Drupal e-commerce with Ubercart 2.x

Build, administer, and customize an online store using Drupal with Ubercart

George Papadongonas

Yiannis Doxaras

PUBLISHING

BIRMINGHAM - MUMBAI

Drupal e-commerce with Ubercart 2.x

First published: March 2010

Production Reference: 1150310

Published by Packt Publishing Ltd.
32 Lincoln Road
Olton
Birmingham, B27 6PA, UK.

ISBN 978-1-847199-20-1

www.packtpub.com

Cover Image by Filippo (filosarti@tiscali.it)

Credits

Authors
George Papadongonas
Yiannis Doxaras

Reviewer
Jose Argudo

Acquisition Editor
Usha Iyer

Development Editor
Chaitanya Apte

Technical Editors
Bhavesh Bhasin
Ishita Dhabalia

Copy Editor
Sanchari Mukherjee

Editorial Team Leader
Gagandeep Singh

Project Team Leader
Priya Mukherji

Project Coordinator
Leena Purkait

Indexers
Hemangini Bari
Rekha Nair

Proofreader
Chris Smith

Production Coordinator
Shantanu Zagade

Cover Work
Shantanu Zagade

About the Authors

George Papadongonas is the owner of Infowonders (`http://www.infowonders.gr`), a web development firm, located in Athens, Greece. He started his career as a freelance web developer in 2000 and works with small and medium size enterprises. He has experience in many development platforms, but fell in love with Drupal in 2007 and with Ubercart in 2008. He is a member of the Drupal Association and an active supporter of the open source community. His Twitter name is `@infowonders`.

Yiannis Doxaras is a senior open source developer and committer with more than eight years of experience in Java EE, PHP, Python, XHTML, CSS, and all related web frameworks. He has studied theoretical physics at UOA and NTUA. He has published more than 100 articles and posts about web technologies-related matters. Being a serial entrepreneur, he has worked with all major Greek Telcos, banks, and government services. In addition, he has coordinated research for European Community-funded projects and he is also an active innovation agent active in South Eastern Europe. Currently, he is focused on the areas of location-aware and augmented reality applications. Although he lives in Athens, he is a frequent traveler and in his limited free time enjoys running and technical reading. He is a dedicated user of open source technologies in every aspect of his academic and professional work.

Acknowledgement

This book would not have been written without the help and support of some great people: first of all, the Packt team and especially Usha Iyer, Leena Purkait, and Chaitanya Apte. Your guidance and patience during our first writing adventure was amazing.

We would also like to thank Dries Buytaert, the creator of Drupal and Ryan Szrama, the creator of Ubercart. Your personal projects became so important, that they affect the lives of thousands of people daily.

And of course, the Drupal and Ubercart communities for providing ideas, solving problems, and contributing fantastic modules.

About the Reviewer

Jose Argudo is a web developer from Valencia, Spain. After finishing his studies he started working for a web design company. Then, six years later, he decided to start working as a freelancer.

Now that some years have passed as a freelancer, he thinks it's the best decision he has ever taken, a decision that let him work with the tools he likes, such as Joomla!, Codeigniter, CakePHP, jQuery, and other well-known open source technologies.

His desire to learn and share his knowledge has led him to be a regular reviewer of books from Packt, such as *Joomla! 1.5 Multimedia*, *Joomla! With Flash*, *Joomla! 1.5 SEO*, *Magento 1.3 Theme Design*, and *Symfony 1.3 Web Application Development*.

Recently he has even published his own book, *CodeIgniter 1.7* (ISBN 978-1-847199-48-5), which you can also find at Packt Publishing's site. If you work with PHP... take a look at it!

If you want to know more about him, you can check his site `www.joseargudo.com`.

To my brother, I wish him the best.

This book is dedicated to my wife Maria. I feel grateful for every day I spend with you.

George Papadongonas

This book is dedicated to all open source developers out there. Their contributions over the years have provided this book a platform to describe and audience to address.

Yiannis Doxaras

Table of Contents

Preface

Ubercart is an exciting open source e-commerce package that fully integrates your online store with Drupal. Ubercart leverages the advantages of Drupal's major core and contributed systems, providing your users with shopping cart functionality that integrates with other parts of your company or community website. It can be used to sell shippable goods, downloadable products, recurring memberships, and event tickets, and to enable complex interactions with Drupal through various add-on contributions.

This book guides you through the process of creating, administrating, and marketing an online store, providing information and tips for raising your sales and your customer support. It will also show you how to maintain and add advanced capabilities to your company's e-shop.

If you are a merchant wanting to invest your precious time to update the online presence of your company, or a website designer wanting to expand your business and offer innovative solutions to your clients, this book has been written for you. You don't need any programming experience to install and customize your electronic store. The procedure is well structured and easy and all we ask you to do is to follow some simple steps in order to complete some specific tasks in each chapter. The book will guide you through the installation and configuration of Drupal and Ubercart. From there it will cover specific areas like managing orders, customizing the frontend and customizing the user interface. At the end it covers the advanced topics of Internet marketing techniques and Search Engine Optimization.

When you reach the end of the book, you'll have a working store and you'll have obtained all the required knowledge to customize it to your specific needs.

What this book covers

Chapter 1, Getting Started, provides some basic definitions on Drupal and Ubercart. In addition, it sets the context of the online store case study that will be developed throughout the book.

Chapter 2, Installation of Drupal and Ubercart, is a quick installation reference for Drupal, the required Drupal modules, and Ubercart.

Chapter 3, Basic Configuration, helps you to understand the basic Drupal and Ubercart configuration.

Chapter 4, Managing Categories, Products, and Attributes, will show how to manage categories using the Drupal taxonomy, how to create new products or import them from external files, and how to use the attribute system of Ubercart.

Chapter 5, Managing Shipping and Packaging, helps the user to manage shipping and packaging in a flexible way. In this chapter you'll learn how to configure them.

Chapter 6, Managing Taxes and Payments, helps the user to manage taxes and payments in a flexible way. In this chapter you'll learn how to configure them.

Chapter 7, Managing Customers and Orders, is focused on CRM, reporting, and after-sales support.

Chapter 8, Customizing the Frontend, will show you how to install a new theme, customize a theme, or create a new theme from scratch.

Chapter 9, User Interface Enhancement Techniques, helps you to implement more complex marketing techniques.

Chapter 10, Optimizing and Promoting Your Store, will show you how to make an Ubercart store faster, more secure, and more search engine friendly.

Appendix A, Hotel Bookings System for Ubercart, shows how Ubercart can be used not only for selling products, but also for offering services as well.

Appendix B, Modules Used in the Book, lists the modules you can download when working through the exercises for each of the chapters.

Appendix C, Free and Commercial Ubercart Themes, gives you details of some very interesting free and commercial Ubercart themes.

What you need for this book

Actually, you only need a computer to start working with this book.
In Chapter 2 we'll show you how to install PHP, a database, and a web server, if they aren't already installed on your system. We'll also show you how to use a remote server and install you store there.

Who this book is for

If you are a business owner wanting to create an e-commerce website for your company or to upgrade a company e-shop to be easier to maintain and have advanced e-shopping capabilities, then this book is meant for you. You don't need any programming experience to use this book.

Conventions

[ ]

[ ]

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to `feedback@packtpub.com`, and mention the book title via the subject of your message.

If there is a book that you need and would like to see us publish, please send us a note in the **SUGGEST A TITLE** form on `www.packtpub.com` or e-mail `suggest@packtpub.com`.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book on, see our author guide on `www.packtpub.com/authors`.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting `http://www.packtpub.com/support`, selecting your book, clicking on the **let us know** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from `http://www.packtpub.com/support`.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at `copyright@packtpub.com` with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at `questions@packtpub.com` if you are having a problem with any aspect of the book, and we will do our best to address it.

1
Getting Started

Building an e-commerce site used to be a very difficult and expensive procedure. A decade ago, if you wanted to build one, you had to hire one or more experienced developers to build an application for you from scratch, after working for several weeks. You had to pay a fair amount of money to build, customize, and support this application. In those days, only big companies could afford this cost.

Nowadays, things are much simpler: e-commerce applications are lighter, faster, and of course cheaper, which gives each merchant the capability to sell his or her products online. If you are a merchant wanting to invest your precious time to update the online presence of your company, or a website designer wanting to expand your business and offer innovated solutions to your clients—this book is written for you. You don't need any programming skills to install and customize an electronic store. The procedure is well structured and easy, and all we ask you to do is to follow some simple steps in order to complete some specific tasks in each chapter.

When you reach the end of this book, you'll have a working store and you'll have obtained all the knowledge required to customize it according to your specific needs.

What is Drupal?

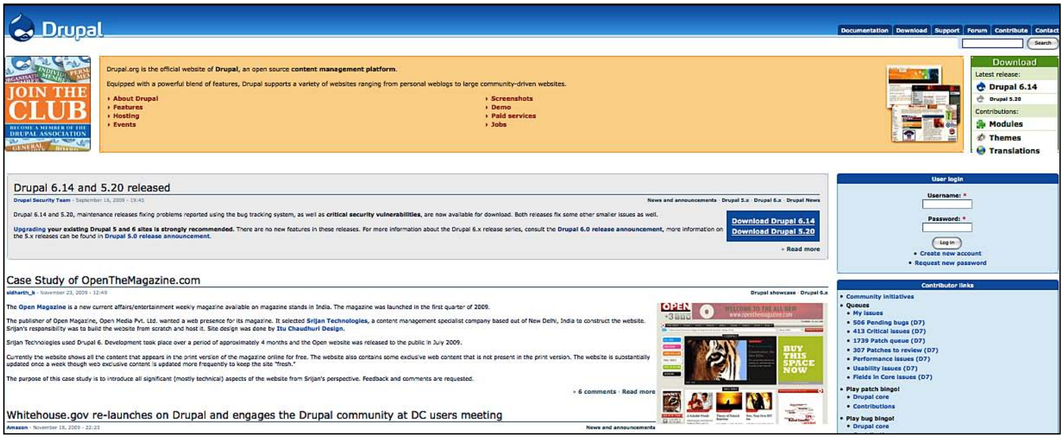
Drupal is an award-winning, free, and open source **Content Management System** (**CMS**), winner in two categories of the **Packt Publishing 2008 Open Source CMS Awards** (`http://www.packtpub.com/article/2008-open-source-cms-award-winner-announced`). Drupal's source code is freely distributed under the terms of the **GNU General Public License 2** (**GPL**). The GPL, unlike permissive free software licenses like BSD, is an example of a powerful copyleft license that imposes constrains on all the derived software components to be available under the same copyleft. Therefore, GPL grants the rights of the free software definition to the recipients of a computer program, and uses copyleft to ensure that the freedom is preserved, even when the software is changed or added to.

Unlike proprietary blogging or content management systems, Drupal's feature set is fully available to extend or customize as needed.

Drupal was started as a small content-sharing personal platform by Dries Buytaert in the University of Antwerp campus. When Dries left the university, it was published on the Internet to continue communication with the group. Thus, Drupal was hosted on Drop.org at first, until it became very popular with users asking for more and more functionality. Then Dries decided to make it open source software and started maintaining it with other code contributors from the open source community. Gradually, it became state-of-the-art software with a devoted community with hundreds and thousands of members. There are millions of websites built with Drupal, from personal blogs to the White House web page.

The main advantages of Drupal are:

- It's open source software.
- It has an active community, ready to support every new member.
- It is optimized for performance and is scalable.
- It is a social media ready with features such as forums, blogs, polls, and managed communities.
- It provides out-of-the-box advanced auditing, reporting, and statistics system.
- It provides extensive support for multilingual sites.
- It is platform and web server independent, and can be deployed on any environment (Windows, Unix, and Mac OS X) in single instances or in a cloud.
- It is database independent and can easily be linked with any major open source database.
- Drupal modules can extend its functionality. New third-party modules are created every day by individual developers.
- Drupal is not oriented only to developers, but to simple users as well.



To learn more about Drupal, browse to `http://www.drupal.org`.

What is Ubercart?

Ubercart is not a standalone application, but it's a module for Drupal. This means that you first have to install and customize Drupal, and then install Ubercart and all its related third-party modules. Imagine how powerful this combination is. You won't have just a selling point for your products, but a complete platform to interact with your clients and market your company.

Some key aspects of Ubercart that make it the number-one solution for online shops with Drupal are:

- Transparent integration with Drupal basic core resulting in advanced reporting, user administration, and community-building features
- Fully extensible with community wide support as it implements Drupal's core API programming patterns
- Out-of-the-box functionality for each key aspect of your online store such as products, checkouts, payments, orders, and shipments
- Activity logging and auditing
- Batch import/export of products in the product repository
- Integrated payment system to integrate between various payment methods and known payment gateways

For the end user to visually comprehend the rich features of the Ubercart and Drupal platform, the following are some examples of success stories. All shops are built on top of Drupal with extensive use of Ubercart and its robust product catalog:

- `http://www.printcasting.com/`, an online store with fixed layout for ad brooking on printed templates (top left)
- `http://www.coco-mat.com`, a store for fabrics and furniture with a flash animated product catalog (bottom right)
- A top notch themes proof of concept for an online store selling office furniture (bottom left)
- A Norwegian online bookstore and DVD shop (top right)

To learn more about Ubercart browse to `http://www.ubercart.org` or you can also visit a live site example page at `http://www.ubercart.org/site`.

Book details

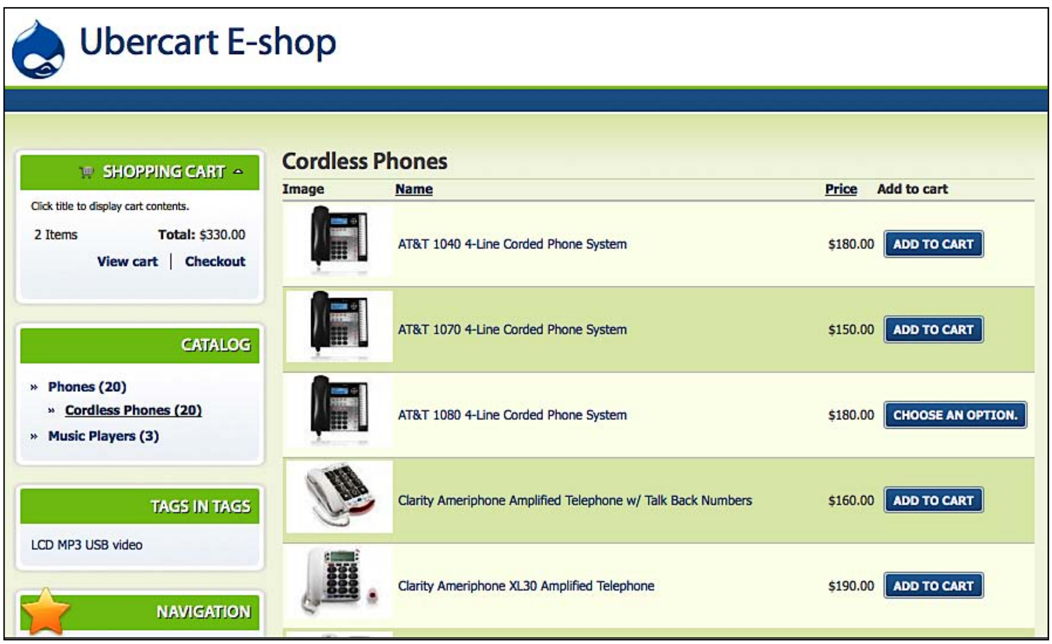
This book will show you how to build a basic store using Drupal and Ubercart. We will guide you through the administrative section of the application and show you how to customize your store. Then, you'll learn how to manage categories, how to create products, and how to import them from external files.

