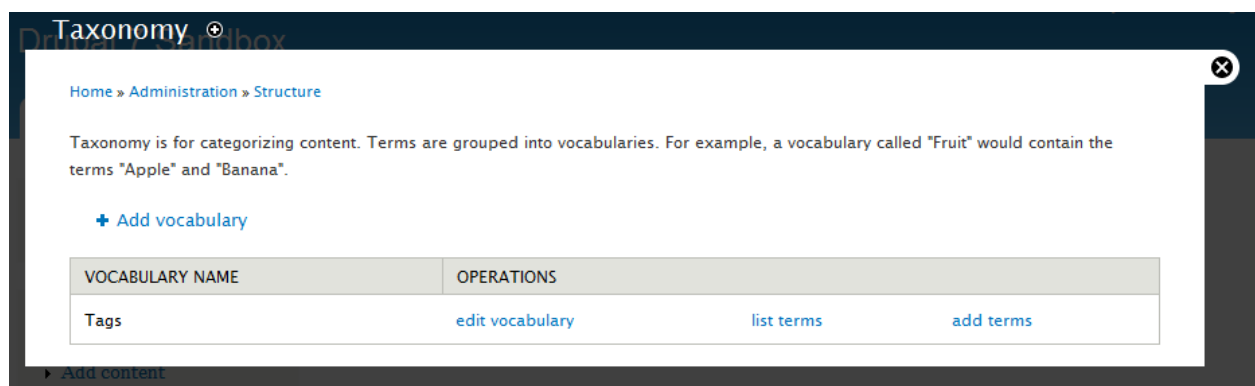


Figure 15: Taxonomy List Window

Now that you have created your vocabulary for your blog topics you need to add the vocabulary field to the *Blog Post* content type so that the field will appear when you are editing content. Click on the **Structure** link in the admin menu and then click the **Content Types** link. You should see something that looks like the image below.

Building Your Blog With Drupal 7

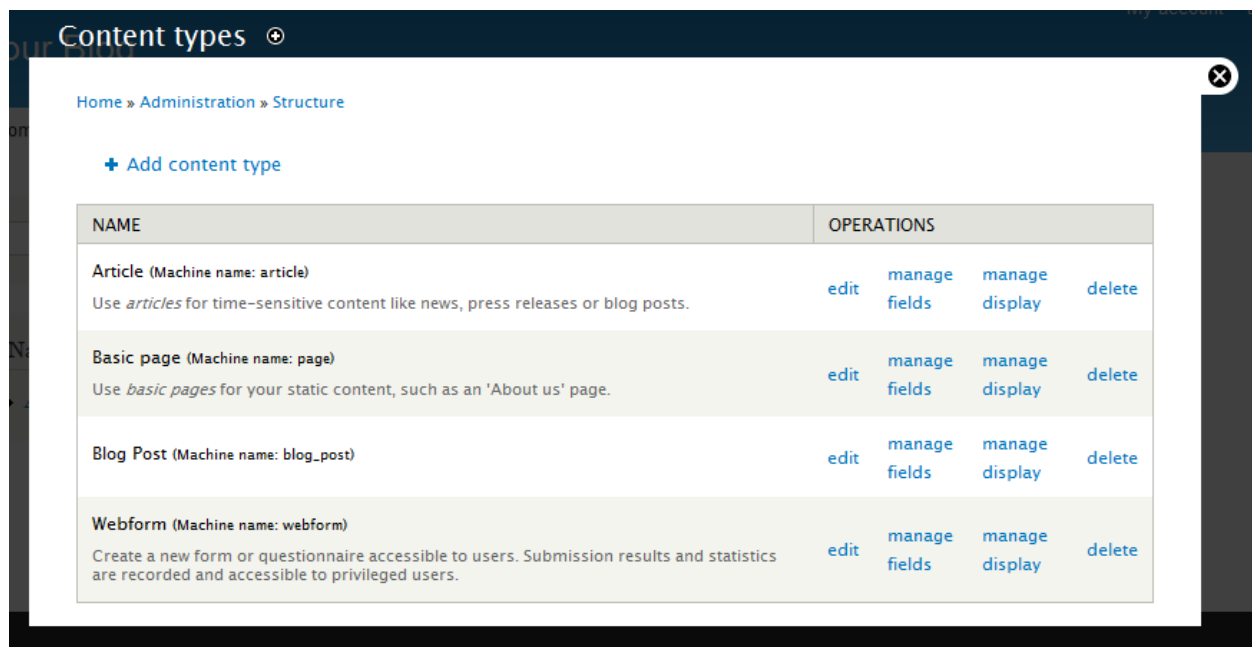


Figure 16: Content Types

In the row that says “Blog Posts” click on the **manage fields** link. In the resulting window add a new field by filling out the empty fields as in the image below.

Building Your Blog With Drupal 7

Blog Post [EDIT](#) [MANAGE FIELDS](#) [MANAGE DISPLAY](#) [COMMENT FIELDS](#) [COMMENT DISPLAY](#) [My account](#)

[Home](#) » [Administration](#) » [Structure](#) » [Content types](#) » [Blog Post](#) [Show row weights](#)

LABEL	NAME	FIELD	WIDGET	OPERATIONS
+ Title	title	Node module element		
+ URL path settings	path	Path module form elements		
+ Body	body	Long text and summary	Text area with a summary	edit delete

[+](#) **Add new field**

field_ - Select a field type - - Select a widget -
Label Field name (a-z, 0-9, _) Type of data to store. Form element to edit the data.

[+](#) **Add existing field**

Tags Term reference: field_tags (Tags) Autocomplete term widget (tagging)
Label Field to share Form element to edit the data.

[Save](#)

Figure 17: Blog Post Fields


Once you have filled in the fields click the **Save** button. The next window you will see allows you to set up the tagging field. There are just a few things you need to do in this window.





1. Check the *Required field* check box;
2. Add *Help text* to describe the field if you like;
3. Click the Save settings button.


Once you have completed those steps you will be returned to the Blog Post Fields window. You'll notice that the Tags field is at the bottom of the list of fields. That's where the field will show up in the editing window as well. I like to have the Tags field above the body field so I recommend that you click and drag the field using the cross (+) sign on the far left of the Tags field. The image below shows you what the window should look like after you drag the field.

Building Your Blog With Drupal 7

[Show row weights](#)


 * Changes made in this table will not be saved until the form is submitted.


LABEL	NAME	FIELD	WIDGET	OPERATIONS
 Title	title	Node module element		
 URL path settings	path	Path module form elements		
 Tags*	field_tags	Term reference	Autocomplete term widget (tagging)	edit delete
 Body	body	Long text and summary	Text area with a summary	edit delete



Add new field

Label


field_
Field name (a-z, 0-9, _)

- Select a field type - 
Type of data to store.

- Select a widget - 
Form element to edit the data.


Add existing field

Label

- Select an existing field - 
Field to share


- Select a widget - 
Form element to edit the data.

Figure 18: Dragging Fields

Click the **Save** button to finish this task. Congratulations, you have completed setting up your category tags. If you click the **Add content** link in the shortcut bar (below the admin bar) then click the **Blog Post** link you should see a window that looks like the image below.

Building Your Blog With Drupal 7

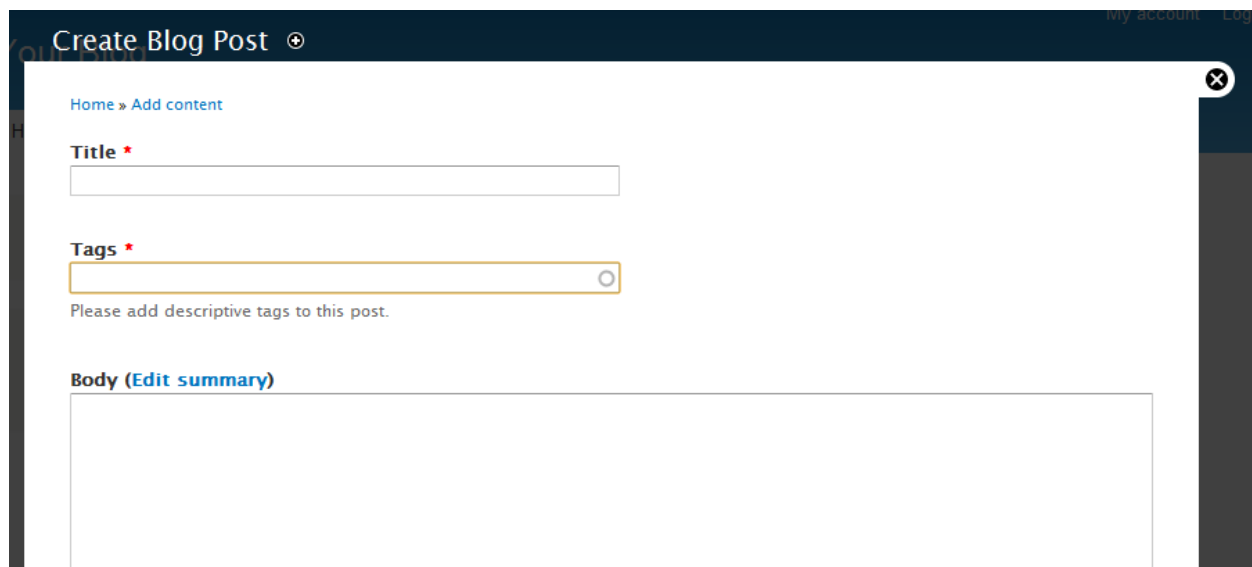


Figure 19: Create Blog Post

Notice how the Tags field is above the Body field in the position where we dragged it to when setting up the field. Also notice the red asterisks (*) that indicate the field is required.

Adding A WYSIWYG Editor

It's time to continue setting up our editing environment by adding a WYSIWYG editor. Although this type of editor is not required to start blogging it sure does make formatting text and adding images to your posts easier. You will recall that earlier we installed the WYSIWYG module. Navigate to the settings for that module by selecting **Configuration** from the admin menu, and then **WYSIWYG profiles** which is in the Content Authoring section. You will see a screen that looks like the image below with instructions for installing the editor code for a wide variety of WYSIWYG editors. While you're free to install your favorite editor I'm going to install the TinyMCE editor.

Building Your Blog With Drupal 7

[Home](#) » [Administration](#) » [Configuration](#) » [Content authoring](#)

A Wysiwyg profile is associated with an input format. A Wysiwyg profile defines which client-side editor is loaded with a particular input format, what buttons or themes are enabled for the editor, how the editor is displayed, and a few other editor-specific functions.

▼ INSTALLATION INSTRUCTIONS

There are no editor libraries installed currently. The following list contains a list of currently supported editors:

[CKEditor \(Download\)](#)

Not installed.

Extract the archive and copy its contents into a new folder in the following location:

`sites/all/libraries/ckeditor`

So the actual library can be found at:

`sites/all/libraries/ckeditor/ckeditor.js`

[FCKeditor \(Download\)](#)

Not installed.

Extract the archive and copy its contents into a new folder in the following location:

`sites/all/libraries/fckeditor`

So the actual library can be found at:

`sites/all/libraries/fckeditor/fckeditor.js`

Figure 20: WYSIWYG Profiles

The image below shows you exactly what the instructions for TinyMCE look like. You'll notice a link to download the TinyMCE libraries and instructions on where to put the unzipped files.

[TinyMCE \(Download\)](#)

Not installed.

Extract the archive and copy its contents into a new folder in the following location:

`sites/all/libraries/tinymce`

So the actual library can be found at:

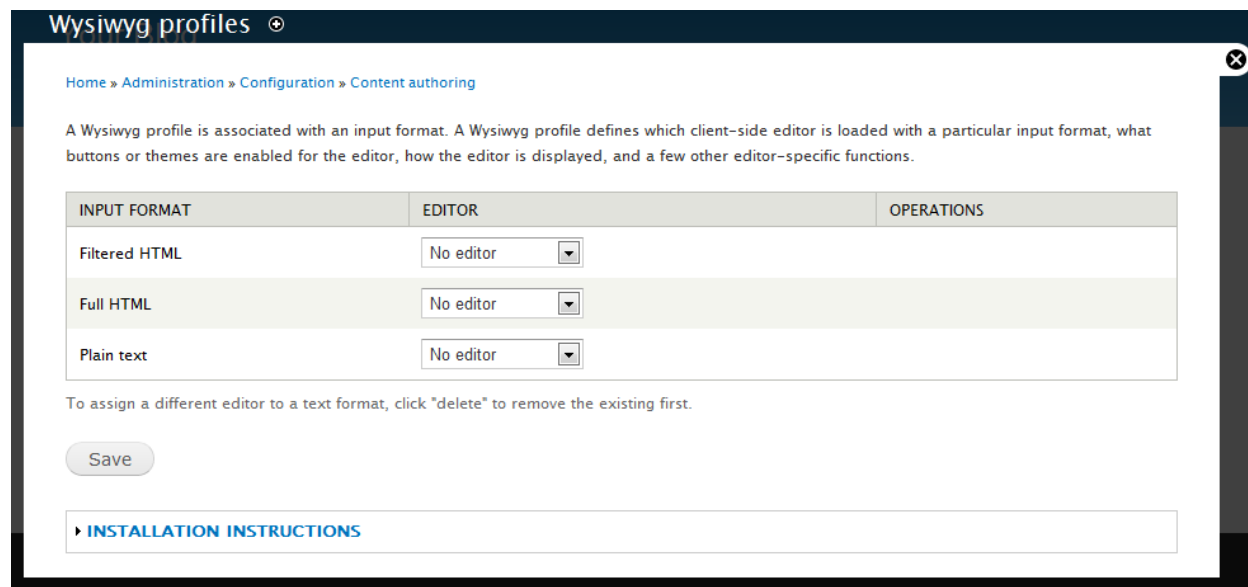
`sites/all/libraries/tinymce/jscripts/tiny_mce/tiny_mce.js`

Figure 21: TinyMCE WYSIWYG Profile

When you download the most recent version of the TinyMCE libraries [from the website](#) you will get a .zip file which you should unzip using your utility of choice. Once the file is unzipped you will have a folder called *tinymce*. Per the WYSIWYG instructions you should upload that entire folder to the *sites/all/libraries* directory. That directory won't exist yet so you should create the folder then deposit the tinymce folder there. Once you

Building Your Blog With Drupal 7

have done that return to the WYSIWYG profile page and refresh the page. Things should look like the image below with an option to select an editor for each text format.



Wysiwyg profiles

Home » Administration » Configuration » Content authoring

A Wysiwyg profile is associated with an input format. A Wysiwyg profile defines which client-side editor is loaded with a particular input format, what buttons or themes are enabled for the editor, how the editor is displayed, and a few other editor-specific functions.

