Chapter 7

Essential Contributed Modules

In this chapter I discuss in-depth seven absolutely essential modules to every project I work on. These are modules that will help take your Drupal site to the next level. I cover everything from creating URL path patterns with Pathauto and using views to query and display data to entering content with WYSIWYG HTML editors. This chapter will help you sharpen your Drupal skills while learning about the modules that will set your site apart.

Adding and enabling a contributed module

Contributed modules are located in either the sites/all/modules directory or a site-specific directory, as discussed in Chapter 2. I generally categorize modules into one of two groups; either it’s a module I will use on all future sites or it’s a module that will be used on only one site. If I plan to use a contrib module on future sites moving forward, such as a What-You-See-Is-What-You-Get (WYSIWYG) HTML editor that will be used on every site I develop, then I add the module to sites/all/modules. However, if it’s a site-specific module, such as one custom developed for a single site, then I always add it to the site-specific directory. After you have installed the module, it will appear on the modules page. Select the check box next to the module you need to enable, scroll to the bottom of the screen, and click Save Configuration—you’re done! After a module is enabled, a system message appears at the top of the screen that says the configuration options have been saved, and several links, help, permission, and configure, appear next to the module.

Defining URL path patterns with Pathauto

Pathauto allows you to automatically create customized URLs for nodes, taxonomy terms, and user pages based on user-supplied or system-generated variables (called replacement patterns in Drupal, but we’ll get into that shortly). This means you can create system-generated URLs pointing to specific web pages, and whenever a node, term, or user is created, it is automatically given a URL. For example, you may want all blogs accessed by navigating to example.com/blog/[user:name]/[node:title], where user:name is the user’s name and node:title is the title given to the blog node. Pathauto makes this
very easy by automating the process; you simply create a URL pattern for the content type, and Drupal creates the URL for each page. Cool! This has many applications and is especially relevant for search engine optimization (also known as SEO, which is covered in Appendix A). All URLs added by Pathauto are displayed by navigating to Configuration ➤ URL aliases and clicking the LIST tab, as shown in Figure 7-1.

![URL aliases](Configuration URL aliases)](Configuration URL aliases)

**Figure 7-1.** Navigate to Configuration ➤ URL aliases, and select the LIST tab to view all URL aliases generated by Pathauto.

### Node, taxonomy term, and user path settings

I’ll start with the PATTERNS tab because there are several topics I need to cover first before moving to general and punctuation SETTINGS. Click the NODE PATHS link to view URL replacement patterns for all content types, as shown in Figure 7-2. You can set both the Default path pattern and the content-type specific pattern from this fieldset. All system content types are listed on this page and are blank by default; if a content type does not have a pattern, then it displays the Default path pattern. Click the REPLACEMENT PATTERNS link to view a list of tokens that automatically generate values depending on the node. These are placed into the fields above (such as Default path pattern) and include a number of values, such as the node’s ID, content type, taxonomy term, created date, and more. The Internal feed alias setting is used to generate aliases to specific nodes’ RSS feeds.

When building and managing a site, there are times when you create nodes before configuring Pathauto. Selecting BULK UPDATE tab to automatically create URLs for each node without an alias upon save. There are also times when you make configuration changes to Pathauto and you want to update all old URLs to the new path pattern. The easiest way to do this is to click the DELETE ALIASES tab, delete all the content aliases, and then go back and Select the types of un-aliased paths for which to generate URL aliases from the BULK UPDATE tab. I do not recommend doing this on production sites, because Drupal does not save the aliases after they are deleted, and if users or search engines try to access a page, they will receive a 404 not found error. If you need to change aliases after a site is in