As we proceed, you'll customize shipping, taxes, and payments, and learn about order management and customer relationship management. You'll install a ready-made Drupal template and you'll also learn how to create one from scratch. Finally, you'll learn how to create a better user interface and optimize your store.

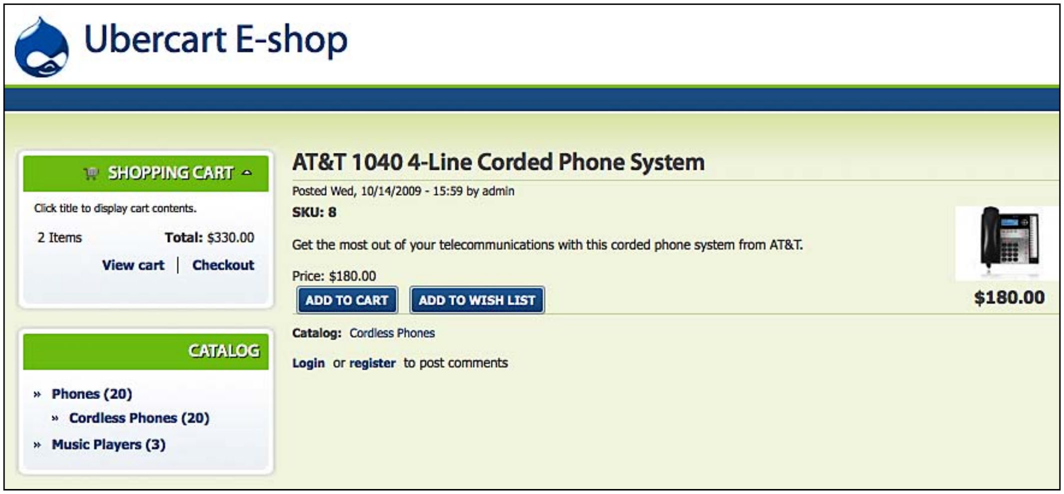
The final store

The store that we are going to build in this book is a fully working electronics store.

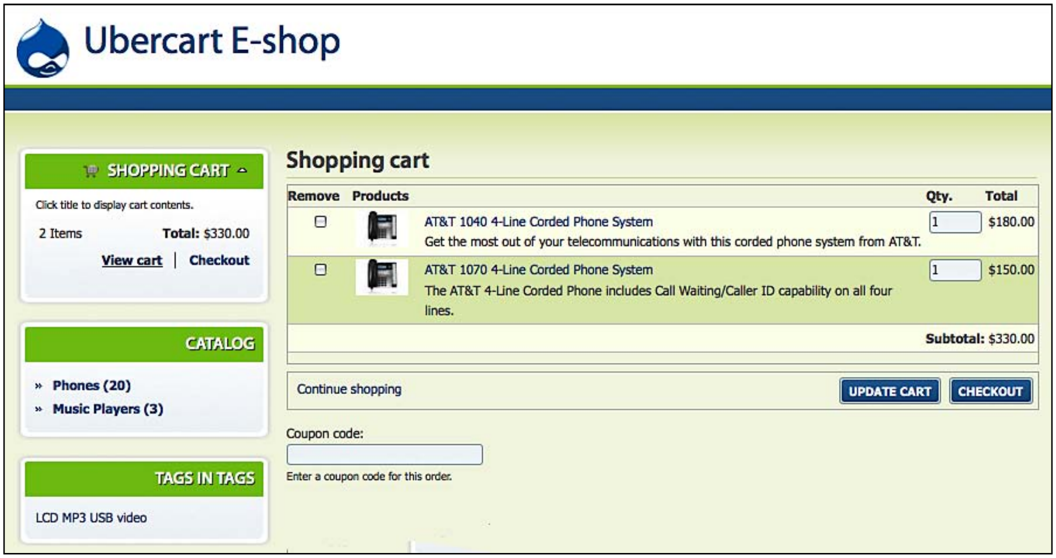
So, let's see the most important sections.



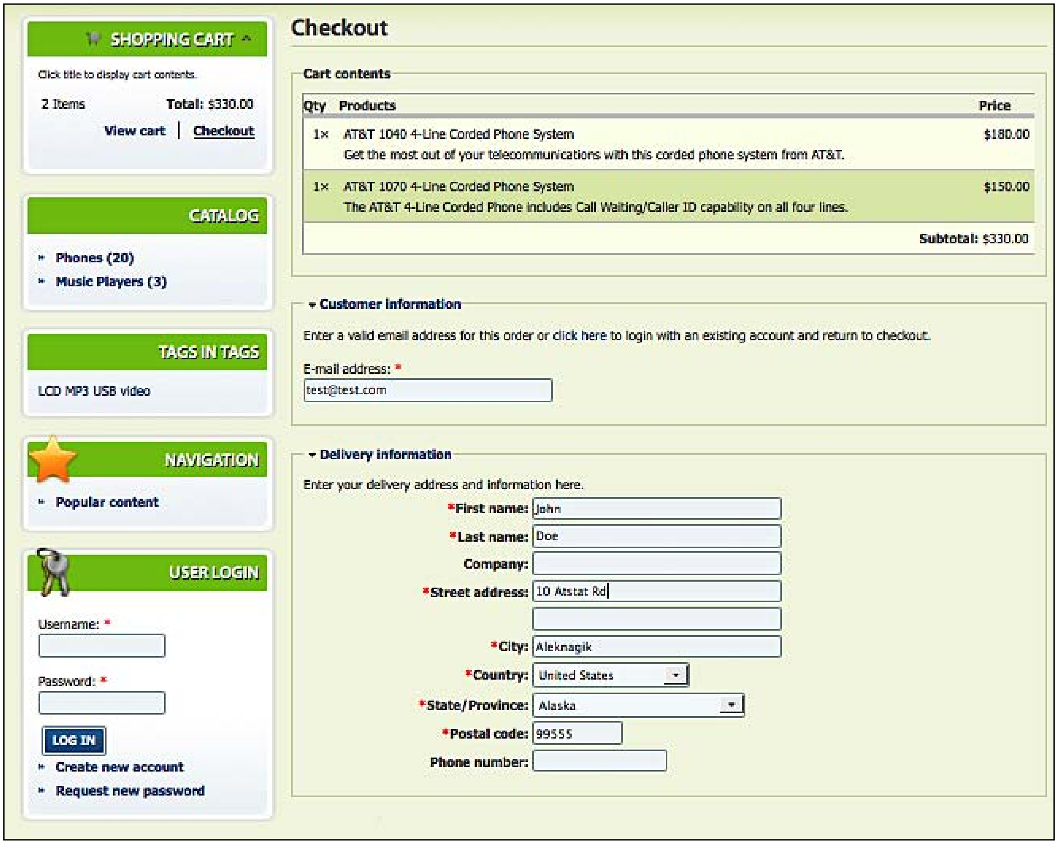
The product catalog is the main navigation system in our store. We have our main categories and subcategories, offering a structured way to the clients to browse the website and locate their favorite products. When the visitors reach a page with no more subcategories, they can see a list of all the available products of this category.



When clients click on a product from the catalog, they can see all the details of this product. If it has attributes, like size or color, they also have to select them. Finally, they can add this product to the shopping cart or the wish list, depending on the functionality that we have installed.



The shopping cart is the page where the customers can check all the products that they have added so far. They can update the quantities, remove one or more products, or continue to the checkout page.



The checkout page is the final step of the shopping process. Here, the customer enters his or her delivery and billing information, selects a payment and a shipping method, and completes his or her order. After that, you, the shop owner, have to collect the products and fulfill the order.

Of course, one of the most important sections of the website is the administration section. You gain access to it by entering your username and password. Throughout this book, you'll spend all your time in this section, customizing not only the store pages, but also the complete Drupal installation.

Summary

In this chapter, we have gone through the following key points:

- An introduction to Drupal, its history and development, and its main features and advantages
- An introduction to Ubercart with some samples of success stories of existing online stores using it
- The general scope of this book
- Visual and conceptual paradigms of the final outcome of the online store

In the next chapter we are going to install a basic Drupal site, add the Ubercart module, and start making customizations.

2
Installation of Drupal and Ubercart

The beginning is half of everything. A proper installation of Drupal and Ubercart will save you from future headaches and will guarantee the correct functioning of your online store. This chapter is a quick installation reference for Drupal, the required Drupal modules, and Ubercart.

Ubercart is not a standalone e-commerce application, but it comes as a Drupal module. That means you have to first install Drupal and all the required Drupal modules on a server with the minimum requirements, before installing Ubercart. You don't have to be an expert programmer or a system administrator in order to complete the following process. We'll first give you a brief explanation of the underlying technologies, and then we'll continue with a step-by-step guide. At the end of this chapter, you'll have the online store installed on your local or remote machine and you'll be ready to make all the required customizations to the frontend and the backend, depending on your needs.

The topics that we will discuss in this chapter are:

- Minimum requirements for Ubercart installation
- Creating a local environment using a web server, PHP, and a database server
- Using a commercial hosting service
- Downloading and installing Drupal and Ubercart
- Using UberDrupal, an Ubercart installation profile

Minimum requirements for Ubercart installation

In order to successfully install and use your online store, your system must meet the following requirements:

Operating system	Windows, Mac OS X, Linux, Unix, BSD, or Solaris
Web server	Apache 1.2 or Apache 2.x, Microsoft IIS 6 or 7, lighttpd
Database	MySQL 4.1 or MySQL 5.0, PostgreSQL 7.4
PHP	PHP 4.3.5. is required, but PHP 5.2.x is recommended

Now, let's elaborate a little more on the above requirements:

- **Operating system**: Drupal works fine in almost every operating system. Actually, you can transfer your Drupal installation from one operating system to another within minutes and no customization is required at all. All you have to do is to move the files and the database without altering any configuration files. For example, you can install Drupal on your local Windows or Mac computer, do all the customizations there, and then upload it to a Linux server to go live.
- **Web server**: The web server is the software that accepts HTTP requests from browsers and delivers web pages to the users of our site. The most popular web server is Apache and we'll use it for our installation. It's secure, extensible, fast, and easy to customize. If you're not an expert in another web server, there is no reason to think of any other solution, because most of the available information and support is about Apache.
- **Database**: The purpose of the database is to store, organize, manage, and retrieve all the data of our website in a structured way. When referring to data, we mean not only the content that you put in your pages, but also every piece of information that Drupal uses for all its functions. In this book, we're using MySQL as a database. Today, it's the #1 open source database, and it's used in millions of websites and applications, from small personal web pages to enterprise systems with millions of users.

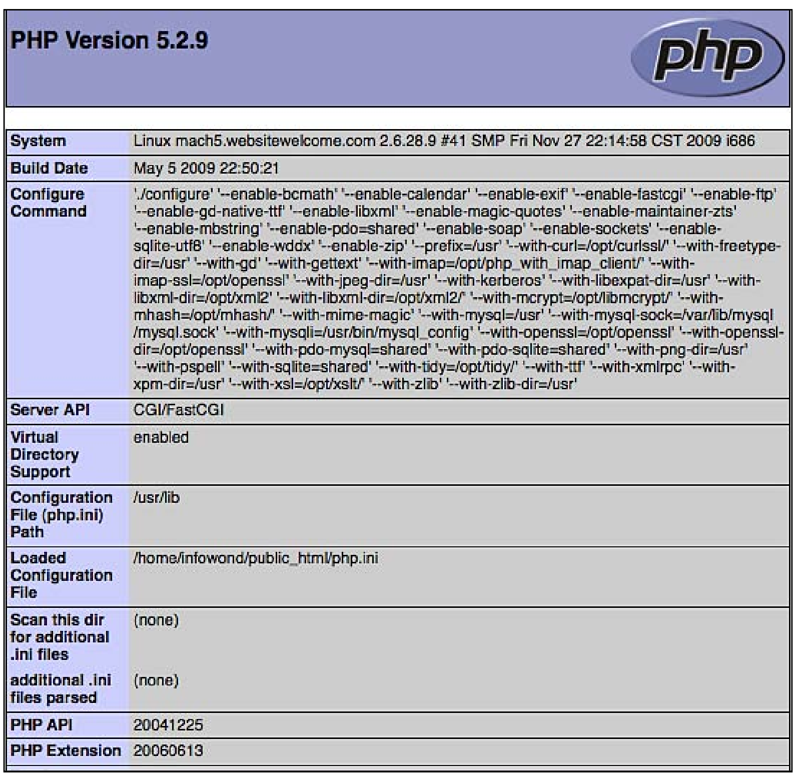
  The MySQL database of a basic installation contains about 50 tables and every new installed module creates one or more new tables. If you check your database after the installation of Ubercart, you'll find that there are 100 tables in your database. These tables contain data such as pages, products, images, categories, orders, payments, caching information for your pages, theming information, comments from your visitors, menus, user information, and so on.

- **PHP**: PHP is a scripting language, ideal for web development. It began as a small personal project, but soon became a very popular scripting language. Drupal is written in PHP, so it's absolutely necessary and there is no alternative to it. Drupal 6 works fine with PHP 4, but PHP 5 is recommended and that's what we're going to use in this book.

  Some PHP extensions are needed for our installation. We'll mention them briefly here, so you can consult your hosting provider or examine your local system to check that everything is fine. The easiest way to check your PHP parameters is to use `phpinfo`. It's a function that returns information about the PHP environment on your server. All you have to do is to create a blank file named `phpinfo.php` in your server and insert the following code using a text editor:

```
<?php
  phpinfo ();
?>
```

  This is a small PHP script that runs the `phpinfo` function and displays the results on your screen. If you browse to `phpinfo.php`, you'll see a page with your PHP configuration.