INPUT FORMAT	EDITOR	OPERATIONS
Filtered HTML	No editor	
Full HTML	No editor	
Plain text	No editor	

To assign a different editor to a text format, click "delete" to remove the existing first.

Save

▶ [INSTALLATION INSTRUCTIONS](#)

Figure 22: WYSIWYG Editor Options

Click on the list box next to Full HTML and select the TinyMCE option then click the Save button. You'll notice that *Edit* and *Delete* buttons appear under the Operations section of the window. Click on the **Edit** button to customize the look and feel of TinyMCE. You will see links to expand five different sections as follows.

1. Basic Setup
2. Buttons and Plugins
3. Editor Appearance
4. Cleanup and Output
5. CSS


Click on the **Buttons and Plugins** section and select the user interface elements that you would like to appear in your editor. The options, like Bold, Italic, Underline, etc. should be self explanatory. You can always make changes later by returning to this window if a particular element that you want is missing. When you're done click the **Save** button.




Building Your Blog With Drupal 7

Adjusting Your Text Formats

Although you have successfully added a WYSIWYG editor you now must adjust your text formats in order to have the editor show up automatically when you get ready to write a new blog post. Text formats are a very important element of Drupal because they allow you to determine what type of content can be entered into any particular field. For the blog post editor we'll choose to apply the Full HTML option that will allow us to use the full range of features of TinyMCE. A more restrictive format such as Plain Text is helpful for commenting and any anonymous input because it prevents people from entering potentially harmful code or embedding images that we don't want. Twitter and Facebook are examples of popular sites that require plain text input.

Since you trust yourself to post what you like we're going to setup the site to allow administrators to use Full HTML and thus the TinyMCE editor. Click **Configuration** on the admin menu and then select the **Text formats** link in the Content Authoring section. You will see three text formats on the following window, Filtered HTML, Full HTML and Plain text. Click on the cross (+) to the left of *Full HTML* and drag it above *Filtered HTML*. The screen should then look like it does in the image below.

 * Changes made in this table will not be saved until the form is submitted.

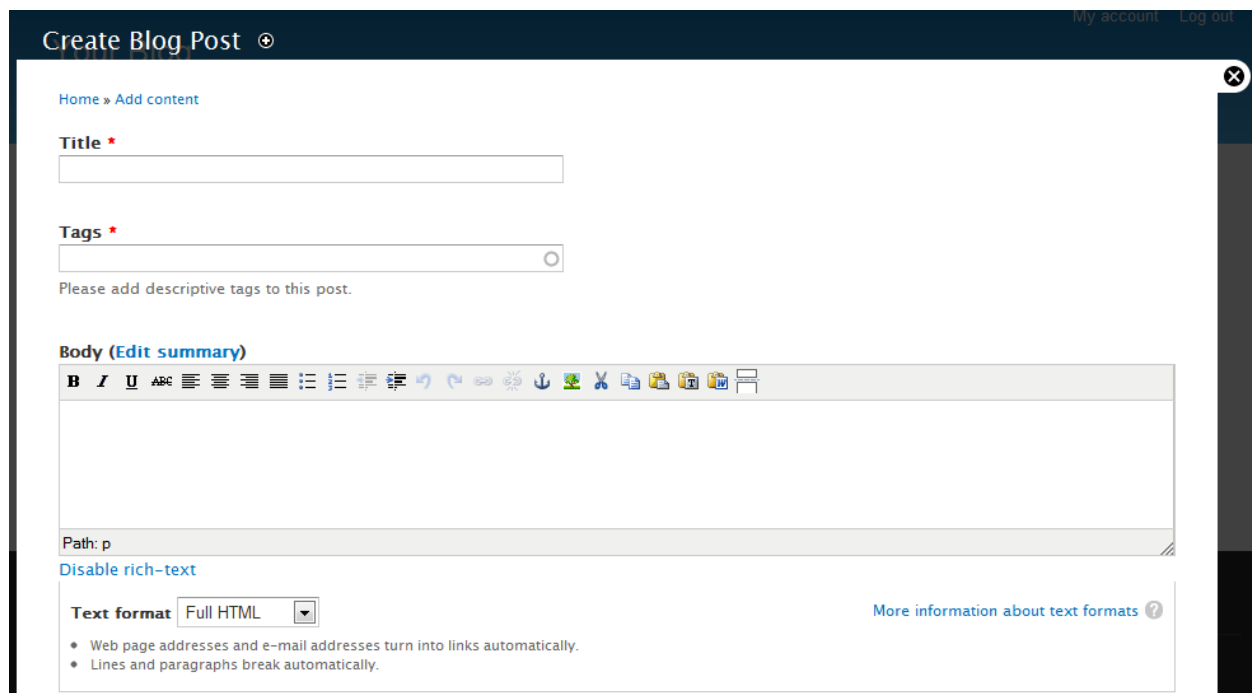
NAME	ROLES	OPERATIONS	
 Full HTML*	administrator	configure	disable
 Filtered HTML	anonymous user, authenticated user, administrator	configure	disable
 Plain text	All roles may use this format	configure	

Save changes

Figure 23: Text Formats

All you need to do is click the **Save changes** button and you're done. Click on **Content** on the admin menu, then click **Add content**, and then select **Blog Post**. The Create Blog Post window should look very similar to what you see in the image below. Notice the buttons that allow you to format text with a click.

Building Your Blog With Drupal 7



Create Blog Post

Home » Add content

Title *

Tags *

Please add descriptive tags to this post.

Body (Edit summary)

Path: p

Disable rich-text

Text format: Full HTML

More information about text formats

- Web page addresses and e-mail addresses turn into links automatically.
- Lines and paragraphs break automatically.

Figure 24: TinyMCE Body Field

If you don't see the WYSIWYG editor go back over the previous steps once again to ensure that you didn't miss anything. Otherwise give yourself a pat on the back for a job well done.

Create A Shortcut To Add Blog Posts

Now is a good time to help make things easier for you. You probably noticed that it took several clicks to get to the *Create Blog Post* window. That's way too many clicks in my opinion. Thankfully Drupal 7 has a customizable shortcut bar below the administration menu. You'll notice that it already has links for *Add content* and *Find content*. Click the plus sign (+) to the right of the *Create Blog Post* page title. Refresh the page and you should now see a Blog Post shortcut added as in the image below. Now you can start to create a new post with just one click.

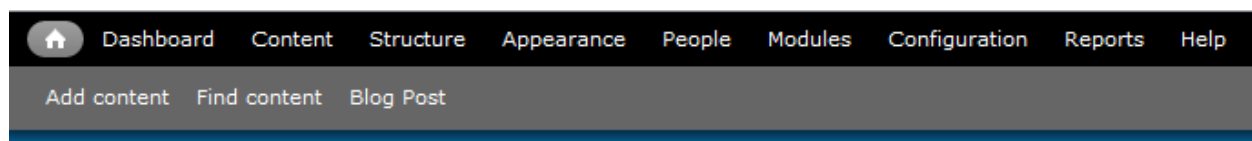


Figure 25: Blog Post Shortcut

Building Your Blog With Drupal 7

Customizing Post Paths

At this point we're very close to the point where you can start creating some sample content. Before we do you should set up one more very important feature. You may recall that one element of our blog will be nice paths (or "URLs") that include the date posted and title of the post. Earlier you installed the PathAuto and Token modules so you should be all ready to set this up.

Click on **Configuration** in the admin menu and in the Search and Metadata section select the **URL Aliases** link. In the upper right corner of the URL Aliases window there are five tabs: List, Patterns, Settings, Bulk Update and Delete Aliases. Click on the **Settings** tab and you should see the image below.

Building Your Blog With Drupal 7

[Home](#) » [Administration](#) » [Configuration](#) » [Search and metadata](#) » [URL aliases](#)

☐ **Verbose**
Display alias changes (except during bulk updates).

Separator

Character used to separate words in titles. This will replace any spaces and punctuation characters. Using a space or + character can cause unexpected results.

Character case
☐ Leave case the same as source token values.
☒ Change to lower case

Maximum alias length

Maximum length of aliases to generate. 100 is the recommended length. 255 is the maximum possible length. See [Pathauto help](#) for details.

Maximum component length

Maximum text length of any component in the alias (e.g., [title]). 100 is the recommended length. 255 is the maximum possible length. See [Pathauto help](#) for details.

Update action
☐ Do nothing. Leave the old alias intact.
☐ Create a new alias. Leave the existing alias functioning.
☒ Create a new alias. Delete the old alias.
What should Pathauto do when updating an existing content item which already has an alias? Considering installing the [Redirect module](#) to get redirects when your aliases change.

☐ **Reduce strings to letters and numbers**
Filters the new alias to only letters and numbers found in the ASCII-96 set.

Strings to Remove

[a](#), [an](#), [as](#), [at](#), [before](#), [but](#), [by](#), [for](#), [from](#), [is](#), [in](#), [into](#), [like](#), [of](#), [off](#), [on](#), [onto](#), [per](#), [since](#), [than](#), [the](#), [this](#), [that](#), [to](#), [up](#), [via](#), [with](#)

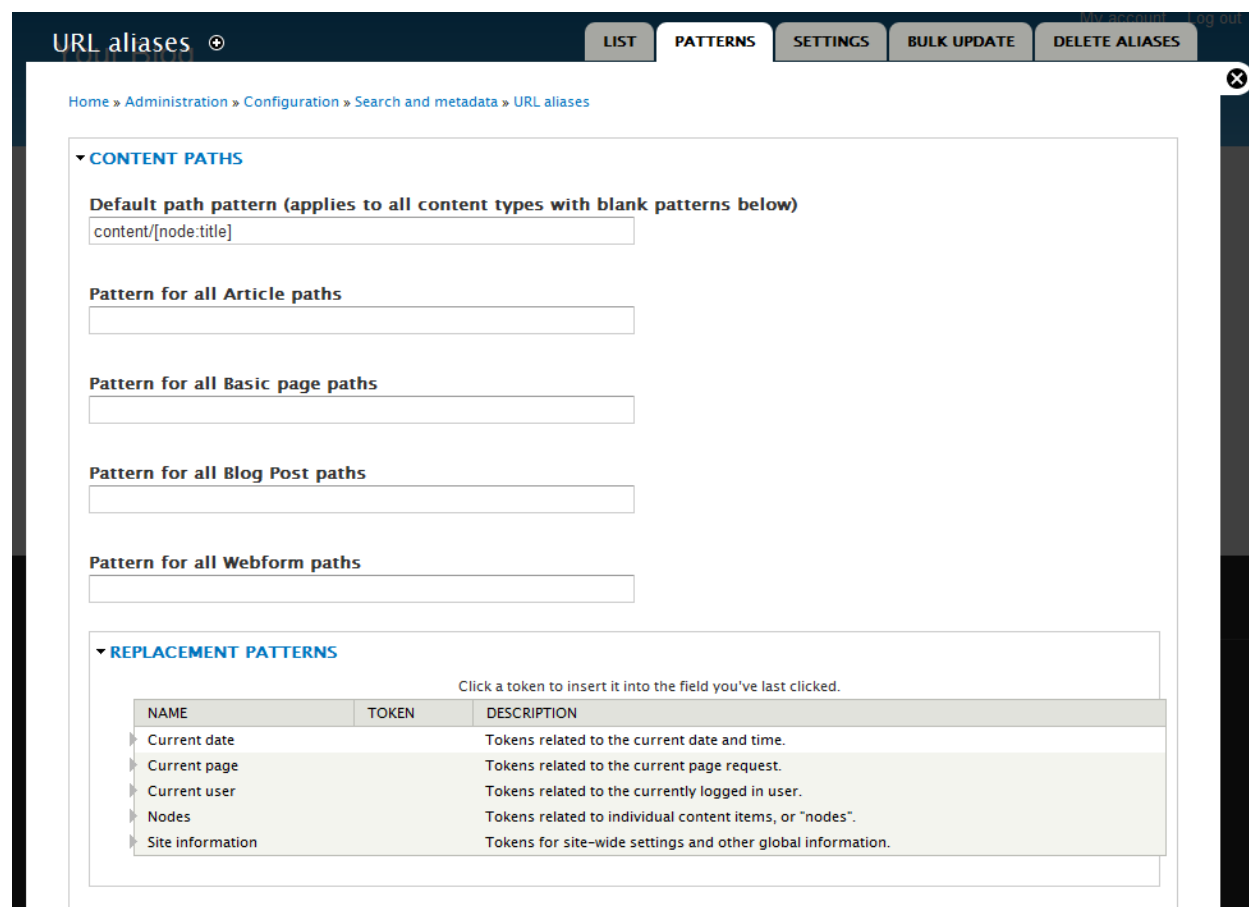
Figure 26: URL Alias Settings

Take note of the various options here. I like to delete the values from the “Strings to Remove” field. With the default settings a post title like “Here’s To Drupal” will end up with the string “heres-drupal” in the title instead of “heres-to-drupal”. Whether or not you want to remove those strings is up to you. If you do delete the values click the **Save configuration** button to make it stick.

The next tab to click on is the **Patterns** tab. This window is very important because this is where you will set up the automatic paths to your blog posts, pages and categories. The two most important elements of this window are the default *Paths* and the *Replacement Patterns*. You’ll notice in the image below that there is a *Default path pattern* set. As it is set up if you create a page titled “Hello” the path to the page will be

Building Your Blog With Drupal 7

set to `yourblog.com/content/hello`. Since I prefer not to have the word “content” in my paths I recommend that you delete the “content/” portion of that field.



The screenshot shows the 'URL aliases' configuration page in Drupal 7, with the 'PATTERNS' tab selected. The breadcrumb trail is 'Home » Administration » Configuration » Search and metadata » URL aliases'. The page is divided into two main sections: 'CONTENT PATHS' and 'REPLACEMENT PATTERNS'.

CONTENT PATHS

Default path pattern (applies to all content types with blank patterns below)

`content/[node:title]`

Pattern for all Article paths

Pattern for all Basic page paths

Pattern for all Blog Post paths

Pattern for all Webform paths

REPLACEMENT PATTERNS

Click a token to insert it into the field you've last clicked.

NAME	TOKEN	DESCRIPTION
Current date		Tokens related to the current date and time.
Current page		Tokens related to the current page request.
Current user		Tokens related to the currently logged in user.
Nodes		Tokens related to individual content items, or "nodes".
Site information		Tokens for site-wide settings and other global information.

Figure 27: URL Alias Patterns

Your *Default path pattern* should now be “[node:title]”. You may be wondering what this [node:title] field is all about. That’s what’s called a replacement pattern. The pattern tells the system what content to insert in the path when a post is created. In this case the system will add the title of the node to the path. Each of the many replacement patterns is described in the *Replacement Patterns* section at the bottom of the image above.

Take a few moments to explore the many different options. For the moment all you will need to do is understand the correct elements to input into the *Blog Post paths* section.

I like to have blog post paths that include the created date in yyyy/mm/dd format and then the title of the post. For example, if I create a post on January 1, 2011 with a title of

Building Your Blog With Drupal 7

“Hello World” the path I prefer would be `yourblog.com/2011/01/01/hello-world`. Here is the exact string (without the “”) that you should enter into the Blog post paths field to achieve that type of path,

`[node:created:custom:Y]/[node:created:custom:m]/[node:created:custom:d]/[node:title]`”.

If you want to have the path include the word “blog” you would simply add the word blog and a slash (“blog/”) to the field. If you make that change be sure to click the **Save configuration** button.

It’s important to point out that the paths to your *Taxonomy Terms* are also controlled via this window. The default pattern is `[term:vocabulary]/[term:name]`. So if you add the tag “berry” to a blog post the direct link to that tag will be `yourblog.com/tags/berry`. You could replace either of those patterns with a word or another replacement pattern. For example if you want your path to be `yourblog.com/categories/berry` the replacement pattern would be “categories/[term:name]”.

Creating Content

Now that the paths are set it’s a good time to create your first blog post. Click the *Blog Post* link in the shortcut bar and it will take you right to the *Create Blog Post* window. Fill out the fields as follows:

Title: Hello World

Tags: test

Body: Hello world. This is my first post.

Click the *Publishing options* tab (it’s the bottom one) and check the box next to the word *Published*. Then click the **Save** button. The resulting page should look like the image below.

Building Your Blog With Drupal 7

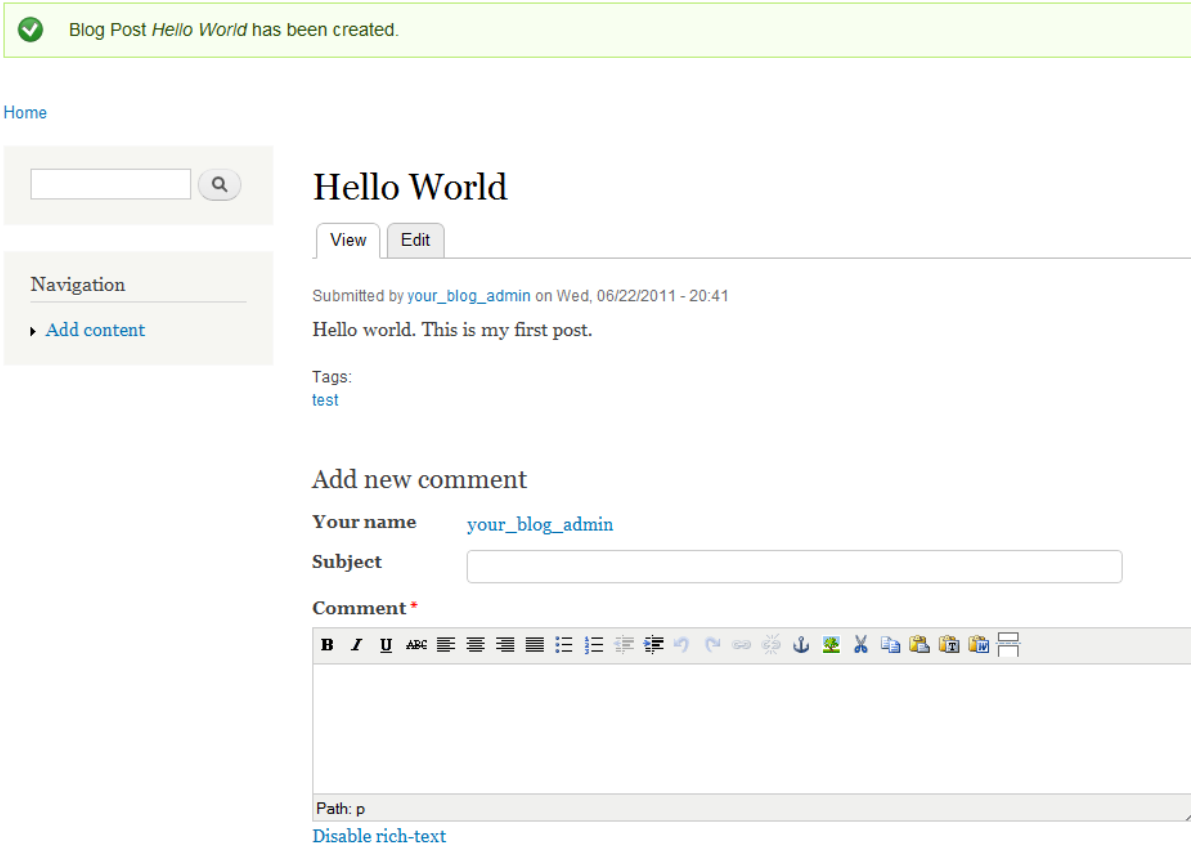


Figure 28: Hello World Post

Congratulations on creating your first item of content. We'll dive into the details of this post in a little while. The next thing you should do is return to the home page of your site. You'll notice that the front page still says "Welcome to Your Blog" and the post you created is nowhere to be found. We'll work on getting the posts on the front page shortly as well. It's important to know that you can always find content that has been created by clicking the **Content** link on the admin menu. If you do that you should see a window that looks like the following image.

Building Your Blog With Drupal 7

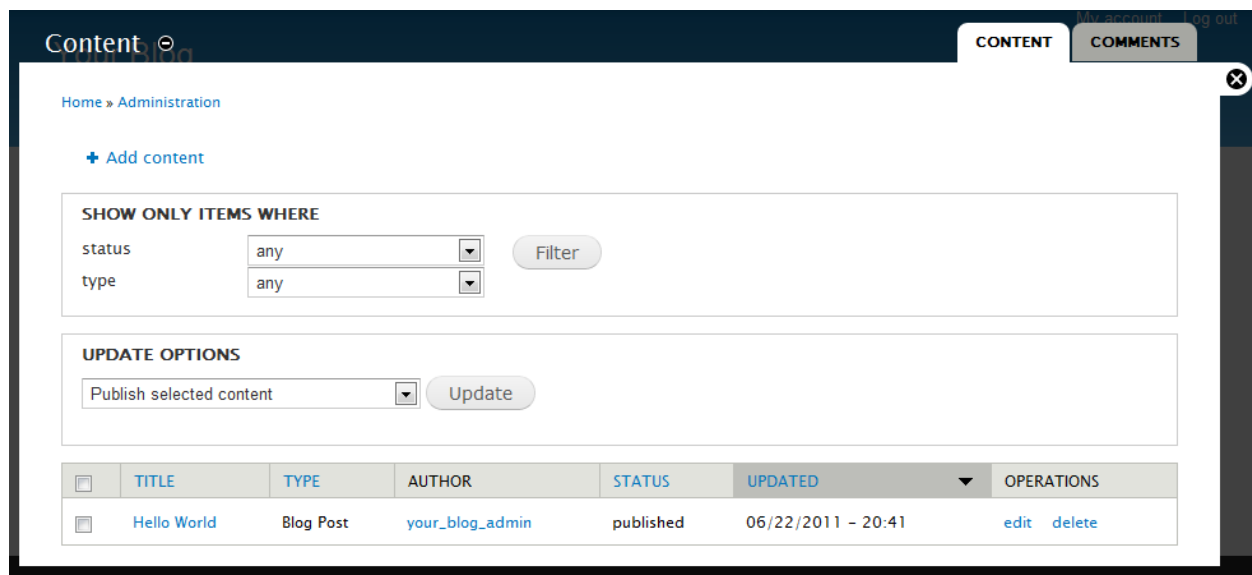


Figure 29: Content View

This window shows you content of different types (like blog posts vs. pages) and of different statuses (like published vs. not published). You can use the list boxes and check boxes to perform operations such as filtering the list of content items and publishing, unpublishing or deleting content. You don't have to do anything more in this window right now but it's good to know where you can find it and what it's used for.

Getting Your Blogs Posts Onto The Front Page

The next things you're going to do are to set up a "view" (which is basically a list) for your blog posts and make that view the front page of your site. It's true that there are simpler ways to do this, namely clicking the **Promoted to front page** checkbox in the add content window, but Views is so integral to Drupal that I think you should get started using it sooner rather than later.

To get started click on the **Structure** link in the admin menu, then select the *Views* link which should be the last one in the Structure window. At the very top of the next window you will see an *Add new view* link. When you click that link you should see a window that looks like the image below.

Building Your Blog With Drupal 7

The screenshot shows the 'Add new view' form in Drupal 7. The form is titled 'Add new view' with a plus icon. In the top right corner, there are links for 'My account' and 'Log out', and buttons for 'LIST' and 'SETTINGS'. The breadcrumb trail is 'Home » Administration » Structure » Views'. The form contains the following fields and options:

- View name ***: A text input field.
- Description**: A checkbox.
- Show**: A dropdown menu with 'Content' selected.
- of type**: A dropdown menu with 'All' selected.
- tagged with**: A text input field.
- sorted by**: A dropdown menu with 'Newest first' selected.
- Create a page**: A checked checkbox.
- Page title**: A text input field.
- Path**: A text input field with the value 'http://localhost/yourblog/'.
- Display format**: A dropdown menu with 'Unformatted list' selected.
- of**: A dropdown menu with 'teasers' selected.
- with links (allow users to add comments, etc.)**: A dropdown menu with 'with links (allow users to add comments, etc.)' selected.
- without comments**: A dropdown menu with 'without comments' selected.
- Items per page**: A text input field with the value '10'.
- Create a menu link**: A checkbox.
- Include an RSS feed**: A checkbox.
- Create a block**: A checkbox.

At the bottom of the form, there are three buttons: 'Save & exit', 'Continue & edit', and 'Cancel'.

Figure 30: Add New View

There are a lot of choices to make when adding a new view so I'll make it easy for you and show you the choices I have made in the image below.

Building Your Blog With Drupal 7

View name *
Blog Posts Machine name: blog_posts [Edit]

☐ **Description**

Show Content of type Blog Post tagged with sorted by
Newest first

☒ **Create a page**

Page title
My Blog

Path
http://localhost/yourblog/blog

Display format
Unformatted list of teasers with links (allow users to add comments, etc.) without comments

Items per page
10

☐ Create a menu link

☒ Include an RSS feed

Feed path
http://localhost/yourblog/feed

☒ **Create a block**

Block title
Recent Posts

Display format
Unformatted list of titles (linked)

Items per page
5

Figure 31: Creating A Blog View

You're free to change any of the details to suit your own needs. The most important thing to understand is that these choices will result in the following features.

1. A list of blog posts with the newest blog post displayed at the top
2. A dedicated page at yourblog.com/blog that shows ten teaser (or summary) posts that include the title, the beginning of the body field and a link to the full post

Building Your Blog With Drupal 7

3. A dedicated RSS feed that people can use to subscribe to the blog in an RSS reader
4. A block that you can place in the sidebar that has links to the five most recent posts

Once you have filled in your choices click the **Save and exit** button to complete the creation of your feed. The screen will refresh and you'll be looking at the new page you created at `yourblog.com/blog`. You should see the "Hello World" post you created and an orange RSS feed icon that links to the blog feed. Now that the view for your blog has been created the next thing you want to do is make that view the front page of your site.

Click on **Configuration** in the admin menu then select the **Site information** link. The second section from the top of the site information page is called "Front Page". Change the "Default front page" field from "node" to "blog" or whatever path you gave to your blog post view. Click **Save configuration** then navigate to your site home page. The front page of your site should now look something like the image below.



Figure 32: Blog As Home Page

Besides your blog posts the blocks to the left side of your posts are another important element of your site. The block you see on the top is the standard search block. I'll have

Building Your Blog With Drupal 7

you leave that in place. The next block down is the navigation block which we really don't need since we have the administration bars at the top of the page. In the next section I will show you how to remove that block and add other blocks that you do need.

Arranging Sidebar Blocks

Click on **Structure** in the admin menu and then select the **Blocks** link which is the first link. You will see a table that looks like the one in the image below. Focus on the "Sidebar first" section of the table.

[Show row weights](#)

BLOCK	REGION	OPERATIONS
Header		
<i>No blocks in this region</i>		
Help		
+ System help	Help	configure
Highlighted		
<i>No blocks in this region</i>		
Featured		
<i>No blocks in this region</i>		
Content		
+ Main page content	Content	configure
Sidebar first		
+ Search form	Sidebar first	configure
+ Navigation	Sidebar first	configure
+ User login	Sidebar first	configure
Sidebar second		

Figure 33: Block Administration Window

Click on the list box in the Navigation row and set the value to "-None-". You'll notice that Navigation disappears from the row. Scroll down to the very bottom of the page and drag the row called "blog_posts: Block" to the space below the Search form field. Then navigate to the bottom of the page and click the **Save blocks** button. Navigate to the home page and you will now see the Recent Posts block below the Search block.

Building Your Blog With Drupal 7

If you like you can also add a Recent Comments block to the sidebar. The Drupal core comments module has created this block for you. Just go to the block administration window and drag “Recent Comments” to the appropriate position or select “Sidebar first” from the list box in the Recent Comments row. Keep in mind that those changes are not saved until you select the *Save blocks* button.

The final block that I’ll have you place for now is the block that will hold your tag (or category) “cloud”. In the Block administration window look for “Tags in Tags” and move that to Sidebar first then save your blocks. If you go to the home page it should now look something like the image below.



Figure 34: Home Page With Blocks

You’ll notice that the “Tags in Tags” block has a dotted line around it with a configuration icon in the upper right portion of the block. The ability to go to the edit views of blocks directly from the block itself is a nifty new feature in Drupal 7. You should see something like that when you put your cursor directly over the block and point to the configuration icon. For all blocks you have the option of editing the title, the access levels and even the pages that they appear on. Tags in Tags happens to be a block that has an additional configuration option. You can edit the number of tags that appear in the block.

Building Your Blog With Drupal 7

If you click on the configure block link that appears when you mouse over the configuration icon you will be taken to the block edit window where, among other things, you can adjust the block title to something like “Topics” and also change the number of tags that will be displayed if you like. Remember to click the **Save** button at the bottom of the window. Once you do that you will be returned to the home page of your site.

One thing you may have noticed when looking at your block configuration is the fact that the *User login block* is placed in the *Sidebar first* section. If you don’t want anonymous users to see that blog you should remove it just like you removed the *Navigation* block. You can always reach the login page of your site by navigating to /user/login.