PHP Version 5.2.9

System	Linux mach5.websitewelcome.com 2.6.28.9 #41 SMP Fri Nov 27 22:14:58 CST 2009 i686
Build Date	May 5 2009 22:50:21
Configure Command	'./configure' '--enable-bcmath' '--enable-calendar' '--enable-exif' '--enable-fastcgi' '--enable-ftp' '--enable-gd-native-ttf' '--enable-libxml' '--enable-magic-quotes' '--enable-maintainer-zts' '--enable-mbstring' '--enable-pdo=shared' '--enable-soap' '--enable-sockets' '--enable-sqlite-utf8' '--enable-wddx' '--enable-zip' '--prefix=/usr' '--with-curl=/opt/curlssl/' '--with-freetype-dir=/usr' '--with-gd' '--with-gettext' '--with-imap=/opt/php_with_imap_client/' '--with-imap-ssl=/opt/openssl' '--with-jpeg-dir=/usr' '--with-kerberos' '--with-libexpat-dir=/usr' '--with-libxml-dir=/opt/xml2' '--with-libxml-dir=/opt/xml2/' '--with-mcrypt=/opt/libmcrypt/' '--with-mhash=/opt/mhash/' '--with-mime-magic' '--with-mysql=/usr' '--with-mysql-sock=/var/lib/mysql/mysql.sock' '--with-mysqli=/usr/bin/mysql_config' '--with-openssl=/opt/openssl' '--with-openssl-dir=/opt/openssl' '--with-pdo-mysql=shared' '--with-pdo-sqlite=shared' '--with-png-dir=/usr' '--with-pspell' '--with-sqlite=shared' '--with-tidy=/opt/tidy/' '--with-ttf' '--with-xmlrpc' '--with-xpm-dir=/usr' '--with-xsl=/opt/xslt/' '--with-zlib' '--with-zlib-dir=/usr'
Server API	CGI/FastCGI
Virtual Directory Support	enabled
Configuration File (php.ini) Path	/usr/lib
Loaded Configuration File	/home/infowond/public_html/php.ini
Scan this dir for additional .ini files	(none)
additional .ini files parsed	(none)
PHP API	20041225
PHP Extension	20060613

The basic PHP requirements are:

- **PHP memory requirements**: 16 MBs are enough for basic sites; however, 64 MBs are recommended for more complex installations.
- **GD2 library**: GD2 is an image manipulation library for PHP. Ubercart needs this library, so if we want to put images to our products, it has to be installed.
- **Register Globals**: This is actually a depreciated PHP feature, but some hosting providers with old systems still use it. It's a security risk, so it has to be disabled for Drupal to install.
- **Safe mode**: Drupal 6 doesn't support PHP's safe mode, because it causes problems to file uploads, so it also has to be turned off.

Creating a local environment using a web server, PHP, and a database server

When you start building your online store, it's better to do it on a local environment. There, you can do all the tests, experiment, try different options and solutions, and correct all the problems in a closed and secure environment before going live.

You have two basic choices: manually install and configure all the components, or use a complete web server package.

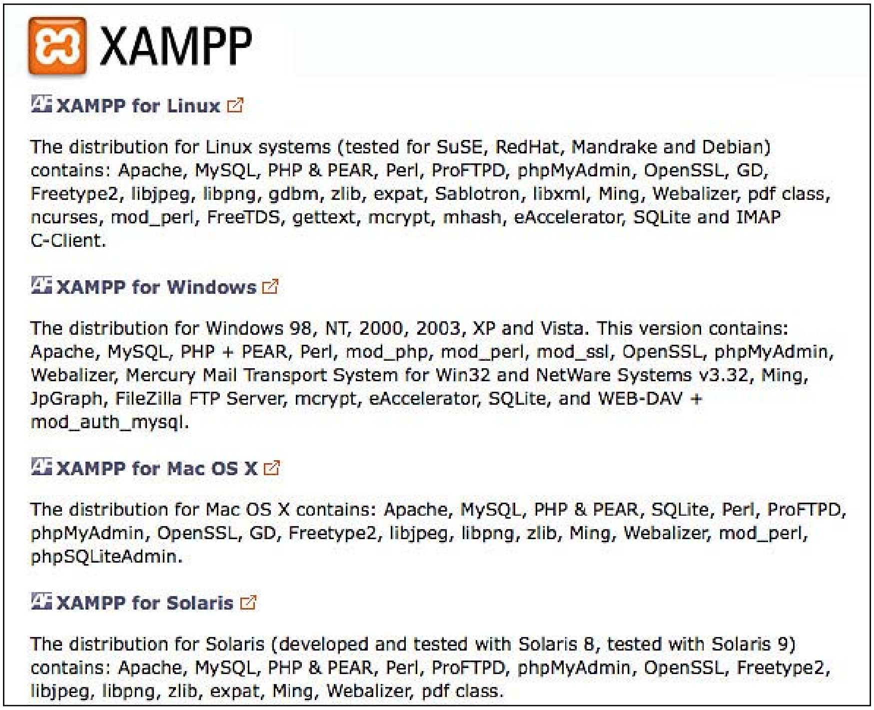
The advantage of the first choice is that you have the absolute control of the entire process, but its disadvantage is that it's time consuming and sometimes difficult to configure it right.

We prefer the second method, because it's fast, easy, and reliable.

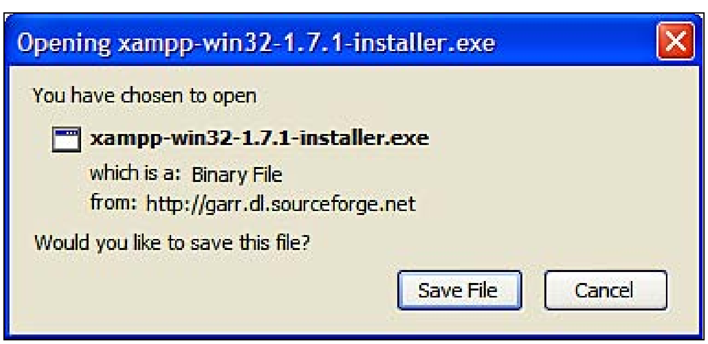
We recommend XAMPP (`http://www.apachefriends.org/en/xampp.html`), a free distribution package for Linux, Windows, and Mac OS X, containing Apache, PHP, MySQL, OpenSSL for Secure Sockets Layer support, ProFTPD FTP server (FileZilla in the Windows version), and phpMyAdmin for the administration of MySQL databases. It's free, easy to install and uninstall, needs little or no configuration, easy to use, and very fast and stable.

So let's start the installation process:

1. Go to `http://www.apachefriends.org/en/xampp.html`. This is the download page for XAMPP. You'll see there that there are four basic distributions for Linux, Windows, Mac OS X, and Solaris. Here, we'll show you how to install it for Windows, but the process is almost the same for every other operating system.



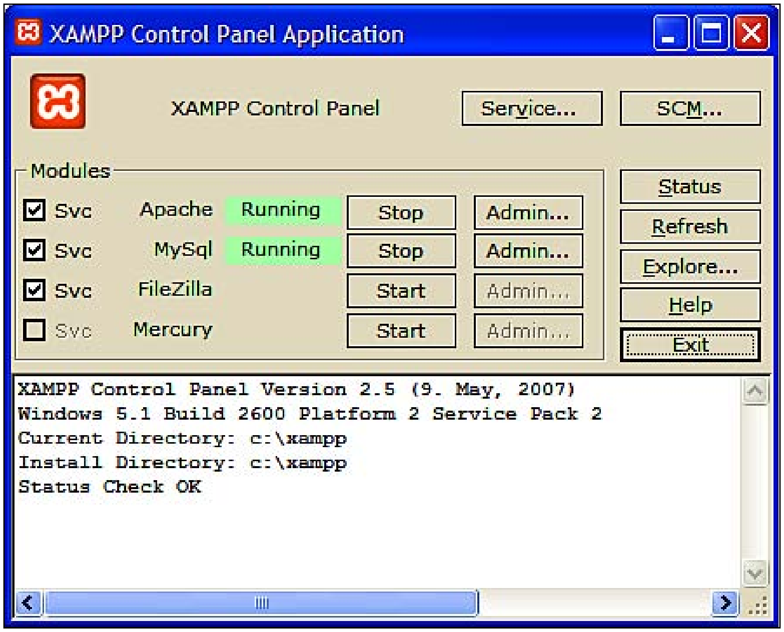
2. Select the appropriate distribution for your operating system. You will be transferred to a new page, where you can see specific details, frequently asked questions, some basic installation instructions, and configuration options. Click on the name of the file and select **OK** to download the file on your computer. It takes only a few minutes, depending on your Internet connection.



3. When the download is completed, double-click on the file in your computer to open it and start XAMPP Setup Wizard. Leave the default settings and click on **Next>** until the installation process is finished.

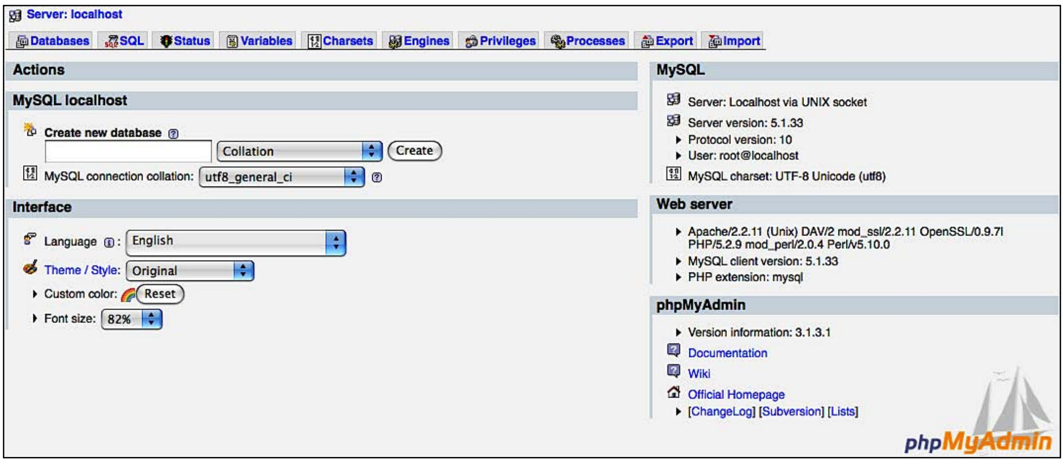
If you check the root folder of your hard disk, you'll now see a new folder named `xampp`. Inside, there are all the files of the package. Among all the others there is one application file named `xampp-control`, which opens the control panel for XAMPP.

4. Double-click on `xampp-control` and open it. The Control Panel opens in a new window. It has a very simple layout, showing the three main applications (Apache, MySQL, and FTP), an indicator about the status for each one of them, and a button to turn them ON or OFF. If everything is OK, you'll see green indicators showing that all XAMPP components are **Running**.



5. Now it's time to test that everything works. Using your browser, open the following URL: `http://localhost`. Select your language and you'll be transferred to the home page of XAMPP. On the left, there is a menu for all the basic actions. You can check the status of the applications, check the security of your installation and your pages, read online documentation about Apache, PHP, and MySQL, test and see the source code of some very interesting demo applications, and use some very useful tools like phpMyAdmin for the administration of MySQL databases and Webalizer for Web Stats.

6. Before you download Drupal, you have to create a blank MySQL database. This database will be used by Drupal to store all the data of your site. So, from the left menu of the XAMPP home page, select **phpMyAdmin**, located at the bottom of the page. It will open in a new window. Under the label **Create new database**, enter the name of your database and click on the **Create** button to continue. If you haven't changed anything, the default MySQL username is `root` and there is no password.



That's it! Now your local environment is ready to install Drupal. If you're not going to use a commercial hosting service immediately, skip the next section and go to the *Downloading and installing Drupal* section.

Using a commercial hosting service

If you're in a hurry, or you just don't want to mess about with local web server installations, you can find a commercial hosting service and host your online store there. Nowadays, most companies support Drupal without any problem, because it's very popular and there is high demand for it. Look for a provider with reliability and speed of access, with plenty of web space and bandwidth. You need FTP access to upload the required files, and enough privileges to create a new database and to edit `.htaccess` and `php.ini` files.

Go to `http://drupal.org/hosting` to find a list of companies that provide Drupal hosting services.

When you select your hosting company and sign up, they'll give you a username and password and a link to access the Control Panel of your package and manage your website. In this section, we'll work with cPanel, but the process is almost identical for the other GUI control panels such as Plesk.

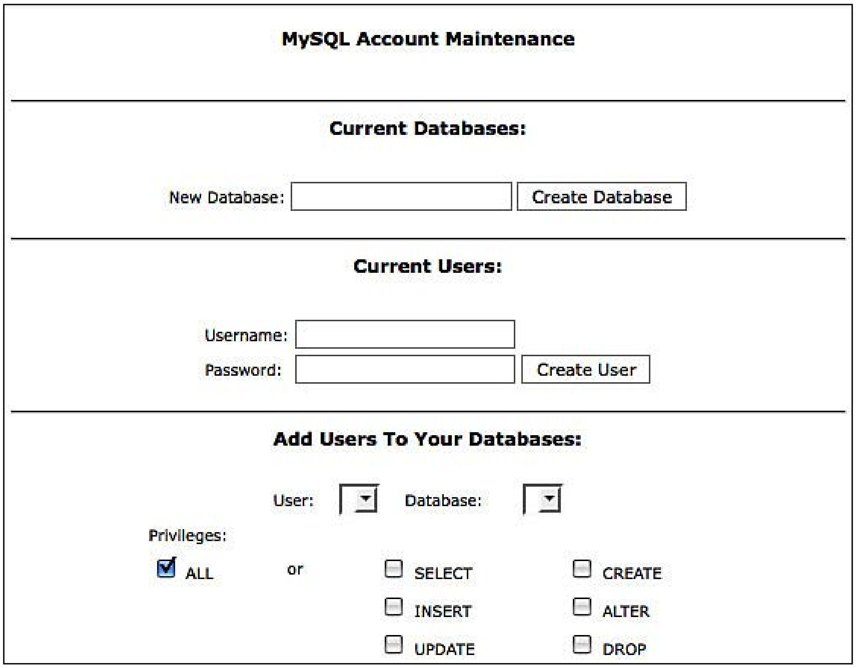
Before installing Drupal, you have to create a database. We'll show you here how to create a MySQL database, but a similar process is followed for a PostgreSQL database.

1. Browse to the URL of your cPanel (usually `http://www.mysite.com/cpanel`). Enter your username and password.

2. You'll be redirected to the home page of cPanel. Select **MySQL® Databases** from the menu.

3. Enter a name for your database and click on the **Create Database** button.
4. Enter a username and a password and click on the **Create User** button.