Creating Your About Page

An “about” page is an important feature of any website. You may need to create other relatively static pages as well. What I’m going to have you do now is create your about page and add it to your menu tabs at the top of the page.

Click the **Add content** link in the shortcut bar then select the **Basic page** link. You will see a page creation window that looks very much like the blog post window but you’ll notice that there will not be a tag field. Fill in the Title with “About” and Body with “This is a placeholder for my about page.” The next thing you should do is go down to the “Menu settings” tab and fill in the selections as indicated in the image below.

Building Your Blog With Drupal 7

Menu settings
About

Revision information
No revision

URL path settings
Automatic alias

Comment settings
Closed

Authoring information
By your_blog_admin

Publishing options
Published

☒ Provide a menu link

Menu link title
About

Description
Find out more about this blog.

Shown when hovering over the menu link.

Parent item
<Main menu>

Weight
1

Menu links with smaller weights are displayed before links with larger weights.

Save Preview

Figure 35: Page Menu Options

The chosen settings will create a page at yourblog.com/about and a menu tab that will appear after Home in the menu tab list below the header of your site. Once you have saved this page you can go back later and make edits to the content to suit the purpose of your website.

Creating Your Contact Form

Some form of general feedback is a key element to any website. For your site I'm going to show you how to create a form using Drupal's contributed Webform module.

Webform gives you the ability to create a form with results that are both emailed to you and saved in the Drupal database. As far as I'm concerned the most important consideration when creating a contact form should be how to protect that form from unwanted spam. The anti-spam solution I will show you involves pairing Webform with the Mollom anti-spam module and service.

The first thing you need to do is create a new webform. Select **Add content** in the shortcut menu and then click the **Webform** link. The window you see will look just like a page content window. Add your title, "Contact" and short description in the body field

Building Your Blog With Drupal 7

like, “Use this form to contact us.” Just like you did when creating your about page create a menu link by selecting the “Provide a menu link” checkbox and filling out the fields in the Menu settings tab. If you set the *Weight* of the menu to “2” it will appear after the about page in the menu tab.

When creating this webform you also need to pay attention to some of the other tabs. In the “Comment settings” tab select the “Closed” radio button. In the “Publishing options” tab un-check the “Promoted to the front page” check box. When you’re finished completing those fields click the *Save* button. You will see a window that looks like the one below. This is the window that you use to add individual fields that your site visitor will fill in on the form.

NAME	TYPE	VALUE	MANDATORY	OPERATIONS
No Components, add a component below.				
<input type="text" value="New component name"/>	<input type="text" value="Textfield"/>		<input type="checkbox"/>	<input type="button" value="Add"/>

Figure 36: Webform Created

Add your components one at a time, clicking add after you have added a *Name*, *Type*, and considered whether or not the field is *Mandatory*. The following image offers guidelines on the fields to add to your contact form.

Building Your Blog With Drupal 7

Home Form components E-mails Form settings

✓ The component positions and mandatory values have been updated.

Show row weights

NAME	TYPE	VALUE	MANDATORY	OPERATIONS
+ Your Name	textfield	-	<input checked="" type="checkbox"/>	Edit Clone Delete
+ Your Email Address	email	-	<input checked="" type="checkbox"/>	Edit Clone Delete
+ Subject	textfield	-	<input checked="" type="checkbox"/>	Edit Clone Delete
+ Message	textarea	-	<input checked="" type="checkbox"/>	Edit Clone Delete
+ <input type="text" value="New component name"/>	<input type="text" value="Textfield"/>		<input type="checkbox"/>	Add

Save

Figure 37: Webform Components

After clicking the **Add** button for each field you will be presented with a window that allows you to fine tune the settings for each field. For now the one field to pay attention to in that window is the “Description” field. The description will appear below the field to give the visitor a little information about the purpose of that field.

The next step you need to take is to configure the email address that the form will be sent to after submission by the site visitor. Click on the **E-mails** link to the right of Form components. Enter the email you want to have email sent to on the “Address” field then click the **Add** button. At that point you will be directed to a window (see image below) that allows you to customize the look of the email sent by your Drupal site.

Building Your Blog With Drupal 7

Home

Edit e-mail settings

E-mail to address

☒ Custom:

☐ Component:

Form submissions will be e-mailed to this address. Any email, select, or hidden form element may be selected as the recipient address. Multiple e-mail addresses may be separated by commas.

E-MAIL HEADER DETAILS

E-mail subject

☐ Default: *Form submission from: Contact*

☐ Custom:

☒ Component:

Any textfield, select, or hidden form element may be selected as the subject for e-mails.

E-mail from address

☐ Default: *rsafuto@gmail.com*

☐ Custom:

☒ Component:

Any email, select, or hidden form element may be selected as the sender's e-mail address.

E-mail from name

☐ Default: *Your Blog*

☐ Custom:

☒ Component:

Any textfield, select, or hidden form element may be selected as the sender's name for e-mails.

Save e-mail settings

Figure 38: Webform Email Settings

While the rest of the default settings are fine to begin with I recommend that you make some changes to the *E-Mail Header Details*. Specifically you can set the various details to display the components (Subject, Your Email Address and Your Name) directly from the form itself. You do that by selecting the *Component* radio button for each field and then selecting the appropriate form component from the list box. Once you have done that click the **Save e-mail settings** button at the bottom of the page.

The final step in configuring your webform involves the remaining form settings. Click the **Form settings** link (to the right of E-mails) to reveal the proper window. The first thing you should do is add a custom *Confirmation message*. Something like the following message is fine, "Thanks you for submitting your message to my blog. If

Building Your Blog With Drupal 7

necessary I will contact you within 48 hours.” The next section below form settings allows you to set a custom redirection location if you wish. The next section allows you to set a *Submission limit* for the form. Below that is the *Status of this form* section which allows you to make the form available (open or closed) or not. Since this will be the site-wide contact form you will leave it open. Finally, Submission Access allows you to fine tune the roles that have permission to submit the form. Make sure that *anonymous user* is checked. That setting will allow any visitor to your site to submit the form. When you’re done click the **Save configuration** button.

If you navigate to your form (there will now be a link in the menu tab) the form should look very similar to the one in the image below.

Building Your Blog With Drupal 7

Contact

[View](#) [Edit](#) [Webform](#) [Results](#)

Submitted by [your_blog_admin](#) on Sat, 06/25/2011 - 16:02

Use this form to contact us.

Your Name *

Your Email Address *

Please enter your email address so we can contact you if necessary.

Subject *

Please add a very brief description about why you are contacting us.

Message *

Please add the details of your message here.

Figure 39: Webform Contact Form

Take note of the guidelines below the *Your Email Address*, *Subject* and *Message* fields. Remember that those were populated via the *Description* field in the component configuration window. You'll also notice the "submitted by..." field at the very top of the form. You can remove that field via the Webform content type edit window. Select **Structure** from the admin menu then **Content Types** and finally select the **edit** link next to the Webform content type. Click the **Display settings** tab (see image below) then uncheck the box next to "Display author and date information," and click the Save content type button.

Building Your Blog With Drupal 7

[Home](#) » [Administration](#) » [Structure](#) » [Content types](#)

Name *

Webform

Machine name: webform [\[Edit\]](#)

The human-readable name of this content type. This text will be displayed as part of the list on the *Add new content* page. It is recommended that this name begin with a capital letter and contain only letters, numbers, and spaces. This name must be unique.

Description

Create a new form or questionnaire accessible to users. Submission results and statistics are recorded and accessible to privileged users.

Describe this content type. The text will be displayed on the *Add new content* page.

Submission form settings

Title

☐ Display author and date information.

Author username and publish date will be displayed.

Publishing options

Published , Promoted to front page

Display settings

Don't display post information

Comment settings

Open, Threading , 50 comments per page

Menu settings

Save content type

Delete content type

Figure 40: Webform Edit Window

When you return to your contact form page the author and date information will no longer be present on the form. Congratulations on adding your blog contact form!

Preventing Contact Form Spam

Before you can start using the contact form on a live website you need to add some spam protection. I'm recommending that you utilize the Mollom module which requires that you connect to the free Mollom web-based anti-spam service. If you sign up at <http://mollom.com> they will issue you one public and one private API key for your site.

Once you have signed up for Mollom you should then go to the Mollom settings on your site to enter your keys. Click **Configuration** in the admin menu and select **Mollom** which is in the *Content Authoring* section. You will see a warning with a link to the **Mollom settings page**. Click through to the settings page. Once there enter your public

Building Your Blog With Drupal 7

and private keys then click **Save configuration**. Then go back to the Mollom configuration window and click the **Add form** link.

In the next window you want to *select Webform: Contact form* from the list box then click **Next**. The next window has a few more settings to consider and they're shown in the image below.

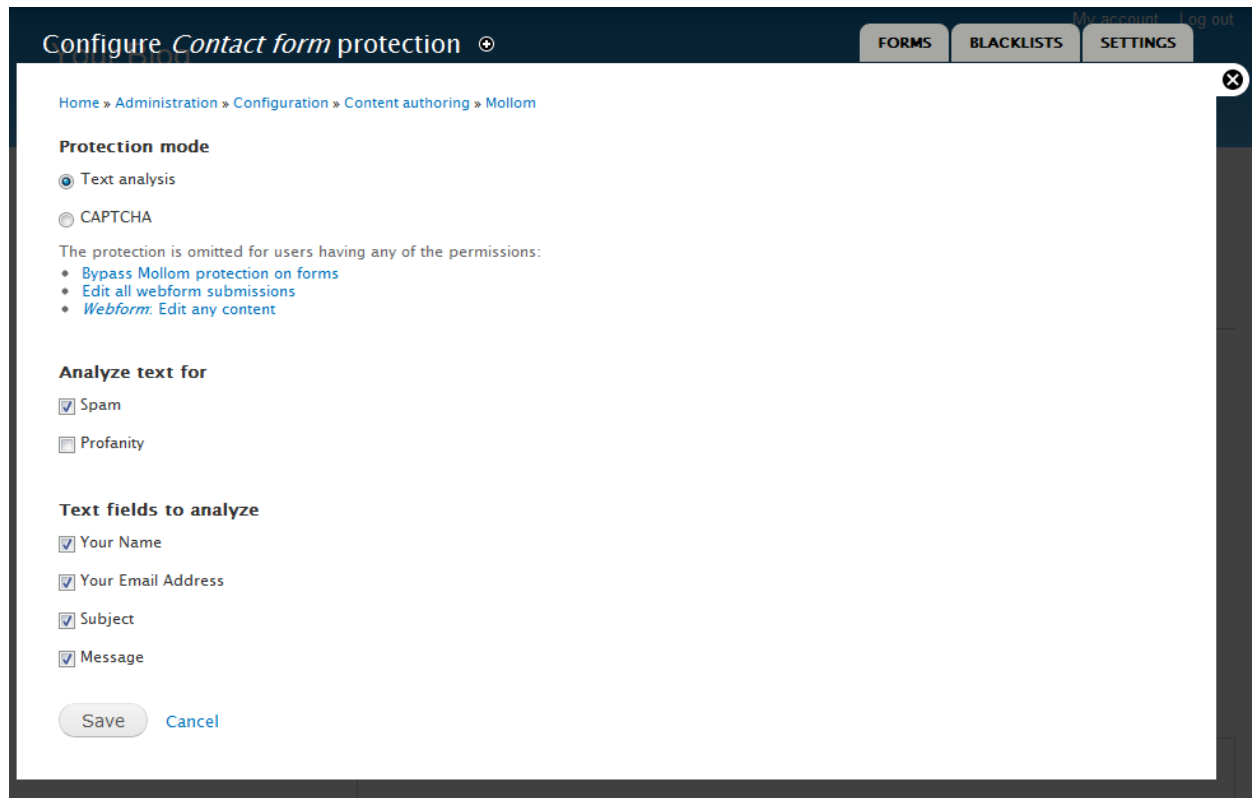
The image shows a screenshot of the 'Configure Contact form protection' window in a Drupal 7 administration interface. The window has a dark blue header with the title 'Configure Contact form protection' and a close button. Below the header, there are three tabs: 'FORMS', 'BLACKLISTS', and 'SETTINGS'. The 'SETTINGS' tab is active. The main content area is white and contains the following sections: 1. 'Protection mode' with two radio buttons: 'Text analysis' (selected) and 'CAPTCHA'. 2. A note stating 'The protection is omitted for users having any of the permissions:' followed by a list of permissions: 'Bypass Mollom protection on forms', 'Edit all webform submissions', and 'Webform: Edit any content'. 3. 'Analyze text for' with two checkboxes: 'Spam' (checked) and 'Profanity' (unchecked). 4. 'Text fields to analyze' with four checkboxes: 'Your Name' (checked), 'Your Email Address' (checked), 'Subject' (checked), and 'Message' (checked). At the bottom, there are 'Save' and 'Cancel' buttons.

Figure 41: Mollom Configuration

The Protection mode settings determine whether the text should be analyzed or a CAPTCHA should be presented. If *Text analysis* is chosen the user will only see a CAPTCHA if Mollom believes that their post may be spam. Otherwise users will always see a CAPTCHA. The *Analyze text for* section lets you check for spam and/or profanity. Finally, the *Text fields to analyze* section lets you choose which fields to analyze for potential spam or profanity. When you're finished with the settings click the **Save** button.

Building Your Blog With Drupal 7

Since you're already working with Mollom now is a good time to add protection for the comment form on your blog posts. You can do that by once again clicking the **Add form** link on the Mollom configuration page but this time select *Comment: Blog Post comment form*.

It's important to note that you have the option of turning off Mollom analysis for certain users on your site. You can fine tune the appropriate settings via the Mollom permissions which can be found in the site-wide permissions window. You get to that window by clicking on **People** in the admin menu and then selecting the **Permissions** tab on the upper right corner of the screen. If you scroll down to the Mollom section you'll see something that looks like the image below.

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR
Mollom			
Administer Mollom-protected forms and Mollom settings	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Bypass Mollom protection on forms	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
View Mollom statistics	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 42: Mollom Permissions

If you want authenticated (aka logged in) users to be able to submit forms without Mollom analysis or CAPTCHAs check the box in the *Authenticated User* column in the *Bypass Mollom protection on forms* row. Scroll to the bottom and click the **Save permissions** button in order to save the changes.

Checkpoint 1

If you have been building your blog step-by-step it's time to offer you congratulations because you have done a lot. In fact, I you have built a very functional website. Since you have come so far I think it's important to compare what you've done with the list of features that I promised at the beginning of this manual.

- A "front" (or home) page with a list of ten partial (or teaser) posts listed in descending order by published date with a pager to navigate to older posts. **Done!**

Building Your Blog With Drupal 7

- An "about" page that gives you the opportunity to tell people a little something about who you are and what the site is about. **Done!**
- A "contact" page that gives people the opportunity to provide you feedback via a simple form whose results are emailed to you. **Done!**
- A "blog post" content type that includes a *what you see is what you get* (WYSIWYG) editor, a field for freely tagging (or categorizing) your blog posts, an image upload tool that allows you to insert images into your posts, and the ability to easily embed YouTube videos. **Partially done!**
- Commenting functionality with moderation so that comments can be added to posts with your approval. **Not done!**
- Spam protection so that your contact form and comment fields aren't deluged by offers for pharmaceuticals, gold watches and World of Warcraft goodies. **Done!**
- A widget attached to each blog post that makes it easy to share the post on a variety of popular platforms. **Not done!**
- A side bar that includes a list of recent comment and blog post titles with links to the full posts. **Done!**
- Links (or "URLs") to blog posts that include the posted date and title in the link. **Done!**
- A tag "cloud" that resides in the sidebar that links to the topics that you have mentioned in your blog posts. **Done!**

As you can see you have accomplished most of what I have set out to help you to do with your site. There is still some work to do yet so let's get right to it.

Comment Moderation Preferences

Even the best spam filters still allow spam through from time to time. That's why you'll want to consider whether or not you would like to moderate comments. Thankfully Drupal makes it easy for you to turn comment moderation on and off.

Go the *Permissions* window by clicking on the **People** link in the admin menu. Then click the **Permissions** tab in the top right hand corner of the next window. Scroll down to the *Comment* section and you should see something that looks like the image below.

Building Your Blog With Drupal 7

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR
Comment			
Administer comments and comment settings	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
View comments	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Post comments	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Skip comment approval	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Edit own comments	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 43: Comment Permissions

The way the settings are configured above anonymous (aka not logged in) visitors may not post comments at all. You can change that by selecting the checkbox in the *Post comments* row of the *Anonymous User* column. Since the checkbox next to *Skip comment approval* is not checked for anonymous visitors their comments will be held for moderation regardless of what the spam filter thinks about their content. When a comment is held for moderation the visitor will see the message below.

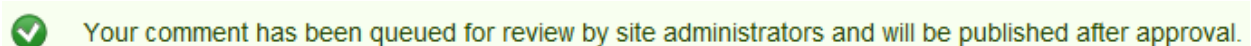


Figure 44: Comment Moderation Message

As the administrator of your blog you'll want to moderate the comments that are held by Drupal. You can see the list of comments held for moderation by clicking on the **Content** link in the admin menu. Then select the **Comments** tab in the upper right part of the screen. In the comment administration window you will see a pair of links (*Published comments* and *Unapproved comments*) in the upper right below the tabs as in the image below. There will be an indicator if unapproved comments exist.

Building Your Blog With Drupal 7

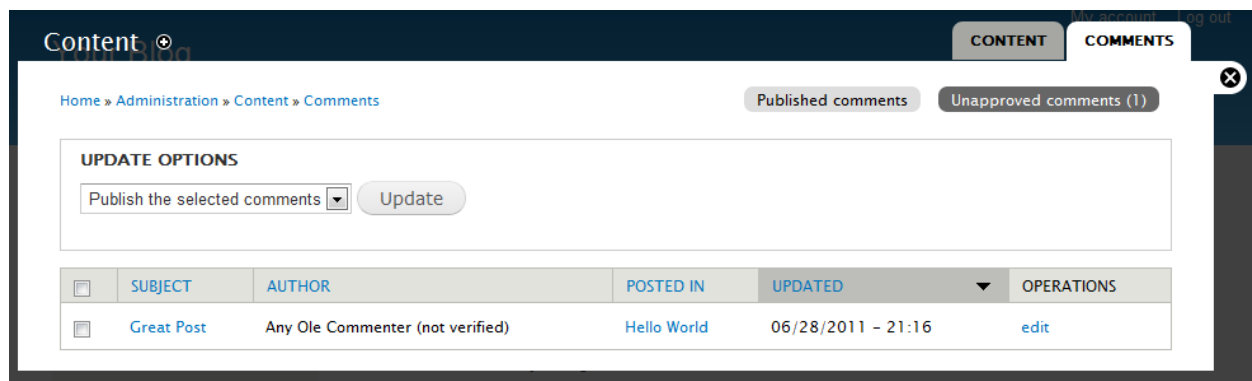


Figure 45: Comment Administration

You can check the content of the comment by clicking on the post title in the *Subject* column. If you want to approve the comment simply select the checkbox next to the comment, select *Publish the selected comments* in the *Update Options* list box and click the **Update** button. If you want to get rid of the comment select *Delete the selected comments* instead.

Easy Content Sharing

What good is great content if it can't be shared? You want to make it easy to share your posts so I recommend using a contributed module to do that. For now I'll show you how to make it happen using the [ShareThis](#) module which you should have installed earlier. If you haven't installed and enabled ShareThis you should do that right away.

Once you have installed ShareThis navigate to [admin/config/sharethis](#) where you can choose a variety of options to customize the sharing experience by selecting the look and feel of the sharing links, choosing sharing services and indicating which content types your sharing links will appear on. The *widget type* and *button style* options are seen in the image below.

Building Your Blog With Drupal 7

SHARETHIS CONFIGURATION SETTINGS:

Choose a widget type: *

- ☐ Multi-Post Widget
- ☒ Direct-Post Widget

Choose a button style: *

- ☒ Large Chicklets
- ☐ Small Chicklets
- ☐ Classic Buttons
- ☐ Vertical Counters
- ☐ Horizontal Counters

☐ Don't show the buttons in teaser view.

Classic
Your users will be redirected to Facebook, Twitter, etc when clicking on the corresponding buttons.
The widget is opened when users click on "Email" and "ShareThis".

Preview




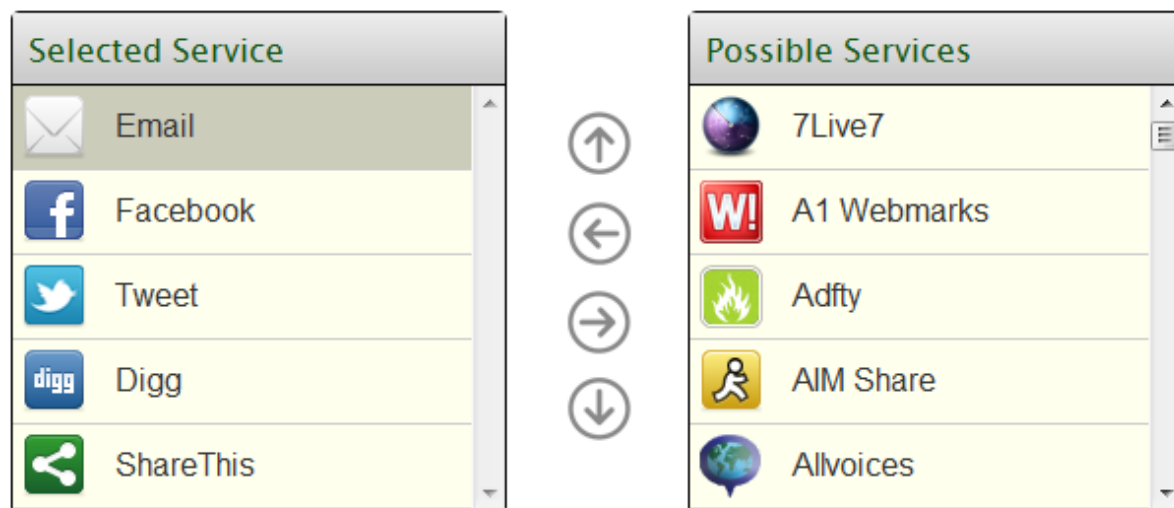
Figure 46: ShareThis Configuration

In addition to the options seen in the image above you can choose to show links from many sharing services and then select the types of nodes that you would like those links to appear on. These particular choices are seen in the image below.

Building Your Blog With Drupal 7

Choose Your Services. *

Add a service by selecting it on the right and clicking the *left arrow*. **Remove** it by clicking the *right arrow*. **Change the order** of services under "Selected Services" by using the *up* and *down* arrows.



Choose the node types you want to show the buttons on. *

- ☐ Article
- ☒ Blog Post
- ☐ Basic page
- ☐ Webform

Figure 47: ShareThis Configuration

Once you have saved your settings click the submit button. After you save go check your blog post to see how it looks. You should see the sharing links presented in a way that is consistent with your ShareThis settings.

Embedding YouTube Videos

Having the ability to easily embed YouTube videos is very important given the popularity of YouTube. I will show you how to make this happen by using two different methods. The first method utilizes the [Media module](#) and its child module [Media: YouTube](#). The second method, using the embedded iframe code, is much simpler but doesn't offer the advanced media library features of the media module.

Building Your Blog With Drupal 7

Media Module Method

The first thing that you should do is install both of those modules plus the required [Styles module](#). When you are enabling the modules at the very least you need to enable *Media*, *Media Internet Sources*, *Media: YouTube*, and all of the modules in the *Styles* group.

Once you have completed those steps you will be editing the profile for the WYSIWYG editor. Click on **Configuration** in the admin menu then select the **WYSIWYG profiles** link. Select the **Edit** link in the *Full HTML* profile row. In the *Buttons and Plugins* section you should select the checkbox for Media browser (it should be at the bottom) and then click the **Save** button.

What you have just done will add a media browser icon to your WYSIWYG editor. Before you can try that out you need to edit your *Full HTML* text format so that it will recognize the markup created by the media browser when it embeds content within your posts. Click on **Configuration** in the admin menu then click on **Text formats**. Select the **Configure** link in the *Operations* column of the *Full HTML* format. Select the checkbox next to *Convert Media tags to Markup* and click the **Save configuration** button.

Now you're ready to try out your new YouTube embedding capabilities. Use the **Blog Post** link in the shortcut bar to open the content editing window. You can see in my editing window below that there is a new icon that has the alternate text, "Add media."

Building Your Blog With Drupal 7

The screenshot shows the Drupal 7 WYSIWYG editor interface. At the top, there is a 'Title' field with a red asterisk, containing the text 'Embedded Video From YouTube'. Below it is a 'Tags' field with a red asterisk, containing the text 'video'. A message below the tags field says 'Please add descriptive tags to this post.' The main area is the 'Body' field, which has a toolbar with various formatting options (bold, italic, underline, text color, background color, bulleted list, numbered list, link, unlink, image, video, audio, embed, etc.). The body text area contains the text 'Check out this YouTube video!' and an 'Add media' button. At the bottom, there is a 'Path' field with the text 'p' and a link to 'Disable rich-text'.

Figure 48: WYSIWYG with Media Browser

If you click that icon you will see a nodal dialog window with tabs named *Upload*, *Web* and *Library*. If you click on the **Web** tab you will see something that looks like the image below.

The screenshot shows a dialog window titled 'Media Browser Web Option'. It has three tabs: 'Upload', 'Web' (selected), and 'Library'. The 'Web' tab contains a section titled 'URL or Embed code' with a text input field and a message: 'Input a url or embed code from one of the listed providers.' Below this is a section titled 'Supported Providers' which lists 'YouTube' twice, each with a small icon. At the bottom of the dialog are 'Submit' and 'Cancel' buttons.

Figure 49: Media Browser Web Option

Building Your Blog With Drupal 7

Copy the link to your desired YouTube video then paste it into the *URL or Embed Code* field. Click the **Submit** button to make it official. You should see a preview image appear at the point where your cursor was located before opening the media browser. When you save the post you should see the video embedded in the same spot where the preview image was located.

What you have also done by saving the video to the post is add the video to your media library. If you go back to the media browser and click the *Library* link you will see thumbnail images of the videos that you have added to your posts. You can also browse the videos you have previously embedded by selecting **Content** from the admin menu and then clicking the **Media** tab on the upper right portion of the screen.

You can change the size of the video embed and the size of the preview image by editing the *Video* file type. Select **Configuration** from the admin menu then click **File types**. In the *Video* row of the file types table click on the **manage file display** link. If you select **Original** in the upper right hand corner you will see the following window which allows you to adjust both the size of the video embed and the size of the preview image.

Clicking on the **YouTube video** tab at the bottom of the page offers you standard *width*, and *height* options, as well as an option to autoplay videos. If you select the **YouTube Preview Image** tab you will be presented with the option to select an *Image style* which will correspond to the image styles that are explained in the next section.

Building Your Blog With Drupal 7

The screenshot shows the 'Manage file display' configuration page in Drupal 7. The breadcrumb trail is: Home > Administration > Configuration > Media > File types > Manage file display. The page has three tabs: 'MANAGE FIELDS', 'MANAGE DISPLAY', and 'MANAGE FILE DISPLAY' (which is active). Below the tabs are buttons for 'Default', 'Link', 'Preview', 'Small', 'Large', and 'Original'. The 'Enabled displays' section lists various display types with checkboxes: Generic file, Table of files, URL to file, Large filetype icon, File style: square_thumbnail, File style: thumbnail, File style: large, File style: medium, File style: original, Image, YouTube Video (checked), and YouTube Preview Image (checked). The 'Display precedence order' section shows a list of displays with a 'Show row weights' link. The 'Display settings' section shows settings for 'YouTube Video' and 'YouTube Preview Image', both of which are enabled. The 'Width' is set to 640 and the 'Height' is set to 480. There is an 'Autoplay' checkbox which is unchecked. A 'Save configuration' button is at the bottom.

File types

MANAGE FIELDS MANAGE DISPLAY MANAGE FILE DISPLAY

Home > Administration > Configuration > Media > File types > Manage file display

Default Link Preview Small Large Original

Enabled displays

- ☐ Generic file
- ☐ Table of files
- ☐ URL to file
- ☐ Large filetype icon
- ☐ File style: square_thumbnail
- ☐ File style: thumbnail
- ☐ File style: large
- ☐ File style: medium
- ☐ File style: original
- ☐ Image
- ☒ YouTube Video
- ☒ YouTube Preview Image

Display precedence order [Show row weights](#)

	YouTube Video
	YouTube Preview Image

Display settings

YouTube Video Enabled	Width <input type="text" value="640"/>
YouTube Preview Image Enabled	Height <input type="text" value="480"/>
	<input type="checkbox"/> Autoplay

Save configuration

Figure 50: YouTube Video and Preview Image Settings

It's important to note two shortcomings of this approach at the time of this writing.

1. You can change the size of the video embed via Configuration => File Styles but can't adjust the dimensions on a per video basis.
2. When viewing the video thumbnails in the media gallery you cannot create a proper title.

Building Your Blog With Drupal 7

I have scanned the issue queue on Drupal.org and it appears that the first issue has been patched so it will probably make it into the production version of the Media: YouTube module soon. It's also worth noting that there is a similar module that supports the Vimeo video service called [Media: Vimeo](#).