5. Check the **ALL Privileges** box and click on the **Add Users to Your Databases** button.



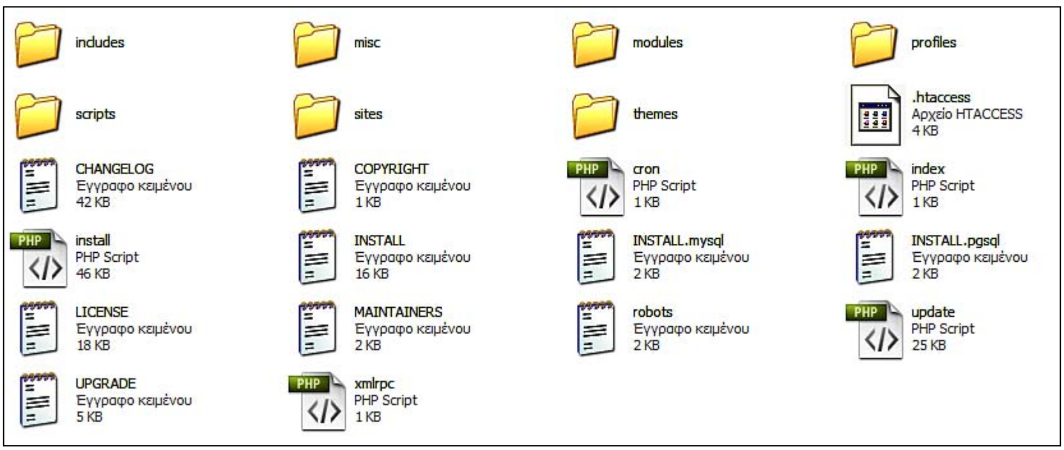
6. Don't forget to take a note of the database name, the username, and the password; we'll use them during the Drupal installation.

Downloading and installing Drupal

No matter if you decided to use a hosting service or to set up a local server; the process for installing Drupal is almost the same. There are only minor differences, which will be mentioned as we proceed with the installation.

The first step is to download, install, and customize Drupal. Click on **Latest Release** from the home page (`http://drupal.org`) for immediate download of the latest version of Drupal, or browse to `http://drupal.org/project/drupal` to check all the versions available. The installation file comes as a compressed file under the format `drupal-6.x.tag.gz`. If you are a Linux or Mac user you can immediately open the file, but if you are using Windows you may need a program like 7-Zip (`http://www.7-zip.org`).

Unzipping the file will create a `Drupal-6.x` folder in your machine. If you check this folder, you'll see that it has more folders in it and that it contains all the files of the Drupal installation.



If you're working on a local server, copy this folder to `c:/xampp/htdocs`. If you're using a hosting service, you have to upload these files to the remote server, using an FTP application such as FileZilla (`http://filezilla-project.org`).

If you go to `sites` and select the default folder of our Drupal installation, you'll find only a file there, named `default.settings.php`. Copy this file to `settings.php` and remember not to delete the original file, because we need both files in order to install Drupal. Uncheck the **read only** property of `settings.php` file if you're working locally, or change the permissions to `666` with the file manager, if you're using a hosting service, so that the file is writable for Drupal installer.

Now, it's time for action; the installation begins! Browse to the **root folder of your** installation. You'll see the first screen of Drupal installer.

Select **Install Drupal in English**. If you want to install Drupal in another language, you have to first download the appropriate language package from `Drupal.org` and follow the instructions.

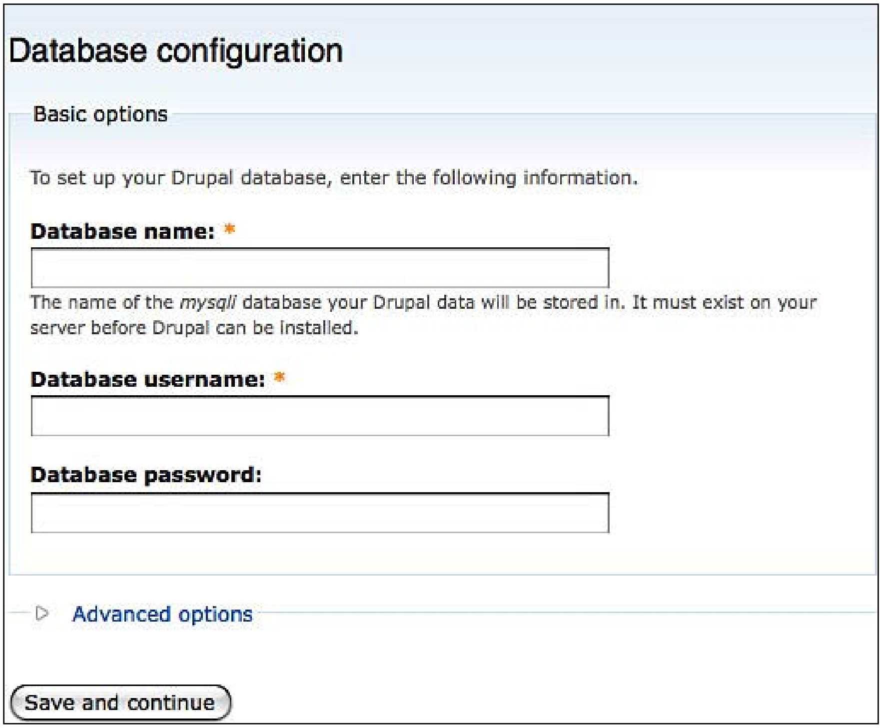
There is a possibility of facing an error message at this stage. It shows up when `register_globals` is enabled. This is a security risk for your website and you have to solve it before continuing.

You have to create a file named `php.ini` in you root folder and add the following line:

```
register_globals = Off
```

This disables `register_globals` and now you can safely continue. If this doesn't work, you may not have enough privileges in your hosting account and you will have to contact your provider to fix it.

Drupal installer verifies that your system meets all the basic requirements, and then you proceed to the next screen. There, you have to write the database name, username, and password that you inserted when you created the database. (You DID take a note of them, didn't you?) Click on **Save and continue** and you'll see the installer progress bar. When all the files are installed, it's time for the basic configuration screen.

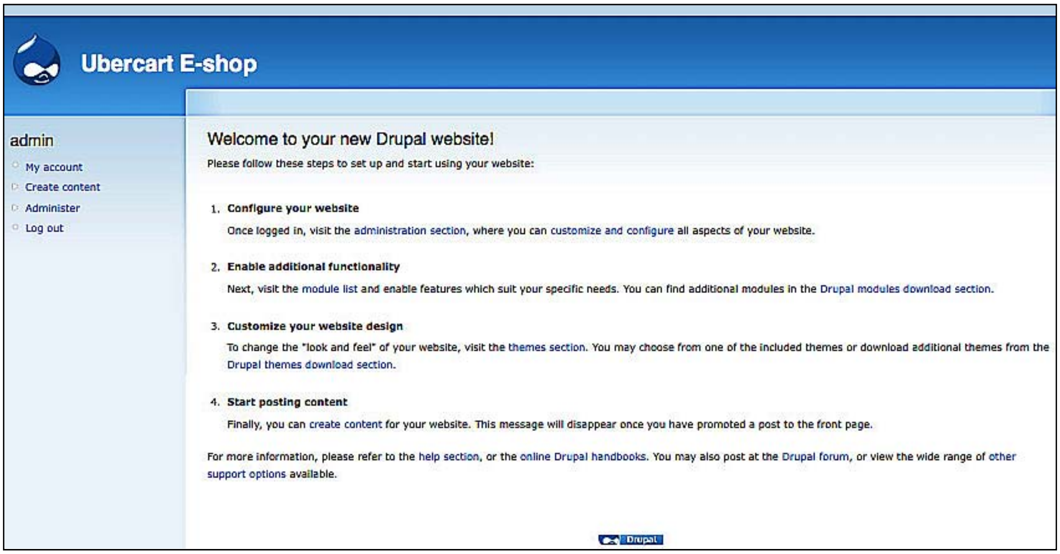


Here, you have to insert the site name and the primary site e-mail address.

Then, you have to create the administrator account. The administrator is a "super user" who has all the privileges and has access to every operation of your website. You have to keep the administrator credentials in a safe place. If you lose them, it's possible to lose the control of your website. Insert the administrator's username, e-mail address, and password.

In the last section of the page, you have to select time zone, decide if you want clean URLs enabled or disabled (enabling clean URLs makes Drupal create shorter and more search engine-friendly URLs), and choose if your system will check for updates automatically. IIS servers need an extra module to create clean URLs. For more information, read `http://drupal.org/node/3854`.

Click on **Save and continue** and you're done! Your first Drupal website is ready!



Now you can see the first page of your new site. Of course, it's not very pretty yet, and there is no actual content in it yet, but we took our first big step.

Downloading and installing all the required Drupal modules

Now we have a functional Drupal site. Can we proceed with Ubercart installation? Not yet! First we have to install a few more Drupal modules.

Drupal third-party modules are not part of the core, and they are created by individual programmers or companies. They add extra functionality to Drupal or they enhance an existing functionality. You can see a full list of all the available Drupal modules at `http://drupal.org/project/Modules`.

Actually, the only module that is necessary for Ubercart to work is **Token**.

As we can read at the home page of this module, `http://drupal.org/project/token`:

> *Tokens are small bits of text that can be placed into larger documents via simple placeholders, like %site-name or [user]. The Token module provides a central API for modules to use these tokens, and expose their own token values.*
>
> *Note that Token module doesn't provide any visible functions to the user on its own, it just provides token handling services for other modules.*
>
> *For Drupal 6, the Token module provides a "Token Actions" module which can be enabled separately. This provides several "actions" for the Drupal core Actions/ Trigger modules to use that take advantage of the Token replacement functionality.*

Ubercart uses Tokens in several functions, like confirmation messages or client e-mails.

The next bunch of modules is required for image support. We want our products to have nice images, and in several sizes. We also wish to enable our clients to magnify these images. Drupal does not provide any out-of-the-box image support, so Ubercart uses not only one, but six different modules. The installation of all these modules seems like a time-consuming and complicated process, but they upgrade Drupal, providing new capabilities required for our e-shop to function. The modules that we have to install are:

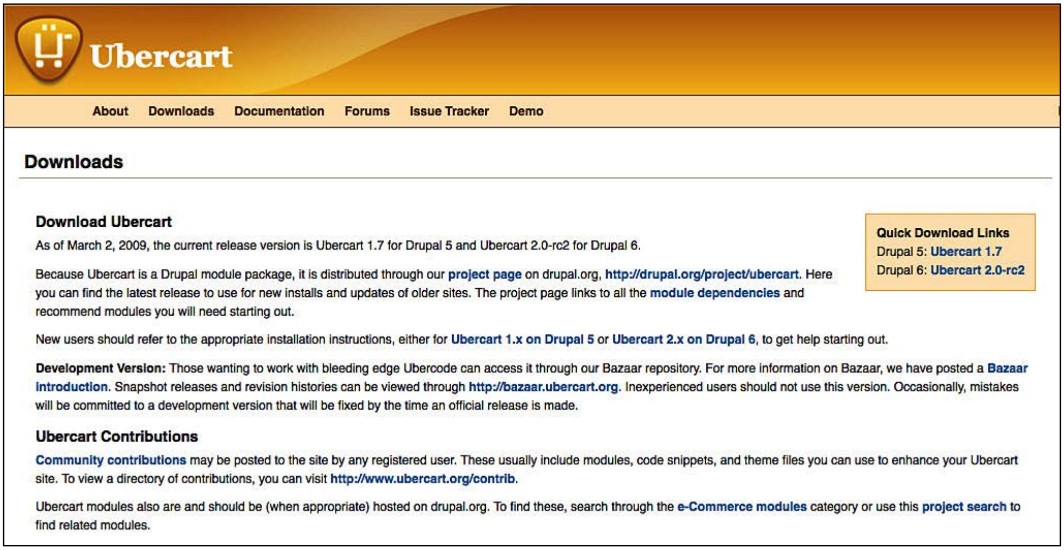
- **Content Construction Kit** (**CCK**): This is the most important contributed module for Drupal. It allows the user to create new content types and to add new fields to existing content types, using only the administrator's interface, without the need of any programming knowledge. This module's page is `http://drupal.org/project/cck`.

- **FileField**: This module is an add-on for CCK. It creates a field for file uploads. This module's page is `http://drupal.org/project/filefield`.
- **ImageField**: This module is also an add-on for CCK. It creates an image field. This module's page is `http://drupal.org/project/imagefield`.
- **ImageAPI and ImageAPI GD2**: ImageAPI uses PHP's GD2 image manipulating library. It's extremely useful, because it allows us to take basic actions, such as resizing, rotation, adding watermarks, cropping, or converting to another format directly from the browser, without the need to invest in an image-editing application. This module's page is `http://drupal.org/project/imageapi`.
- **ImageCache**: This module allows us to create predefined standards for images. Ubercart uses ImageCache to adjust all the images of the products that we upload to specific preset dimensions and dynamically generates the files for product catalog, thumbnails, and shopping cart photos. This module's page is `http://drupal.org/project/imagecache`.
- **Thickbox**: This module brings all the functionality of the jQuery plugin Thickbox (`http://jquery.com/demo/thickbox`) to Drupal. Clicking on an image loads the full-size version of it in a new layer, without reloading the whole page. It provides automated integration with all of the aforementioned modules. This module's home page is `http://drupal.org/project/thickbox`.
- **Google Analytics**: This module adds the Google Analytics web statistics solution to your website. Not only does it collect general stats such as number of visitors, most popular pages, or bounce rate, but it also tracks transaction data and item data. E-commerce tracking and analytics is a very useful tool that helps you to analyze the performance of your business and to manage your marketing strategy. This module's page is `http://drupal.org/project/google_analytics`.
- **Views**: This module provides the site administrator with a web interface that allows easy presentation of the website content. In *Chapter 8, Customizing the Frontend*, we'll see some creative ways to use Views, such as creating custom lists of our products and custom functionality like up-selling and cross-selling. This module's page is `http://drupal.org/project/views`.

To install all of these modules, first you have to download them from their home page. Then create the `sites/all/modules` folder, unzip the files, and copy the whole directories there. Finally, browse to `http://localhost/admin/build/modules`, select these modules from the (long and overwhelming) list and click on **save**.

Downloading and installing Ubercart

Now that we finished with Drupal installation and have enabled all the required Drupal modules, it's finally time to install Ubercart. Browse to `http://www.ubercart.org/downloads` and download the latest Ubercart 2.x version for Drupal 6. This also comes as a compressed `tar.gz` file, so you have to uncompress it just as you did for Drupal 6.