Embed Code Method

You can also embed YouTube videos by using the iframe embed code provided YouTube. Although this method is easy I have found that it causes some interesting usability issues when editing posts. Specifically, if you happen to attempt to edit a post with an embedded YouTube video that uses the iframe code the video embed will continue to be overlaid on the screen as in the image below.

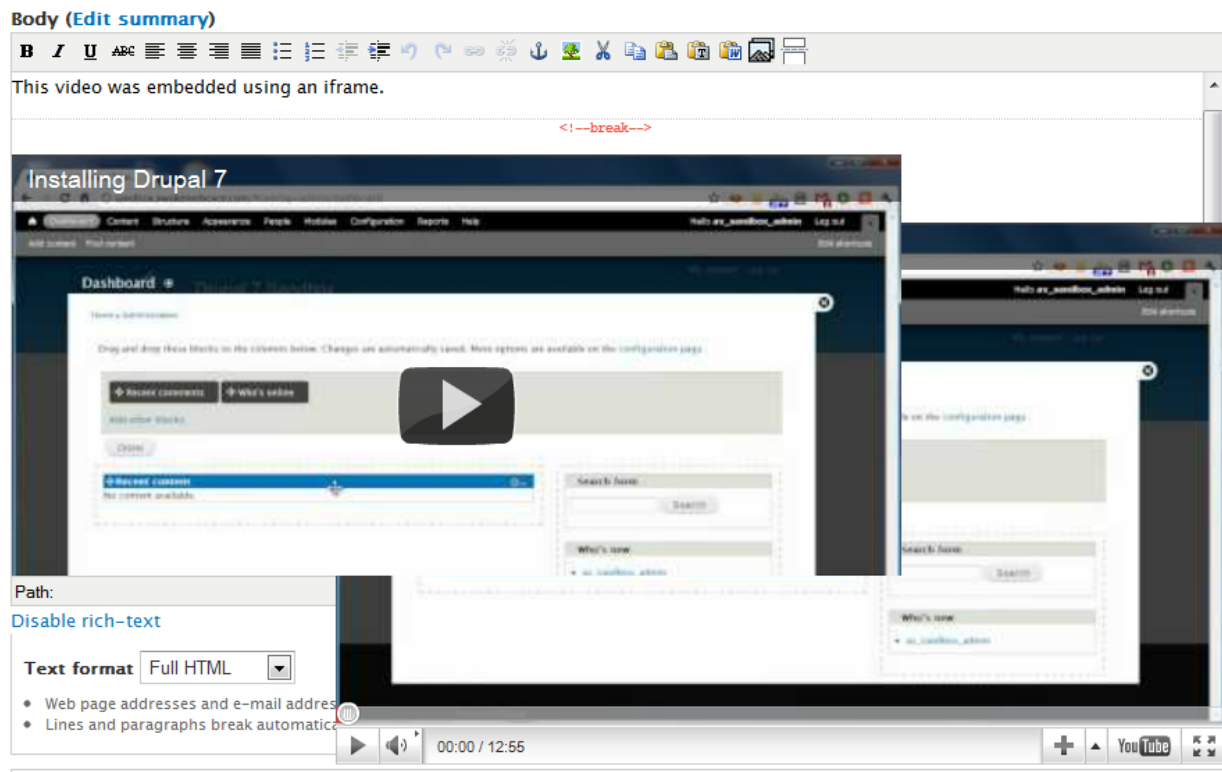


Figure 51: YouTube Iframe Overlay

This particular issue is not a problem if you launch the edit window from a page that does not have the video displayed. You also have the option of disabling overlays which you can do by disabling the core *Overlay* module.

Building Your Blog With Drupal 7

If you still want to use this method all you need to do is copy the embed code that YouTube offers and paste it into the plain text editing view of your blog post. To access the plain text editing view of your post editor click on the **Disable rich-text** link that is below the post editor on the left side. That link toggles the rich text view on and off. Paste the embed code at the desired position and then click the Enable rich-text link to continue editing the post as you normally would.

Inserting Images Into Posts

In addition to using videos you'll also want to spice up your blog posts with images. Thankfully, you already did a good amount of the work necessary to easily integrate images when you added the media module. Before you start adding images you need to know something about the Drupal file system and image styles.

The settings for the Drupal file system can be found by selecting **Configuration** in the admin menu and then selecting the **File System** link in the *Media* group. The most important thing to focus on right now is the *Public file system path* field. That is set to *sites/default/files* by default and unless you change that setting all of our original uploaded pictures will reside in that directory.

There is also a link in the *Media* group for **Image styles**. If you click on that you will see four styles as in the image below. Each one of those styles controls the options for how your original image upload will be displayed in your post.

Building Your Blog With Drupal 7

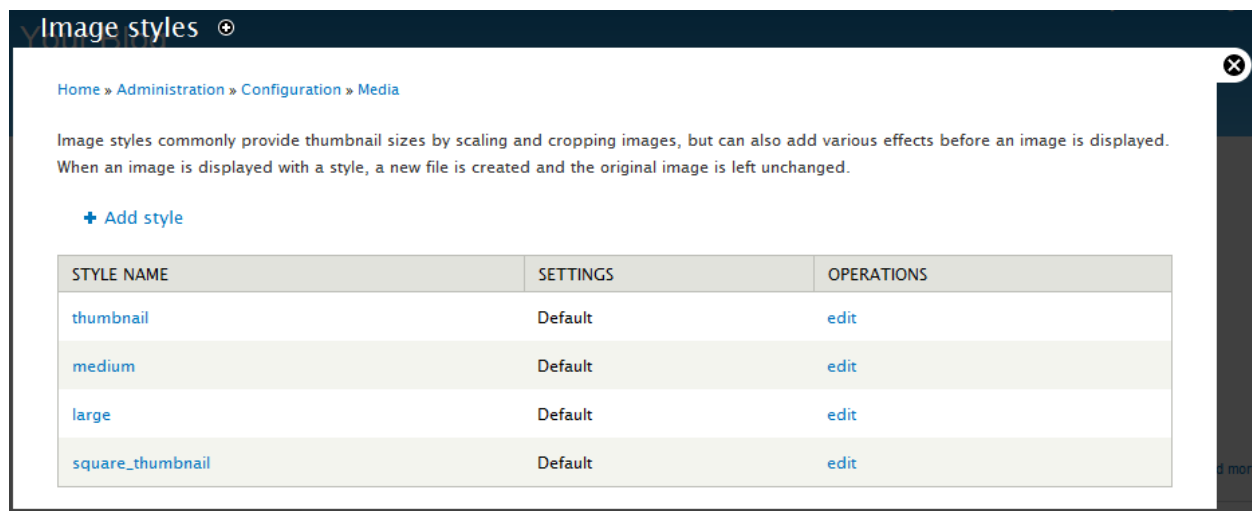


Figure 52: Image Styles

By selecting one of the styles when inserting an image you can standardize images into the preset sizes and create a cleaner, more professional look on your site. You can use the default styles as they are, edit the default styles or create new ones. The table below provides a brief summary of the standard styles. It's important to clarify the meaning of the two effects mentioned in case you're new to image manipulation in Drupal. The *scale* effect will resize the image while retaining the original dimensions. For example if you scale an 800 by 600 pixel image to 400 by 400 the resulting image will be 400 by 300 pixels. The *scale and crop* effect will scale the image dimensions and then crop the image to an exact length and width that you specify.


Style Name	Effect(s)	Size
Thumbnail	Scale	100 by 100 pixels
Medium	Scale	220 by 220 pixels
Large	Scale	480 by 480 pixels
Square Thumbnail	Scale, Crop	180 by 180 pixels

Table 2: Default Image Styles

Those default options may not meet your needs so I'll step through the process of making a change to one of those default styles. For the example I'll have you edit the *Large* style to expand the size to 600 pixels. Click on the **edit** link the large style row.


Building Your Blog With Drupal 7

Then select the **Override defaults** button. In the next window click the **edit** link in the *Scale 480x480* row. Change both the *width* and *height* fields to 600 then click **Update effect**. The window in the image below should appear, confirming that the update was applied successfully.

 The image effect was successfully applied.

Preview


original (view actual size)



800px

600px

large (view actual size)



600px

450px

Image style name

large

This image style is being provided by *image* module and may not be renamed.

[Show row weights](#)



EFFECT	OPERATIONS
 Scale 600x600	edit delete
 <input type="text" value="Select a new effect"/> <input type="button" value="Add"/>	

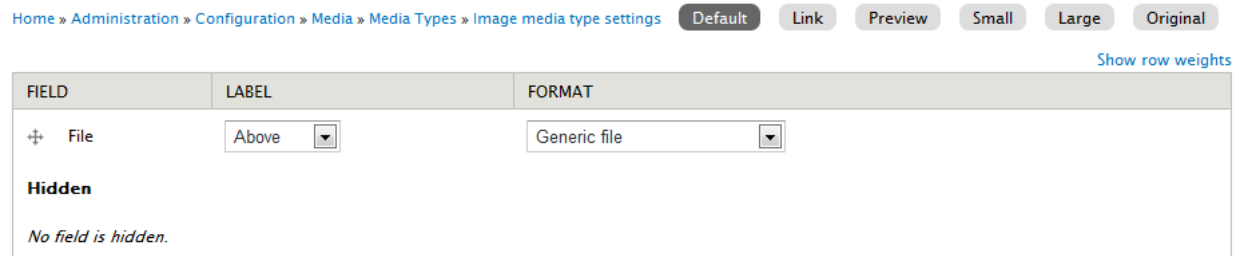
Figure 53: Image Style Editing




If you go back to the image style configuration window you will notice that the *settings* column of the *large* style row will now be populated with *Overridden* and there will be a *revert* link to the right of the *edit* link. You can return the style to the original default setting by clicking the *revert* link.

Since you're using the Media module to manage images there is an additional level of interaction between the Media module and image styles that you need to understand. By default the Media module will associate your image styles with its Image media type. You can find those settings in the *Media* group by clicking on the **Media Types** link. In the next window click on **manage display** in the *Image* row. You will then have access

Building Your Blog With Drupal 7

to the settings that directly translate to the ones you'll see when inserting image. Notice that there are links (in the upper right) to Default, Link, etc. in the upper right part of the image below.



FIELD	LABEL	FORMAT
 File	Above 	Generic file 

Hidden

No field is hidden.

Figure 54: Image Media Type Settings

If you click on each of those links you'll see the association between the particular display type and the associated image style. For example, if you click on the **Large** link you will see that the image file is associated with the format of the *File style: Large* which means that the image will be scaled to 600 by 600 when you choose the *Large* option when inserting media. If you click on the *Small* link you will notice that there is no association. You can create an association by selecting *File style: thumbnail* in the format list box and clicking the **Save** button.

Now that you know something about image styles it's time to actually insert some images into posts. Click on the Blog Post link in the shortcut menu to begin creating a new post. Fill out your title and tags then focus on the body field. Click the add media icon that I pointed out in figure 48 above. That will open up a dialog window with a **Choose File** button that you should click. Browse your hard drive for a nice picture and click the **Submit** button.

Upon uploading the photo you will be presented with a couple of options before inserting the image into the body of your post. The image below shows that you can choose from one of the media styles mentioned above. You can also choose the description which will be represented in the alt text of the image.

Building Your Blog With Drupal 7

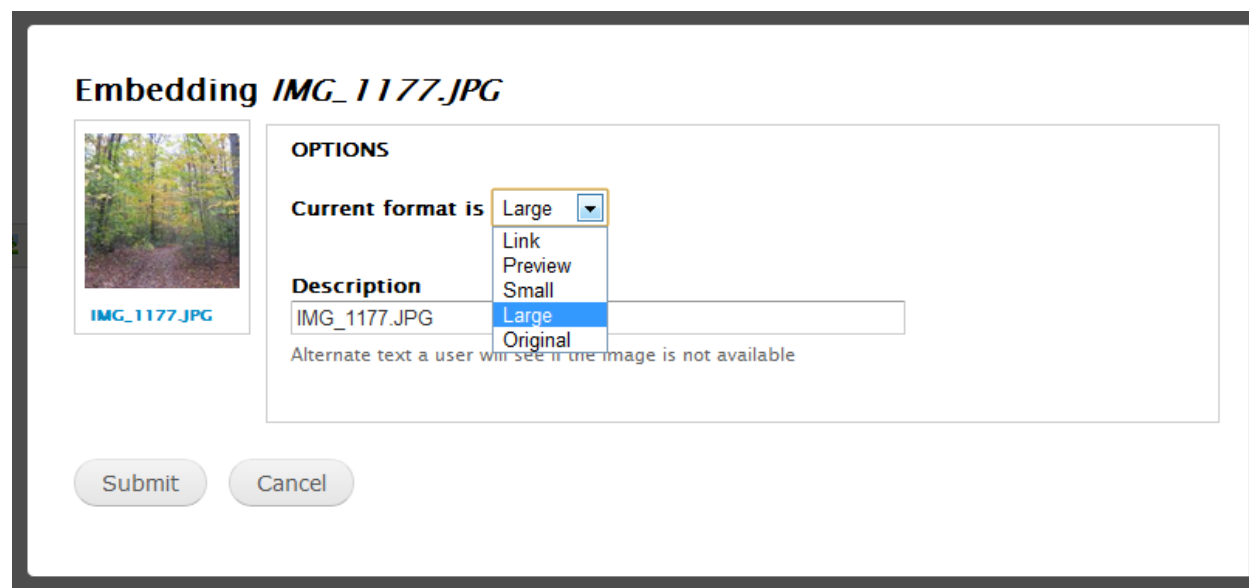


Figure 55: Image Embed Options

Once your choices have been made and you click the **Submit** button the image will be inserted into your post and you can finish editing the post and save it as you normally would. Keep in mind that this file is now available to insert in other posts by selecting the *Library* option in the add media dialog window. It can also be found in the Media gallery mentioned in the Embedding YouTube Videos section.

Checkpoint 2

The three features listed below were not complete when we reviewed our first checkpoint. Well they're complete now! Congratulate yourself on a job well done.

- A "blog post" content type that includes a *what you see is what you get* (WYSIWYG) editor, a field for freely tagging (or categorizing) your blog posts, an image upload tool that allows you to insert images into your posts, and the ability to easily embed YouTube videos. **Done!**
- Commenting functionality with moderation so that comments can be added to posts with your approval. **Done!**
- A widget attached to each blog post that makes it easy to share the post on a variety of popular platforms. **Done!**

Building Your Blog With Drupal 7

Managing Themes

The look and feel of a website is without a doubt one of the most important elements that people consider when building a site. Drupal has a very robust theme system that makes it easy to change the basic look and feel of your site. In this section I will step through the basics theme elements of your Drupal site.

The default theme for Drupal 7 is called Bartik. There are three other themes (Seven, Garland and Start) included with Drupal 7. All available themes can be found by clicking on **Appearance** in the admin menu.

Building Your Blog With Drupal 7

Set and configure the default theme for your website. Alternative [themes](#) are available.

[+ Install new theme](#)

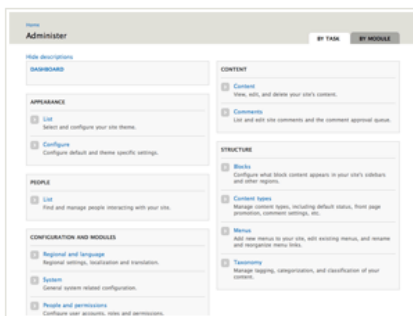
ENABLED THEMES



Bartik 7.4 (default theme)

A flexible, recolorable theme with many regions.

[Settings](#)



Seven 7.4

A simple one-column, tableless, fluid width administration theme.

[Settings](#) | [Disable](#) | [Set default](#)

DISABLED THEMES

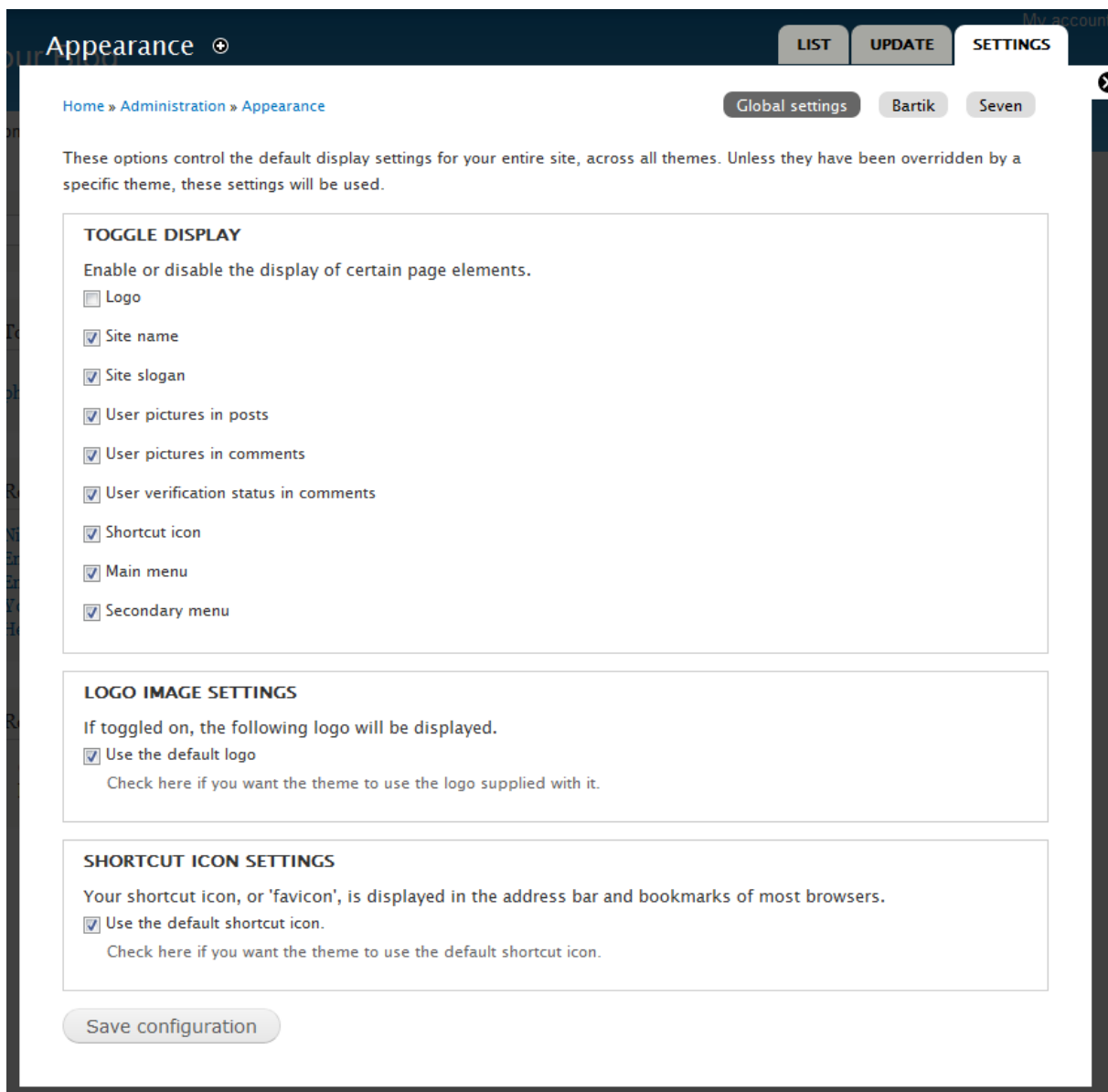


Figure 56: Theme Administration Window

You will notice that themes on your site are separated into two groups: *enabled* and *disabled*. If a theme is enabled you can quickly switch to change the site to that theme and also adjust the settings. To make an enabled theme the current theme of your site just click the **Set default** link.

There are two types of settings that control the look of the themes on your site: global and theme specific settings. Global settings for your themes can be accessed by clicking the **Settings** tab that is in the upper right part of the administration window.

Building Your Blog With Drupal 7



The screenshot shows the 'Appearance' administration page in Drupal 7. At the top, there's a dark blue header with 'Appearance' and a plus icon. To the right are buttons for 'LIST', 'UPDATE', and 'SETTINGS'. Below the header, a breadcrumb trail reads 'Home » Administration » Appearance'. On the right, there are tabs for 'Global settings', 'Bartik', and 'Seven'. The main content area explains that these options control default display settings. It is divided into three sections: 'TOGGLE DISPLAY' with checkboxes for Logo, Site name, Site slogan, User pictures in posts, User pictures in comments, User verification status in comments, Shortcut icon, Main menu, and Secondary menu; 'LOGO IMAGE SETTINGS' with a checkbox to 'Use the default logo'; and 'SHORTCUT ICON SETTINGS' with a checkbox to 'Use the default shortcut icon'. A 'Save configuration' button is at the bottom.

Figure 57: Global Theme Settings

When viewing the global settings you'll notice a variety of checkboxes as well as links to the theme specific settings for other enabled themes.

Editing Enabled Themes

A variety of specific settings can be adjusted for each of your enabled themes. To adjust the settings for an enabled theme click the **Settings** link next to the appropriate theme in the theme administration window. It's important to note that each theme is like a

Building Your Blog With Drupal 7

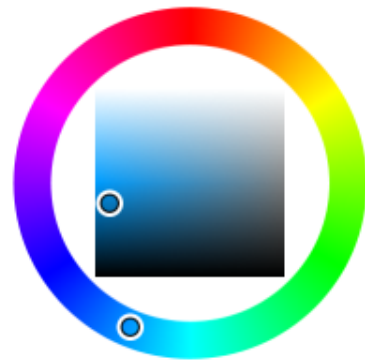
module in that it can have its own unique types of settings. While some themes will have simple settings others may offer you many options. The important take away is that you should spend some time reviewing the settings for any theme that you are considering using.

The core Bartik theme has settings that allow you to adjust the colors of the site with just a few clicks. You can either choose a standard color set or customize your own set of colors using the color wheel shown in the image below. You'll also notice below the color settings that you can see a preview of what the site will look like with the new colors.

Building Your Blog With Drupal 7

COLOR SCHEME

Color set	Blue Lagoon (default) ▼	
Header top	#0779bf	
Header bottom	#48a9e4	🔒
Main background	#ffffff	🔒
Sidebar background	#f6f6f2	🔒
Sidebar borders	#f9f9f9	🔒
Footer background	#292929	🔒
Title and slogan	#fffeff	🔒
Text color	#3b3b3b	🔒
Link color	#0071b3	🔒



Preview



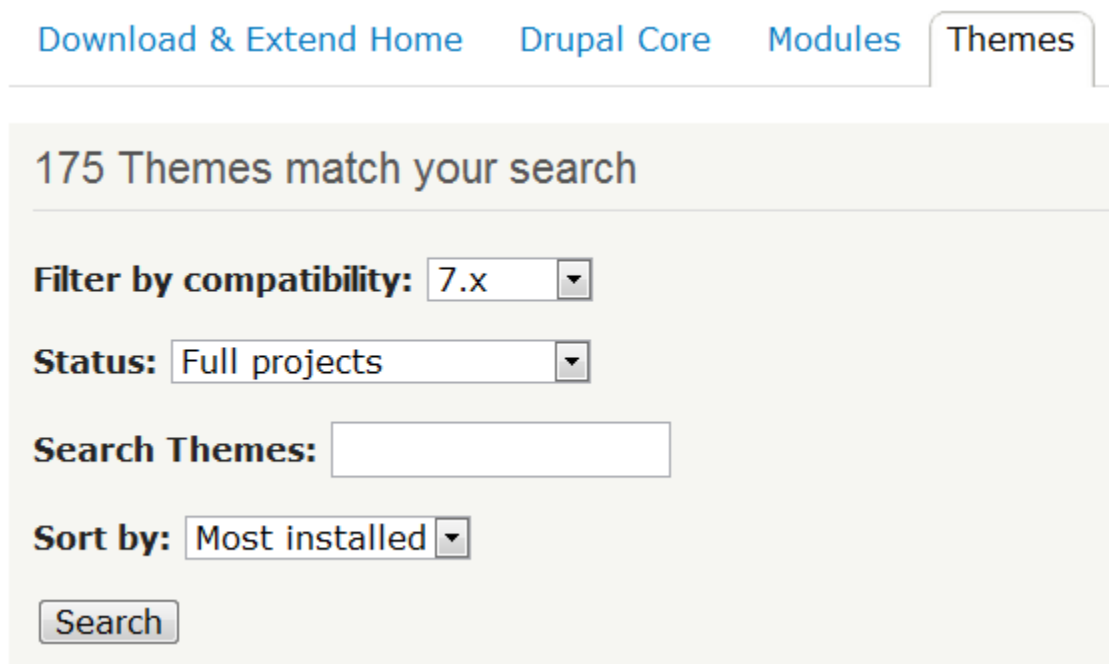
Figure 58: Bartik Color Wheel

Building Your Blog With Drupal 7

Below the theme preview you will find settings just like the ones on the global settings page. Although they are the same settings the changes to the settings on the theme specific settings window will only apply to the theme being edited.

Adding New Themes

At some point you'll want to experiment with more than just the core Drupal themes. In order to act on that desire you will first need to find Drupal 7 compatible themes. The best place to start your search is at the Drupal.org theme repository at <http://drupal.org/project/themes>. There are a few options that you can choose to filter the results as seen in the image below.



Download & Extend Home Drupal Core Modules Themes

175 Themes match your search

Filter by compatibility: 7.x ▼

Status: Full projects ▼

Search Themes:

Sort by: Most installed ▼

Figure 59: Drupal.org Theme Filters

Once you have found a theme that you would like to try you install the theme on your Drupal site via a process that is very similar to the module install process. Copy the link to the install file (which should end in either .tar.gz or .zip), navigate to the theme administration window and click on the **Install new theme** link. Once there paste the link to your file into the *Install from a URL* field and click the **Install** button. Upon

Building Your Blog With Drupal 7

successful installation of the theme you should see a message similar to the one in the image below.

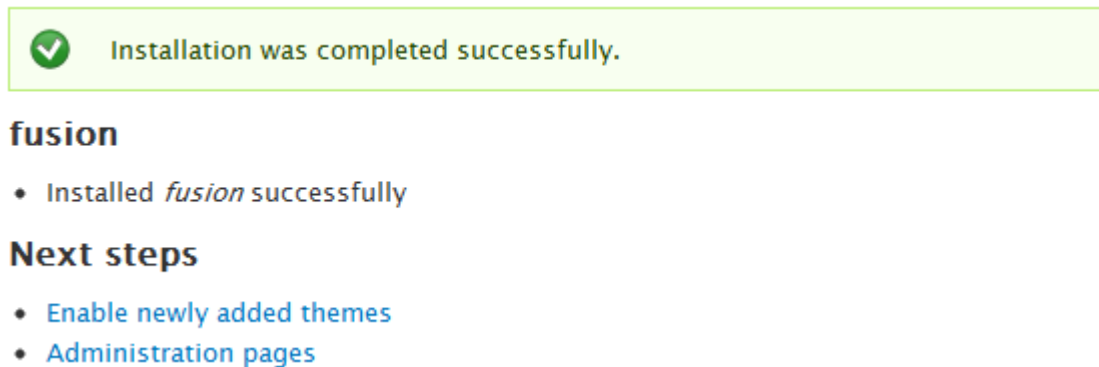


Figure 60: Theme Install Successful

You're now ready to begin exploring your newly added theme.

Administering Your Drupal Site

Backing Up Your Site

Frequent backups of your site are a good idea. Backups will allow you to recover the site to a certain state in the unlikely event that something goes wrong. There are two major elements of a Drupal site backup: the files, and the database.

Backing up your files is as easy as copying them from your server and downloading them to your local machine. If you have a particularly large site that has many uploaded files this could take a while depending on the speed of your internet connection.

Copying your site files every so often is an important task though so consider it time well spent.

There are different ways to make a copy of your database. More experienced users can use [phpMyAdmin](#) to take an export of the tables from the Drupal site. If you know and have used phpMyAdmin before you probably don't need my guidance on exporting the database. There is a good post on Drupal.org called [Exporting a database with PHPMyAdmin](#) that you can use as a step-by-step guide as well. You also have the

Building Your Blog With Drupal 7

option of using a module. If you would like to do that then I recommend the [Backup and Migrate module](#).

Backup and migrate settings can be found in the *System* group of the *Configuration* window. The module offers options for backing up the current database and restoring earlier versions of the database. You can back up the database to your web server or your local machine. If you want to back up to your server you must specify a private file system path in the *file system settings* window. For the moment I'm going to walk you through the process for downloading a copy of your database.

When you go to the backup and migrate settings you will have the option to take an immediate backup using the default settings as shown in the image below.