Then, you have to copy it to `sites/all/modules` directory, as we did with the previous modules. Actually, Ubercart is not just one module, but rather a package of modules. So now, we have plenty of new options on our module page. Some of them are required (Ubercart-core) and some of them are optional (Ubercart-core (optional) and Ubercart-extra, Ubercart-fulfillment, and Ubercart-payment). You don't have to install every single Ubercart module. Spend some time now and think what you need and what you don't. Of course, if you change your mind you can return here and add or remove features, depending on your needs.

Required modules

The most important modules of the Ubercart package are:

- **Cart**: Required for a functioning shopping cart
- **Conditional Actions**: The conditional actions are required to configure shipping and taxes
- **Order**: Required for order management and fulfillment
- **Product**: Creates a product content type for your store
- **Store**: Required for store setup and management

Optional core modules

You can install a basic but functioning version of Ubercart without using these modules, but in most cases they are needed for a competitive and attractive store:

- **Attribute**: Using attributes, you can create a customizable product that differentiates based on specific characteristics, such as size or color.
- **Catalog**: This module creates a product catalog with categories and subcategories.
- **File downloads**: This module is required if you want to sell downloadable products, such as music or video files.
- **Payment**: This module is required for receiving payments at your store.
- **Report**: This module creates reports about your store, orders, products, and clients.
- **Roles**: A role can give different privileges to different clients.
- **Shipping**: This module is required for shipping management.
- **Taxes**: This module is required for tax management.

Extra modules

These extra modules add even more functionality to Ubercart, transforming your store into a competitive selling point:

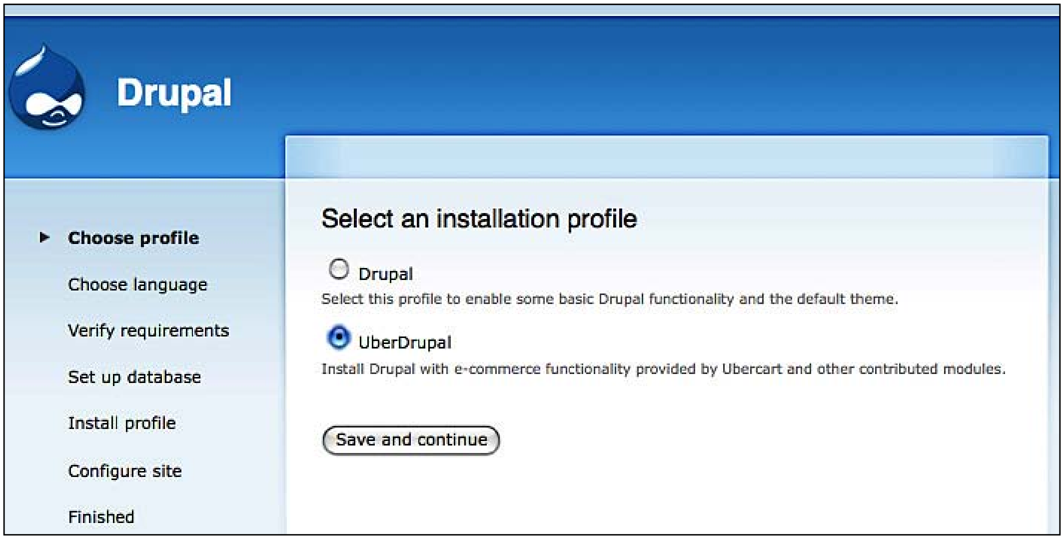
- **Cart Links**: This module lets you create specialized links to purchase products from other nodes.
- **Google Analytics**: This module installs Google Analytics for Ubercart.
- **Product kit**: Using this module you can combine two or more products and sell them together as a product kit.

- **Stock**: This module is required if you need to manage the stock levels of your products.
- **Payments**: This group of modules adds new payment methods and integrates with payment gateways:
    - **Authorize.net**: This module integrates your store with the Authorize.net gateway.
    - **Credit card**: This module allows you to accept credit cards. Pay attention, because this method doesn't offer any kind of encryption for security.
    - **CyberSource**: This module integrates your store with the CyberSource payment gateway.
    - **Google Checkout**: This module integrates your store with the Google Checkout payment gateway.
    - **PayPal**: This module integrates your store with the PayPal gateway.
    - **Test Gateway**: This payment method exists only for testing purposes, as it doesn't actually processes payments.
- **Fulfillment**: This group of modules add new shipping methods:
    - **Flatrate**: This module adds a flatrate quoting method to your store.
    - **U.S. Postal Service**: This module integrates your store with the USPS to automatically calculate shipping rates.
    - **UPS**: This module integrates your store with UPS to automatically calculate shipping rates.
    - **Weight Quote**: This module adds a weight quote method to calculate shipping based on the total weight of the order.

You don't have to install all of these Ubercart modules right now. As we proceed, we'll install the required modules for each chapter.

Using UberDrupal, an Ubercart installation profile

When we create an installation profile, we actually give specific instructions to Drupal installer on what extra modules to install, what language to use, what themes to enable, or what settings to use after the installation. It's a very powerful tool with great potential, as it allows developers to create predefined packages of Drupal, created for a specific purpose, for example blogs, image galleries, magazines, or e-shops. The only disadvantage of this procedure is that it's not fully automated, as you still have to first download all the required modules and then run the installation profile.



A few months ago, a new installation profile was created named UberDrupal. It automatically installs all the core Ubercart and required Drupal modules, which we mentioned earlier. It also allows some basic pre-configuration and creates a store with the default settings.

If you want to use this installation profile, you first have to download all the required Drupal modules and all the Ubercart modules, and put them inside the `../sites/all/modules` folder.

Then, download UberDrupal from `http://drupal.org/project/uberdrupal`, and put it inside the `profiles` directory. Now, when you run the installer you'll see two options for **Select an installation profile**: **Drupal** or **UberDrupal**. Selecting the second option enables the UberDrupal installation profile.

The starting steps of this installation are identical with the standard Ubercart installation. We choose our language and we enter the details of our database.

After the installation, we are able to do some basic configuration of our website, right from the installer. The whole process is not perfect yet, but it's very convenient, especially for novice users.

Summary

In this chapter, we talked about the technologies that are required for Drupal and Ubercart to work. We discussed how to create a local server to host your installation, or how to use and customize a package from a professional hosting company.

Then we showed you how to install Drupal and all the required modules for Ubercart. We also examined the Ubercart package and did a brief analysis of the modules contained in it.

Finally, we presented you with UberDrupal, a Drupal profile for easier and faster installation of Ubercart.

3
Basic Configuration

After having installed our Drupal site and all the required modules, now it's time to start configuring the platform as well as our online shop. In the first part we are going to see how to set up Drupal, and in the second part we are going to check all the configuration options in the store administrator section of Ubercart.

The topics we will discuss in this chapter are:

- Drupal basic configuration
- Ubercart configuration

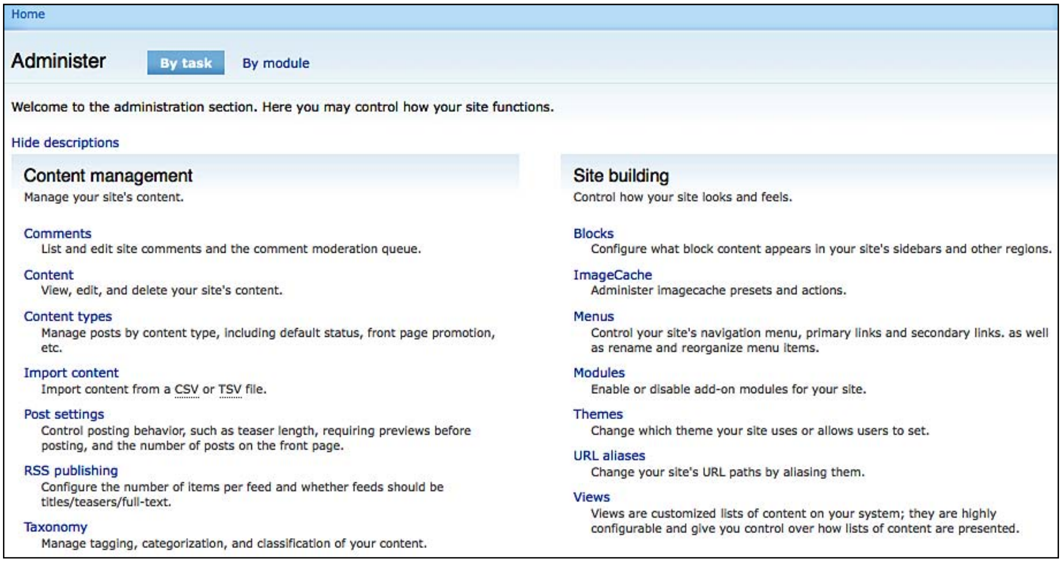
Drupal configuration

It's true that Drupal isn't the easiest CMS system in the world, especially for new users with no programming experience. However, if you invest the required time to get familiar with Drupal's logic and especially the options of the administrator menu, soon you will be a Drupal expert.

General view of the administration page

To reach the main administration page of Drupal, just click on the **Administer** link from the admin menu, located in the left sidebar at the home page of your installation (`http://localhost/Drupal` or the folder name of your installation). Usually, the first time the user reaches this page, he or she realizes that it is full of menus, submenus, options, and descriptions, and thinks how difficult and complex Drupal is. We are sure that you are facing the same issue right now, but don't worry; take a deep breath and be calm. As you follow us during the next steps, you'll realize that this page is structured in a logical order. It has some basic sections, each containing a related group of settings.

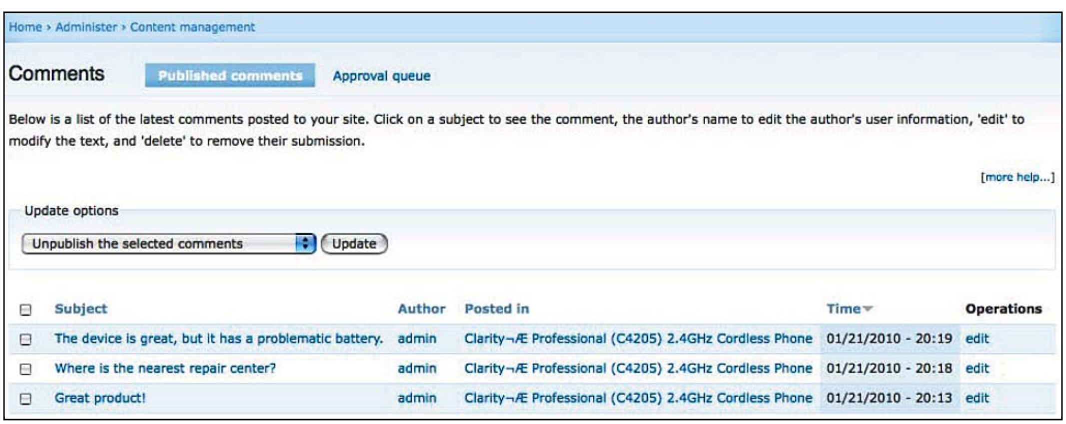
We'll start exploring these sections one by one, explaining all the available options and all the required customizations to you.



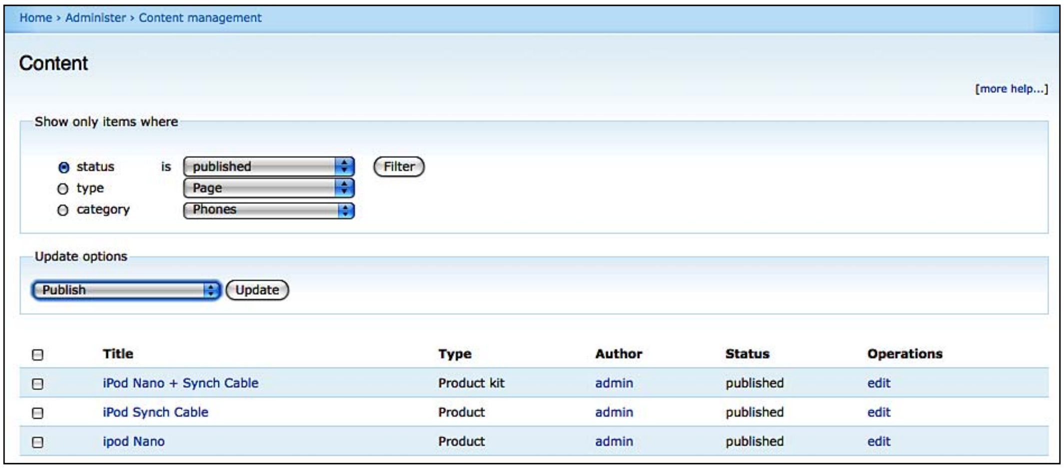
Content management

This menu contains a number of submenus. A description for each submenu is as follows:

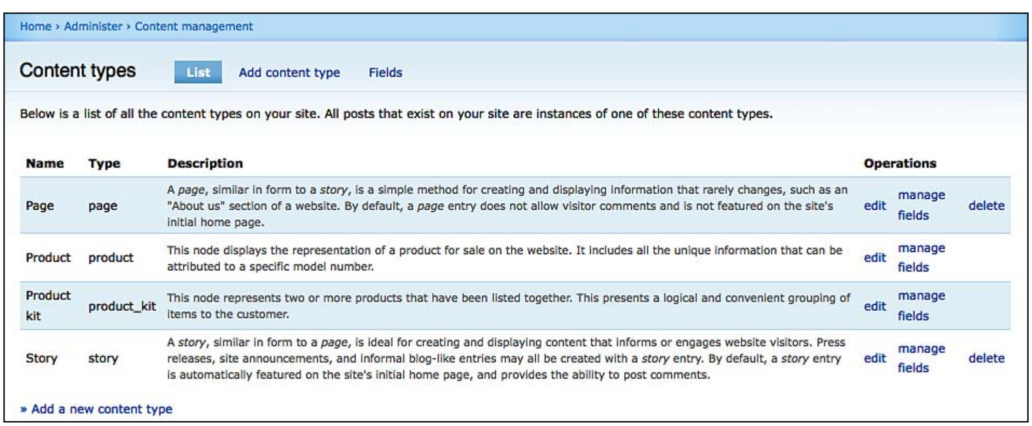
- **Comments**: In this section you can view, publish, unpublish, and delete the comments that users post on your site. You can click on **Administer** from the left menu or the header breadcrumbs to return to the main admin page.