The screenshot shows the 'Backup and Migrate' configuration page. At the top, there are tabs for 'BACKUP', 'RESTORE', 'DESTINATIONS', 'PROFILES', and 'SCHEDULES'. Below the tabs, there are two buttons: 'Quick Backup' and 'Advanced Backup'. A yellow warning box states: 'You must specify a private file system path in the file system settings to backup to the server.' Below this, a text line says: 'Use this form to run simple manual backups of your database. Visit the help page for more help using this module'. The 'QUICK BACKUP' section contains a form with the following elements: 'Backup from' (Default Database), 'to' (Download), 'using' (Default Settings), and a 'Backup now' button. At the bottom, a text line says: 'For more backup options, try the advanced backup page.'

Figure 61: Backup and Migrate Quick Settings

If you click the Backup now button with the settings in the image above a dialog window will pop up that prompts you to save a .mysql file to your hard drive. You can customize the options, such as the file name and compression by clicking on the **Advanced Backup** link. If your database is large you may want to apply compression in order to save space and prevent potential problems with upload limits when restoring the backup.

In the event that you need to restore the database to a previous version you can do so by clicking on the **Restore** tab. The process of restoring the database involves

Building Your Blog With Drupal 7

uploading one of the saved database exports from your hard drive. You can also set a schedule for automatic database exports by clicking on the **Schedules** tab. The scheduling of database backups relies on cron which I will cover in the next section.

Running Cron

If you are familiar with Drupal from an earlier version you are probably familiar with cron. Cron triggers many important periodic tasks such as checking for module updates, updating the search index and running periodic tasks for other modules. You can run cron in one of three ways.

1. Click on the direct cron link
2. Configure the site to trigger cron via settings
3. From an external program or script

The direct cron link can be found at the location of your site *status report* which can be accessed via the **Reports** link in the admin menu. A section called *Cron maintenance tasks* can be found near the top of the status report. That report will tell you when cron was last run. That section includes a link to run cron manually and also includes a tokenized link that you can use to run cron from outside scripts or programs. The key in the link is new to Drupal 7 and it prevents an outside program from running cron without your permission.

Drupal	7.2
Access to update.php	Protected
CTools CSS Cache	Exists
Configuration file	Protected
Cron maintenance tasks	Last run 2 hours 5 min ago
You can run cron manually .	
To run cron from outside the site, go to http://localhost/yourblog/cron.php?cron_key=7psvVomZ3oEjEGjKhScAs	

Figure 62: Cron Status Report

Building Your Blog With Drupal 7

You can configure cron to run periodically via the **Cron** link in the *System* group of the *Configuration* window. As you can see in the image below the settings are very basic. You can either choose to run Cron by clicking a button or you can choose the interval for cron runs. It's important to note that the built in cron is triggered by a user accessing a page on your site. So if your site receives no traffic at the time when cron is supposed to run then cron will run later when someone finally accesses a page on the site. If you must run cron at a specific time then you should arrange to have it run via an external site or script.

The Status Report

In the previous section I mentioned the status report. The status report offers a quick overview of the situation with your Drupal website. Each line of the report is color coded to indicate how things are going with the site. Green shading means a particular item is okay. Yellow shading means you have something to look into. Red shading means that something needs to be fixed right away.

One very important bit of information on the status report is the *Drupal core update status*. That indicator lets you know if there are available updates to Drupal core. If a check hasn't occurred in a while then the row turns yellow as in the image below.

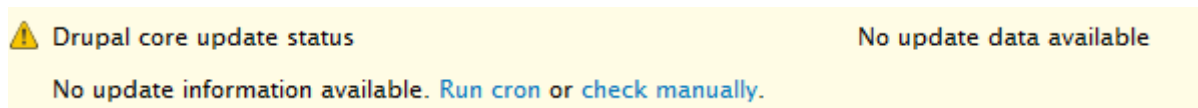


Figure 63: Core Update Status

A little further down in the status report is the *Module and theme update status* indicator. That bit of information lets you know if there are updates available.

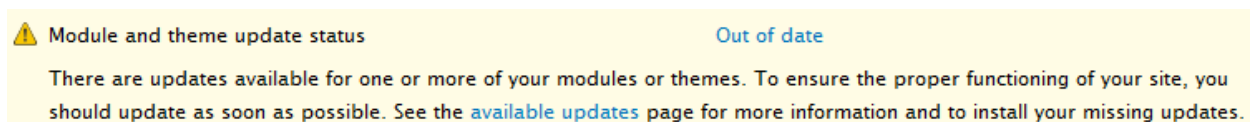


Figure 64: Module and Theme Update Status

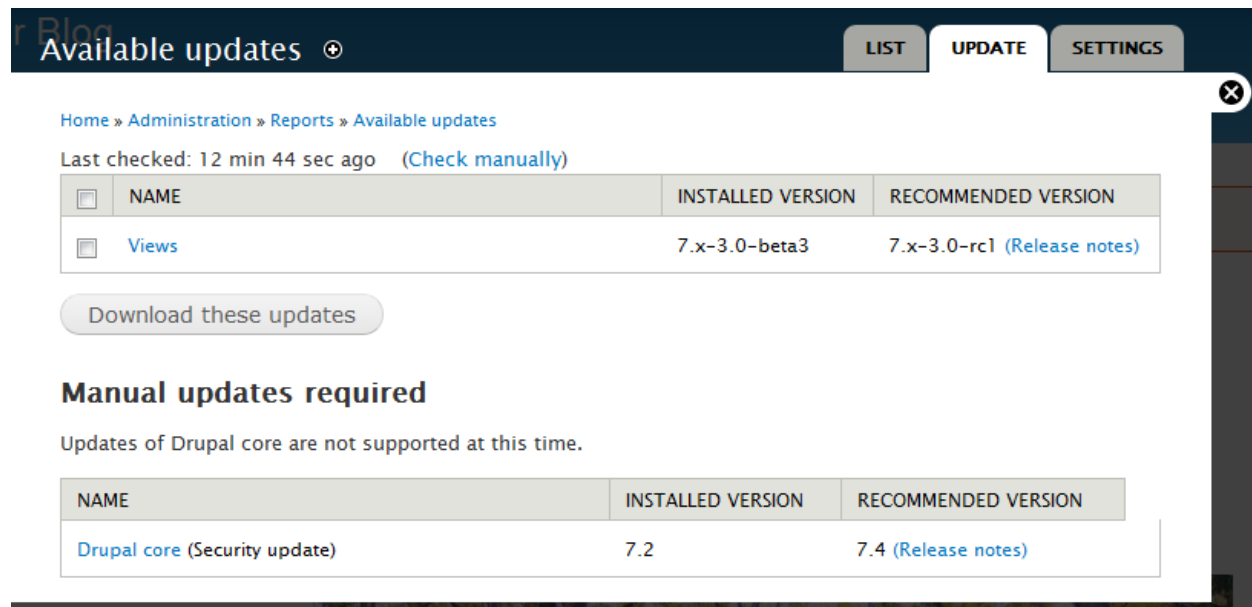
Building Your Blog With Drupal 7

In both cases if updates are required then links are included to click through and see which updates are available. You can also see the status of core, modules and themes by clicking on the **Available updates** link in the *Reports* window.

Chances are that you will find it helpful to be notified via email when an update is either available or required. If you click the **Settings** tab in the *Available updates* window you will find update notification options. In the next section I'll give you some information on how to make the actual updates.

Module, Theme And Core Upgrades

Modules and themes are easy to update in Drupal 7. When viewing the *Available updates* window click on the **Update** tab to open the window that will allow you to install the updates. As in the image below you will see which items have updates available and will find links to install, download or view release notes.



Available updates ⓘ

Home » Administration » Reports » Available updates

Last checked: 12 min 44 sec ago (Check manually)

<input type="checkbox"/>	NAME	INSTALLED VERSION	RECOMMENDED VERSION
<input type="checkbox"/>	Views	7.x-3.0-beta3	7.x-3.0-rc1 (Release notes)

Download these updates

Manual updates required

Updates of Drupal core are not supported at this time.

NAME	INSTALLED VERSION	RECOMMENDED VERSION
Drupal core (Security update)	7.2	7.4 (Release notes)

Figure 65: Automated and Manual Updates

Contributed modules and themes can be installed automatically by selecting the box next to the name and clicking **Download these updates**. When the updates are complete you will be prompted to continue with the site in *maintenance mode*. Maintenance mode

Building Your Blog With Drupal 7

takes your site offline temporarily why new code is being installed. You will then be prompted to make updates to run database updates. Continue until Drupal indicates that database updates are complete. If necessary, remove the site from maintenance mode by clicking the **Go online** link in the maintenance mode system message.

It's important to note that you can put the site into maintenance mode any time by selecting the **Maintenance mode** link in the *Development* group of the *Configuration* window. Once there you can select a check box to toggle the maintenance status. You can also customize the message that is displayed to visitors when they come to your site while it is in maintenance mode.

Drupal core updates must still be done manually. Take the following steps to perform a manual update.

1. Download and unzip the new Drupal release.
2. Put your site into maintenance mode.
3. Backup your *sites* directory and Drupal database.⁴
4. Delete all the files and folders except the *sites* directory.
5. Upload the new Drupal release files to the root of your website.
6. Navigate to /update.php and follow the prompts.
7. Review the feedback to ensure the updates completed.
8. Return to the administration pages and put the site back online.

The System Log

It's important that you are aware of the Drupal system log and how you might want to use it. The system log keeps a record of system messages, some of which may be very useful to you while you run your site. You can access the system log by clicking on the **Reports** link in the admin menu and then clicking on the **Recent Log Messages** link. Once there you will see a window that looks like the image below.

⁴ If you have edited your .htaccess file and/or robots.txt then back these files up too.

Building Your Blog With Drupal 7

▼ FILTER LOG MESSAGES

Type

access denied
actions
content
cron
field
file
mail
mollom

Severity

emergency
alert
critical
error
warning
notice
info
debug

Filter

▼ CLEAR LOG MESSAGES

This will permanently remove the log messages from the database.

Clear log messages

	TYPE	DATE ▼	MESSAGE	USER	OPERATIONS
⚠	php	07/08/2011 - 19:35	Warning: Invalid argument supplied for foreach() in...	your_blog_admin	
	php	07/08/2011 - 19:35	Notice: Undefined variable: tags_tmp in tagadelic...	your_blog_admin	

Figure 66: System Log Window

Notice that you can filter the messages to focus on only ones that you are interested in. For example, if you click on the *cron* option in the *Type* box then click the **Filter** button you will only see messages related to your cron runs. You can filter even further down by selecting a *Severity* level in the box to the right of *Type*. You can return to seeing everything by clicking the **Reset** button that appears once results have been filtered.

So what should you look for? Good question! Is something isn't working the way you think it should work or if you are seeing warning or error system messages during the process of managing your site then it's a good idea to check the log to try to find out more details about the problems. What then? If the issue relates to a problem with a specific module then you probably want to check the issue queue for the module on Drupal.org to see if other people have experienced the same issue. If you can't find the issue then it's usually a good idea to post a new issue, indicating what happened and including the message from your log.

If the log message you're investigating isn't specific to a particular module and you don't understand what the problem is then I recommend using the official [Drupal forum](#) to ask


Building Your Blog With Drupal 7

questions and research the issue. A Google search on the issue or specific message certainly can't hurt either.

Adding Additional Authors

At some point you may want to add someone other than yourself as an author on your blog. Adding new authors is simple but it does require that you understand a little bit about roles and permissions in Drupal. First I'll show you how to set up a new role just for your added authors and then I'll show you how to add the new user account.

Select the **People** in the admin menu then click the **Permissions** tab. On the *Permissions* window you will see a *Roles* link in the upper right portion of the window. Click on the **Roles** link. Add a new role by entering the role name (I'm calling mine "author") in the blank text field at the bottom of the *Name* column and click the **Add role** button. You should receive a confirmation like the one in the image below.

 The role has been added.





Roles allow you to fine tune the security and administration of Drupal. A role defines a group of users that have certain privileges as defined on the [permissions page](#). Examples of roles include: anonymous user, authenticated user, moderator, administrator and so on. In this area you will define the names and order of the roles on your site. It is recommended to order your roles from least permissive (anonymous user) to most permissive (administrator). To delete a role choose "edit role".

By default, Drupal comes with two user roles:

- Anonymous user: this role is used for users that don't have a user account or that are not authenticated.
- Authenticated user: this role is automatically granted to all logged in users.

NAME

OPERATIONS

 anonymous user <i>(locked)</i>	edit permissions
 authenticated user <i>(locked)</i>	edit permissions
 administrator	edit role edit permissions
 author	edit role edit permissions

Add role

Save order

Figure 67: Role Added Confirmation

Building Your Blog With Drupal 7

Take note of the fact that you can drag your new role name in order to change the left-to-right order when you are viewing the roles in the global permissions window. The next step you need to take is to give the role the appropriate permissions. To do that click on the **permissions** link in the upper right portion of the window. That will take you back to the global permissions view that includes all of the other roles. I use this window to change permissions for a role because I find it easier to review a particular role in relation to other roles.

When you look at the *author* column you will notice that a number of the permissions have been checked and grayed out. Those permissions have been inherited from the *Anonymous* (or not logged in) *User* and the *Authenticated User* roles. I could have made things easier by having you edit the *Authenticated User* role to add post authoring capabilities, but creating a separate role for authors is a better practice because you may want to use that authenticated role to use to allow commenters to register on your site.

What you want to do now is check off the appropriate permissions for your author and click the **Save permissions** button. So what are the appropriate permissions? In the end that is really up to you but I'm going to offer my suggestions in table below as a starting point.

Building Your Blog With Drupal 7

Section	Permission
<i>Comment</i>	Edit own comments
<i>Filter</i>	User the Full HTML text format
<i>Media</i>	Administer media
	Import media files from the local filesystem
<i>Media Internet Sources</i>	Add media from remote services
<i>Mollom</i>	Bypass Mollom protection on forms
<i>Node</i>	Access the content overview page
	View own published content
	View content revisions
	Blog Post: Create new content
	Blog Post: Edit own content
	Blog Post: Delete own content
<i>Overlay</i>	Access the administrative overlay
<i>Search</i>	Use Search
	Use advanced search
<i>Toolbar</i>	Use the administration toolbar
<i>User</i>	View user profiles

Table 3: Recommended Author Permissions

Final Words

If you followed all of the steps in this guide you *should* have been able to create a site with Drupal 7. Of course nothing is ever guaranteed to go as smooth as you would like it to. If you had problems with a specific module I highly recommend that you research the issue queue on Drupal.org for the particular module that is causing you trouble. If you have general questions about this book you can use the Building Your Blog With Drupal 7 forum at <http://learnbythedrop.com/forum> to ask questions or provide feedback.

Building Your Blog With Drupal 7

I hope that you have gotten as much out of reading this guide as I have by writing it. If this guide has helped you then please consider visiting

<http://learnbythedrop.com/buildingyourblog> to explore the ways that you can say,

“Thank you!”