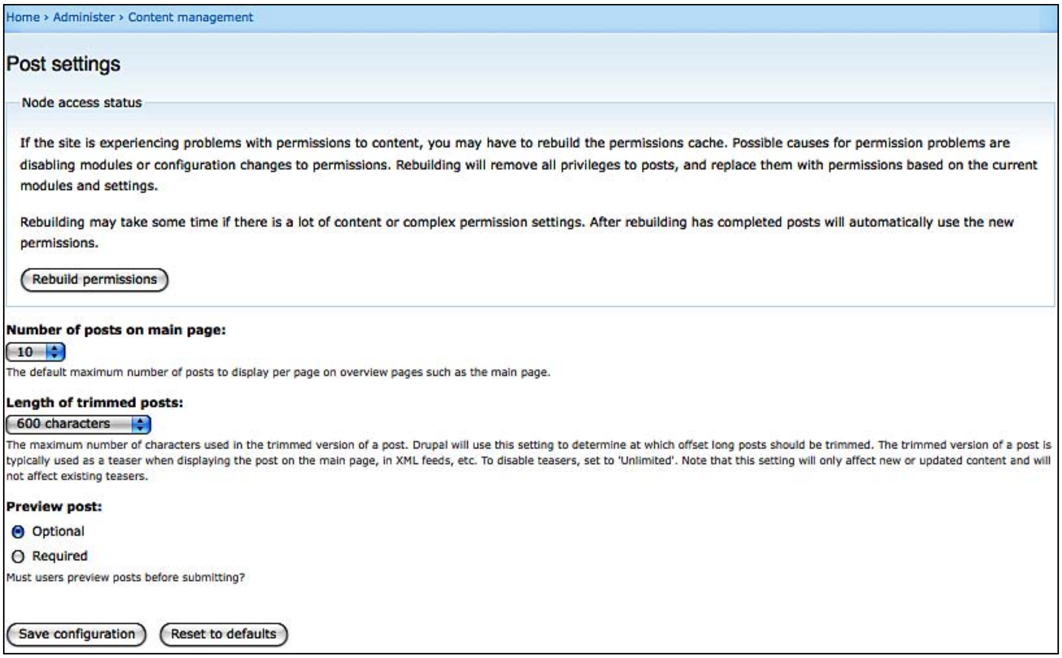
- **Content**: In this section you can manage your content. You can view your content items and filter them by selected criteria, publish or unpublish them, promote them to the first page or to the top of the list, or delete them completely from your site. Keep in mind that you cannot create new content here. If you want to do so, you have to click on **Create content** from the left-hand side menu.



- **Content types**: By default, Drupal creates two content types: **Page** for the creation of static pages and **Story** for the creation of dynamic content. Ubercart adds two more content types: **Product** and **Product kit**. When you install and enable new modules, for example the blog module or the image module, a related content type will be created. If you have specific needs, you can add a new content type by hand. Add the necessary fields and decide how each field will be displayed to the users, thanks to the CCK module that we installed in *Chapter 2, Installation of Drupal and Ubercart*.

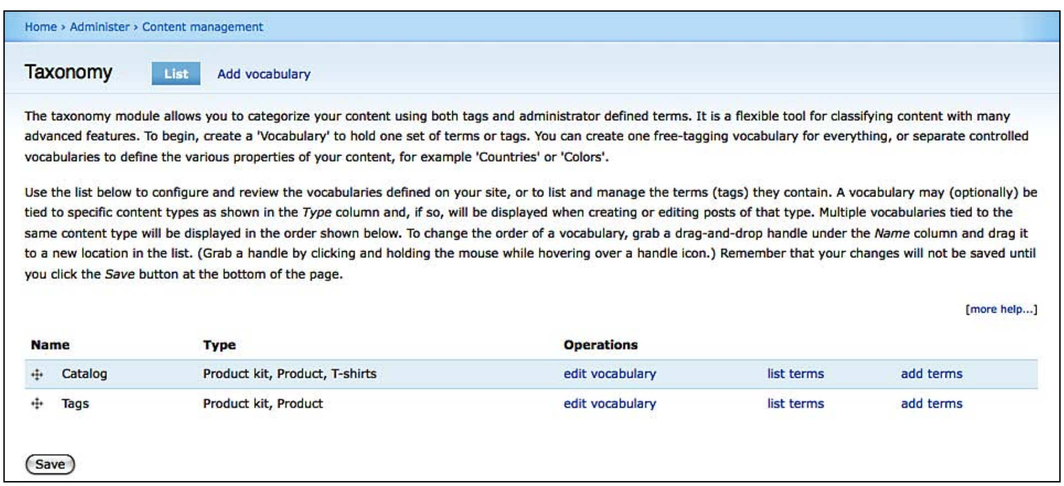


- **Post settings**: This section helps you with the basic configuration of your posts, such as the number of posts on the main page, the length of trimmed posts, and the activation of the preview function when you post a content item.



- **RSS publishing**: By enabling **Rich Site Summary** (**RSS**) publishing you can allow your customers to subscribe to your site and receive regular updates using special software called RSS reader. In this you can set the default number of items to include in each feed and define the content of the feed.

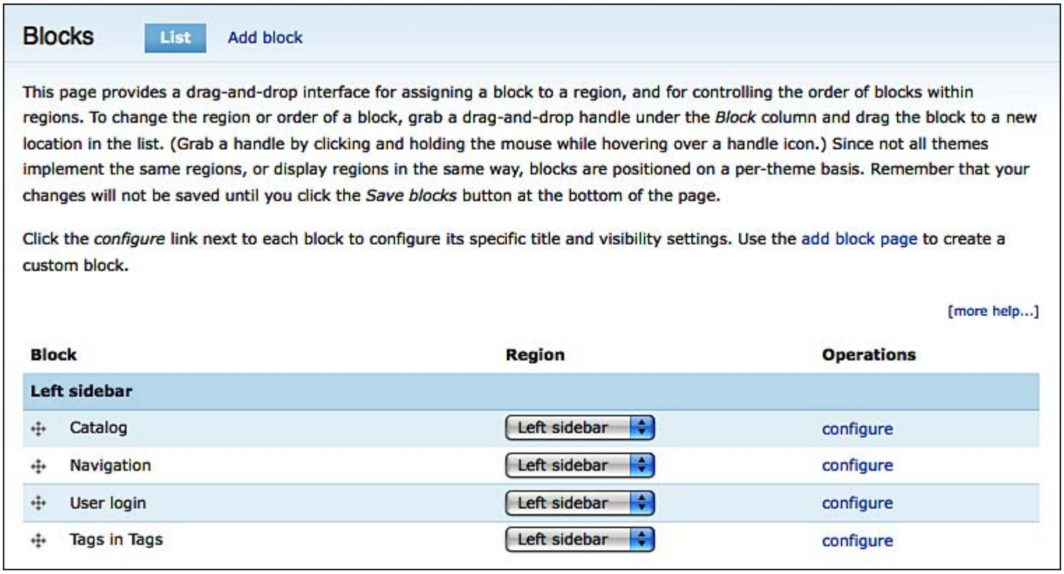
- **Taxonomy**: In this page, you can define the categories and tags for your content and products. We'll take a closer look at Drupal taxonomy in *Chapter 4*, where we'll show you how to better organize your content.



Site building

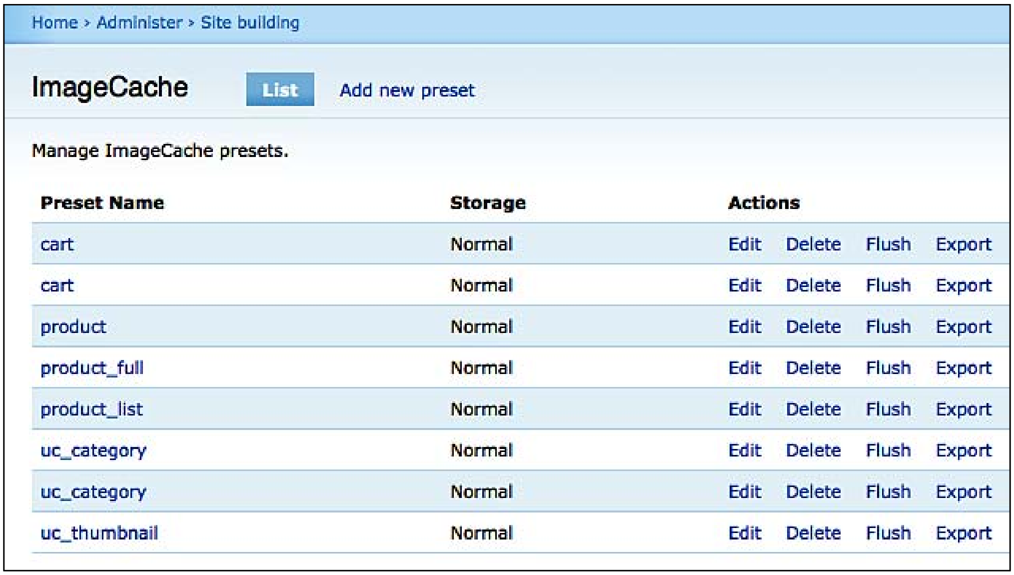
With this section, you can customize your site by using its different functions, such as blocks, menus, and so on:

- **Blocks**: Blocks are units of code, used to display content in specific regions of your site. Usually, they are created either by the installed modules or when you add a new menu, but the site administrator can also create a block from scratch. In the **Blocks** page, you can add a new block, configure a block, assign a block to a region, or delete a block. We'll take a closer look at blocks in *Chapter 8, Customizing the Frontend*, where we'll show you how to change your Drupal theme.

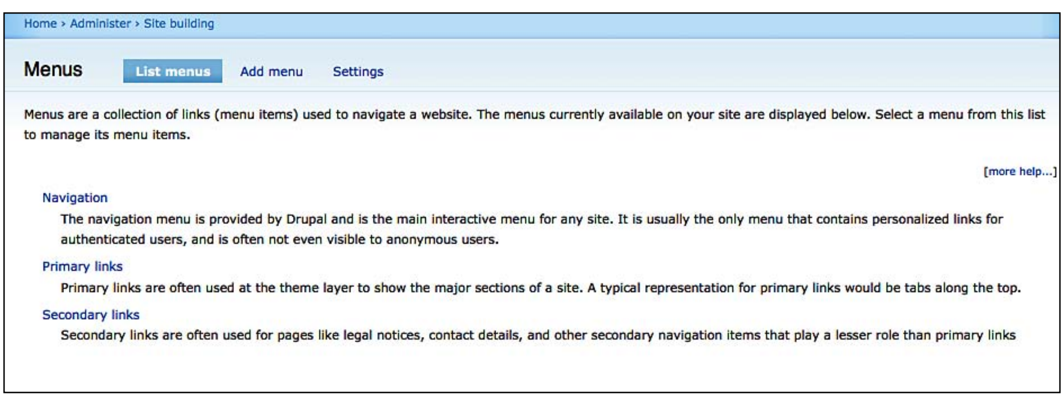


- **ImageCache**: This page helps you create image presets for specific pages of your site. When you install Ubercart, it automatically creates image presets for the main pages of the site such as cart, product list, product page, and categories.

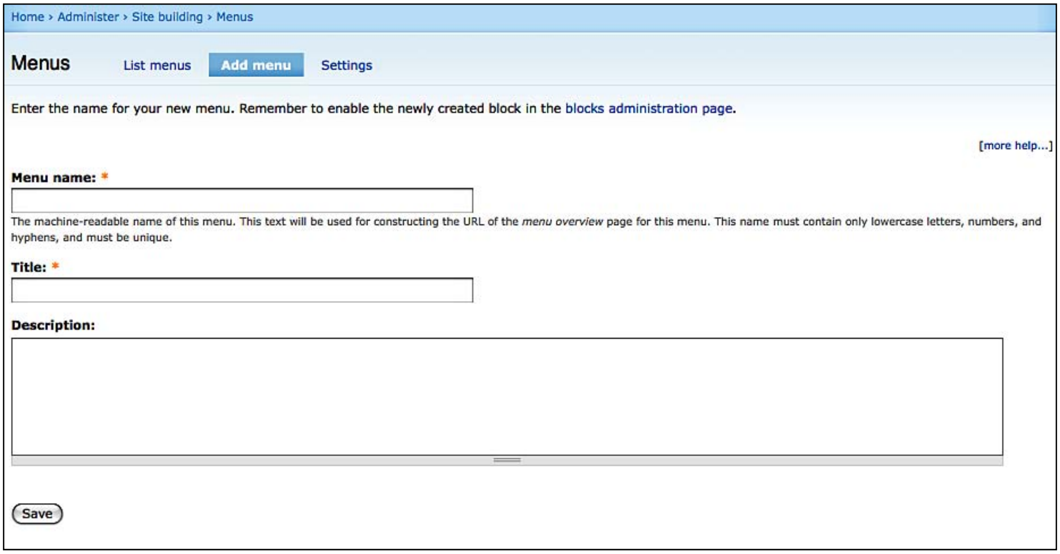
If you are not satisfied with the default dimensions of images, you can edit them and assign new properties to them. You can manipulate the original images using six actions—crop, scale, desaturate, resize, rotate, and sharpen. You can also add a new preset or delete one of the existing.



- **Menus**: This page allows you to control the menus of your site. The three default menus created by Drupal are **Navigation**, **Primary links**, and **Secondary links**. The Navigation menu is located on the left of your site. It contains useful links for the administrator. The use of Primary links is to provide links to the important pages to the visitors of the site. On the other hand, the Secondary links menu is useful for creating navigation items linking to less important sections of your site.



If these menus are not enough, you can create a new menu by clicking on the **Add menu** button. Enter a name for the menu, a title, and a description that is visible to site administrators.

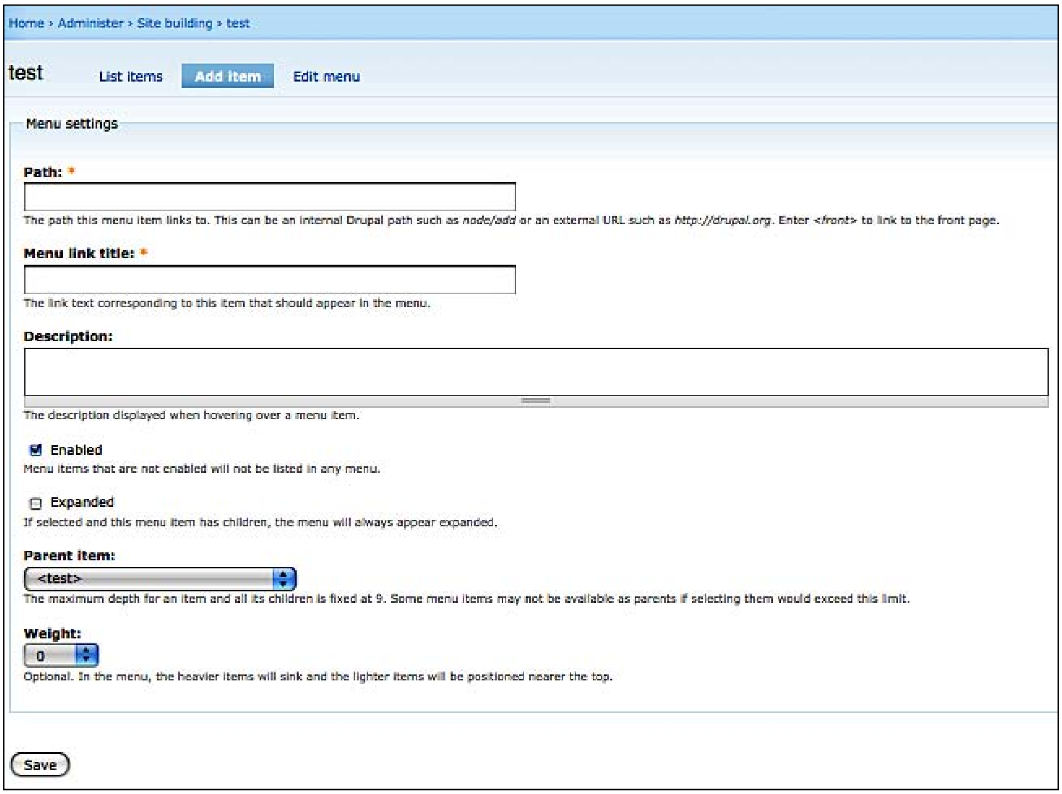


If we click on a menu, we can view the existing items and edit them. If we click on **Add item**, we can create a new menu item by completing the following fields:

- **Path**: This is the URL of the page that you want to be linked by this menu item. You can use an internal Drupal path, an external URL, or <front> for the front page of your site. Unfortunately, you cannot search for a page path right from this field, so you have to remember it from your browser's address bar.
- **Menu link title**: This is the title of the menu item, which is visible to your visitors.
- **Description**: You can add a description that is visible to your visitors when they hover the cursor over the menu link title.
- **Enabled**: When you create a menu item, it needs to be enabled in order to be visible in the menu.
- **Expanded**: Checking this option, when a menu has children your menu will always appear expanded.

    - **Parent item**: In this field, you can select the parent menu of the item.
    - **Weight**: Finally, you choose a weight to select the position of this item in the menu list. Remember that a heavier item will be placed lower than a lighter one, so apply the lowest number to your first item.

After completing all the fields, click on **Save** to store this specific item.



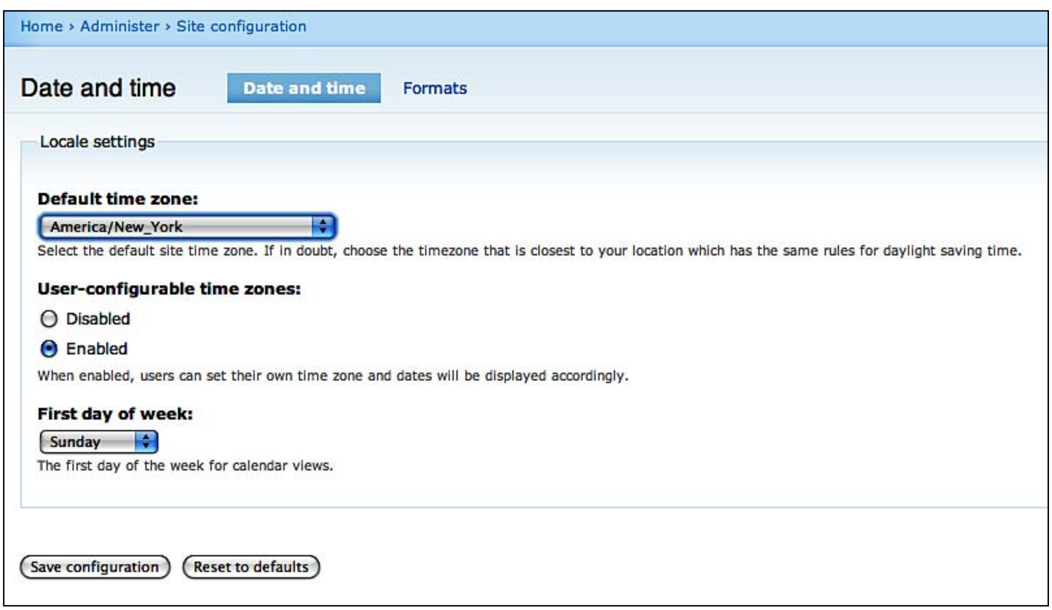
- **Modules**: In *Chapter 2*, we examined what modules are, and we installed all the required modules for Ubercart. Following the same procedure, you can add new functionalities to your website, by uploading new modules in your `site/modules` directory and enabling them in this page.
- **Themes**: Drupal allows you to change the look and feel of your site. In the themes page, you can see the pre-installed themes and enable them or configure them. We have an entire chapter devoted to themes: *Chapter 8*.

- **URL aliases**: When you create a new content item (also known as a Node), Drupal creates the path for it using the format `path/x`, where `x` is an auto number starting from 1 and increasing by 1 for every new item. Adding a new alias allows you to show more readable and easy-to-remember URLs.
- **Views**: This gives you the opportunity to create custom lists of content, filtered and sorted by selected criteria. In *Chapter 9, User Interface Enhancement Techniques*, we'll see how we can enhance the Ubercart default catalog by creating custom functionality.

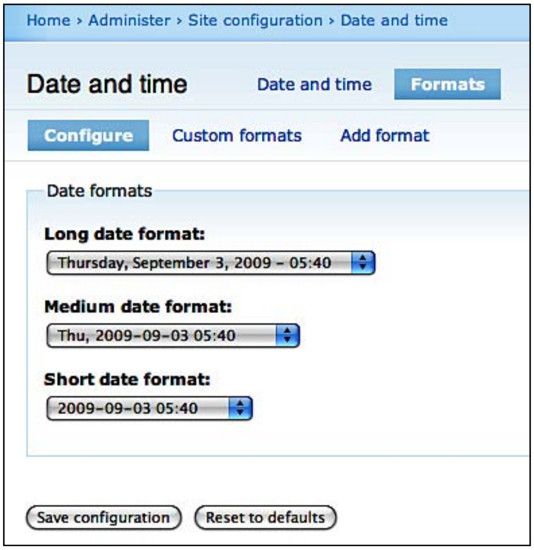
Site configuration

In this section, there are many options, allowing you to customize every detail of your store. Most of them are created by Drupal installation but many modules add new options to site configuration page. There's no need to explore all of them right now, so we'll show you how to customize only the critical ones.

- **Date and time**: In this page, you can make some basic adjustments about how Drupal handles date and time. You can select the default time zone, depending on where your store is located. You can also allow your clients to enable their own time zones and use the time of their region. You can define the first day of the week for your calendars. After inserting all the required information, click on **Save configuration** to store them.

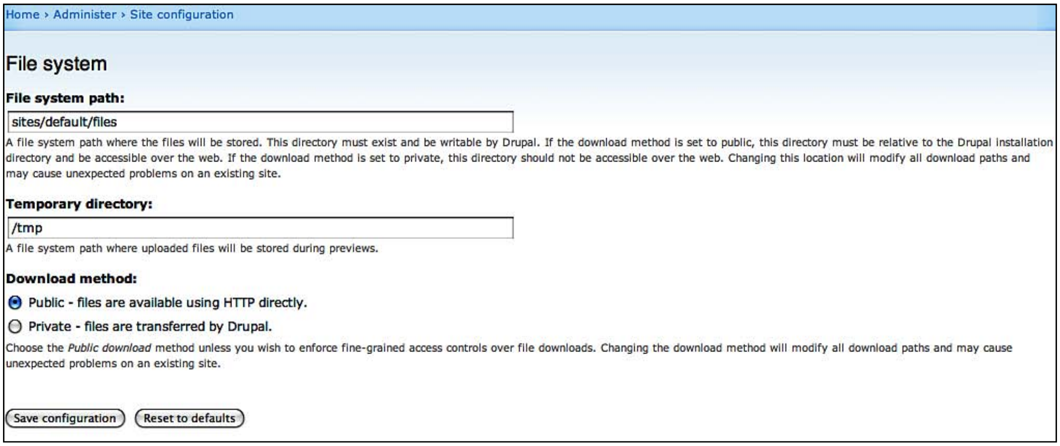


- **Formats**: This tab defines how exactly date and time will be displayed. Different countries use different formats and you also may have some personal preference. Select the date format for long date, medium date, and short date and click on **Save configuration**. If you are not satisfied with the date formats that Drupal proposes, you can click on **Add format** and create a custom format using the parameters for the PHP date function.

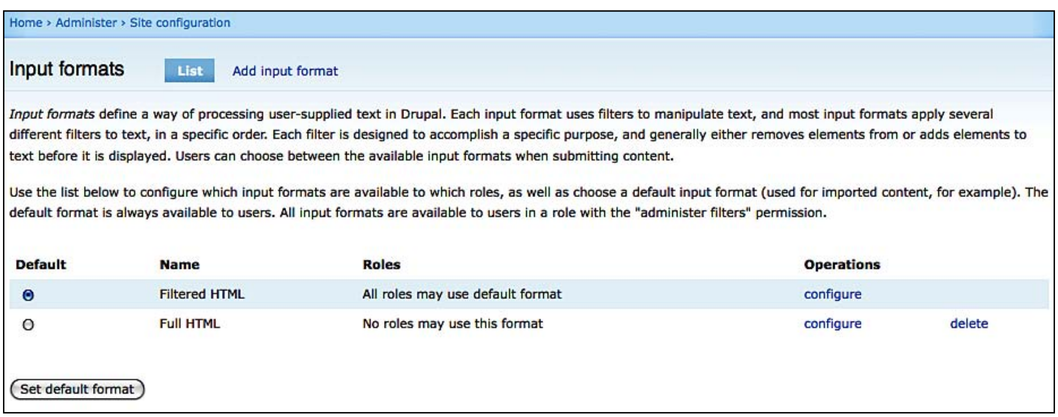


- **File system**: In this page, there are three main options about how Drupal handles the files that we upload:
    - **File system path**: In this field, insert the path where the files will be stored. If you want to use a path different from the default, you have to first create it in your server and then declare it here.
    - **Temporary directory**: This is used when you preview the files that you want to upload. You don't have to change this path.
    - **Download method**: Finally, you can select between the private and public file download methods. You are not going to store any sensitive information in this folder, but only product images, so select the public method.

When you finish, click on **Save configuration**.

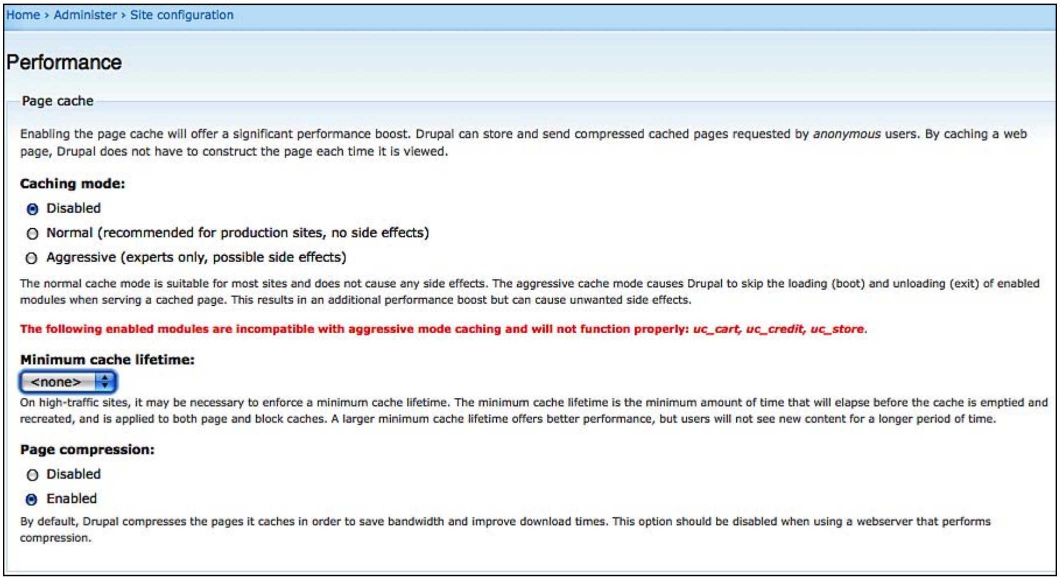


- **Input formats**: This allows you, as the site administrator, to create filters and restrictions for user-created text. By default there are two input formats: **Filtered HTML**, which uses a restricted set of HTML, and **Full HTML**, which allows any HTML tag to be inserted. You can configure these input formats or you can create new ones. Some modules create new input filters allowing you, for example, to input flash movies or videos. Filtered HTML is a safer option for your store, so there is no need to change anything in this page.

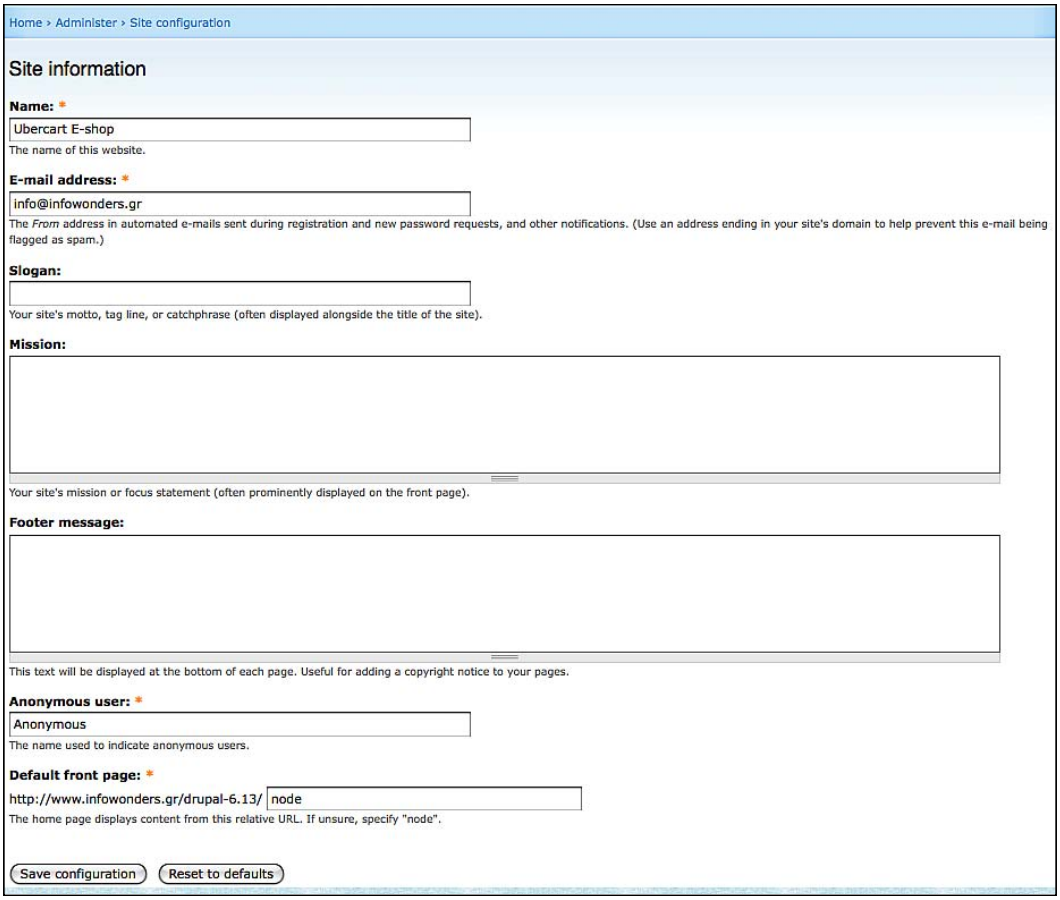


- **Performance**: The main purpose of this page is to allow to you to enable some very important optimization features, thus making the store much faster. These features are:
    - **Page cache**: Drupal can store a compressed copy of the page in the database and send it to anonymous users. Using this method, it creates the page only once and it serves it again and again during the cache lifetime. It is recommended to enable the **Normal** caching mode, because the **Aggressive** caching mode is not compatible with some basic features of Ubercart.
    - **Minimum cache lifetime**: If you are not renewing the content of your store very often, you can increase the minimum cache lifetime to 6 or 12 hours, or even a day. Keep in mind though, that anonymous users will be able to see the new content only after a longer period of time, as you increase this setting.
    - **Page compression**: This feature decreases the size of the pages and makes downloads faster, so you have to keep it enabled.
    - **Block cache**: This feature enables caching, but for blocks instead of pages. If page caching is not enough, you can further optimize your site by enabling it.
    - **Bandwidth optimizations**: Bandwidth optimizations are also very useful, because they allow you to compress **CSS files** and **JavaScript files**. You should enable them just right before you launch your site, because they can cause problems during theme and module development.
    - **Clear cached data**: This button allows you to clear and rebuild cache data. Usually, during theming or module development, caching doesn't allow you to see the recent changes that you made to the website, so you have to come here and clear it.

Finally, click on **Save configuration** to store your settings.



- **Site information**: This page allows you to set the basic information of your store. Insert the name of the shop, the main e-mail address, the slogan, mission statement of your site, and a message to be displayed at the footer of your store's pages. Set how you want to name your anonymous users and define the address of the home page.



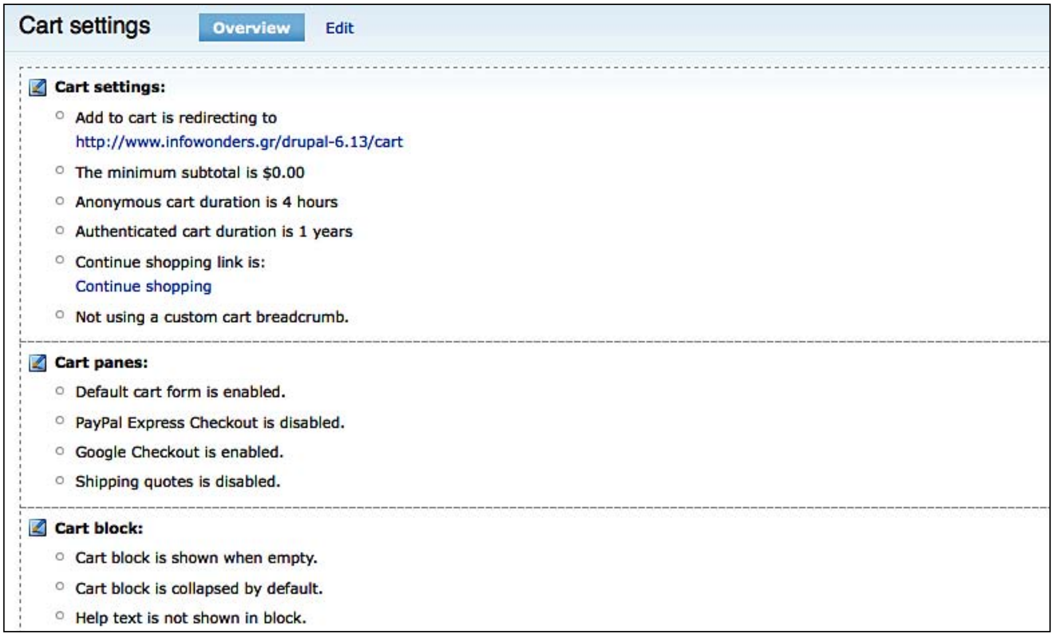
When you're finished, click on **Save configuration**

Store administration

During the next chapters, when we show you how to build your store and customize it for your needs, we'll spend more time on the store administrator area. For now, we need to work only in the **configuration** section. This section contains various settings, such as cart settings, catalog settings, and so on.

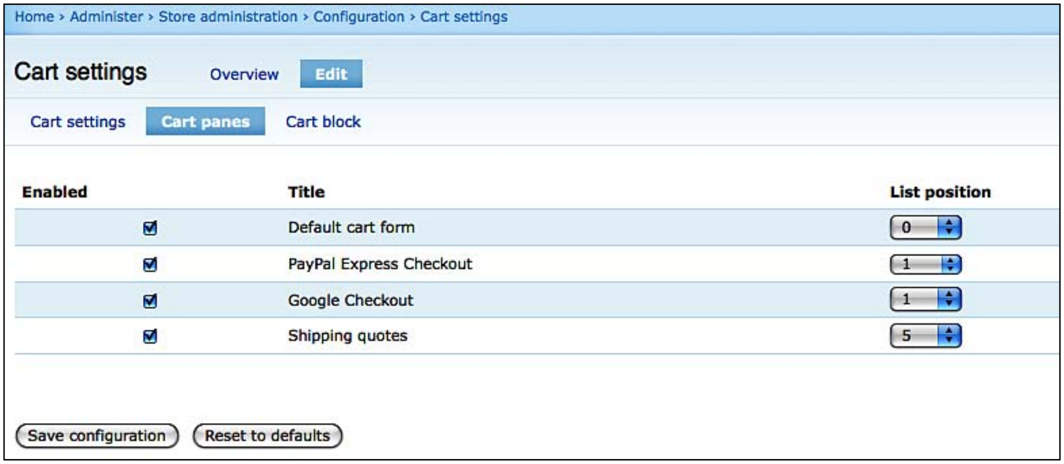
Cart settings

The **Cart settings** page has three settings groups: **Cart settings**, **Cart panes**, and **Cart block**. Click on **Edit** to go to the cart settings page:



- In the **Add to cart redirect** field, enter the name of the page to which the customer will be redirected when he adds a product to his cart. If you leave this field empty, there will be no redirection.
- If you want to set a minimum order subtotal to proceed to checkout, enter this value to the corresponding field.
- The next two options allow you to set the duration for the cart of anonymous users and authenticated users. Actually, this is a matter of personal preference, so select the **Duration** and the **Unit of time** for both of them.
- When the user adds a product to the cart, he or she will be redirected to the cart page. The **Continue shopping element display** option gives you the opportunity to select if there will be a continue shopping link, and whether it is just a text link or a button.
- The next four fields, **Default continue shopping link URL**, **Custom continue shopping link text**, **Custom cart breadcrumb text**, and **Custom cart breadcrumb URL** are used to create custom functionality for our cart. You don't need to change anything here, so leave these fields blank.
- Click on **Save configuration**.

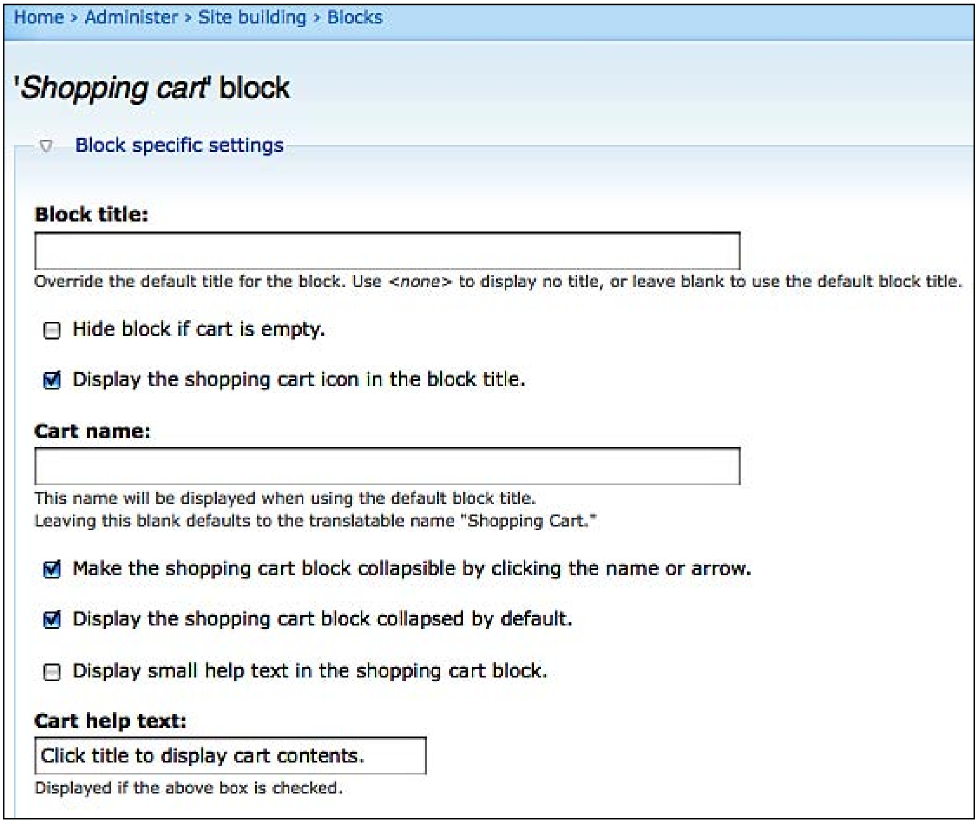
Now select the second tab, **Cart panes**. Here, you can select the individual panes that create the cart view page. You need at least the **Default cart form**, but for your store let's enable all four available panes, which are related to shipping and payments. Click on **Save configuration**.



Select the third tab, **Cart block**. Click on **Click here** to go to the shopping cart block configuration page. The cart block is usually located on the left sidebar and shows an overview of the order.

- The default name for this block is **Shopping cart**, but you can override it by inserting a title in the **Block title** field.
- In the next group of options you can define whether you want to hide the cart block if the shopping cart is empty or you want to display the shopping cart icon in the block title. You can also decide whether the shopping cart is going to be collapsible or is going to be displayed collapsed by default. You can also show a small help text in the shopping cart block.
- In the **Cart help text** field you can change the default help message.
- In the final section, you can control the visibility of this block. You can show it or hide it (by default), show it only to **specific roles**, or only on **specific pages**.

- We don't really have to change anything here, so just click on **Save block** to continue.



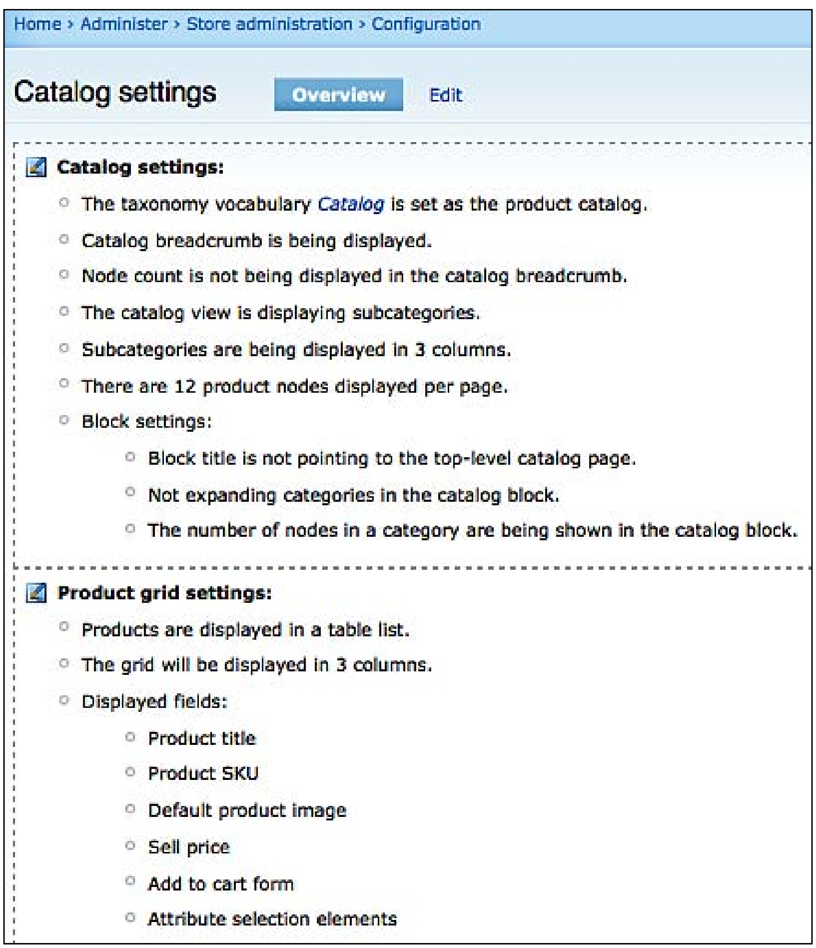
- Return to the main store configuration page and select **Catalog settings**. We will read about these settings in more detail in the next section.

Catalog settings

With catalog settings you can control how the product catalog will be displayed to your customers. It has two tabs: **Catalog settings** and **Product grid settings**. Click on **Edit** to customize it.

- The **Catalog settings** page has three option groups, allowing you to customize different elements of the catalog vocabulary.

- If you want to use the custom catalog vocabulary, you can select it from the catalog vocabulary drop-down menu in the **Catalog top level** options group. In the same group you can select whether to display the catalog breadcrumb, to display the number of products in the catalog breadcrumb, or to display subcategories in the catalog view. You can also select the number of columns in the grid of categories.
- In the next field you can select how many products will be shown on every catalog category page.
- Finally, in the **Catalog block settings** option group you can select whether to make the block title a link to the top-level catalog page, categories to be expanded in the catalog block, or whether to display the number of products in the catalog block.
- Click on **Save configuration**.

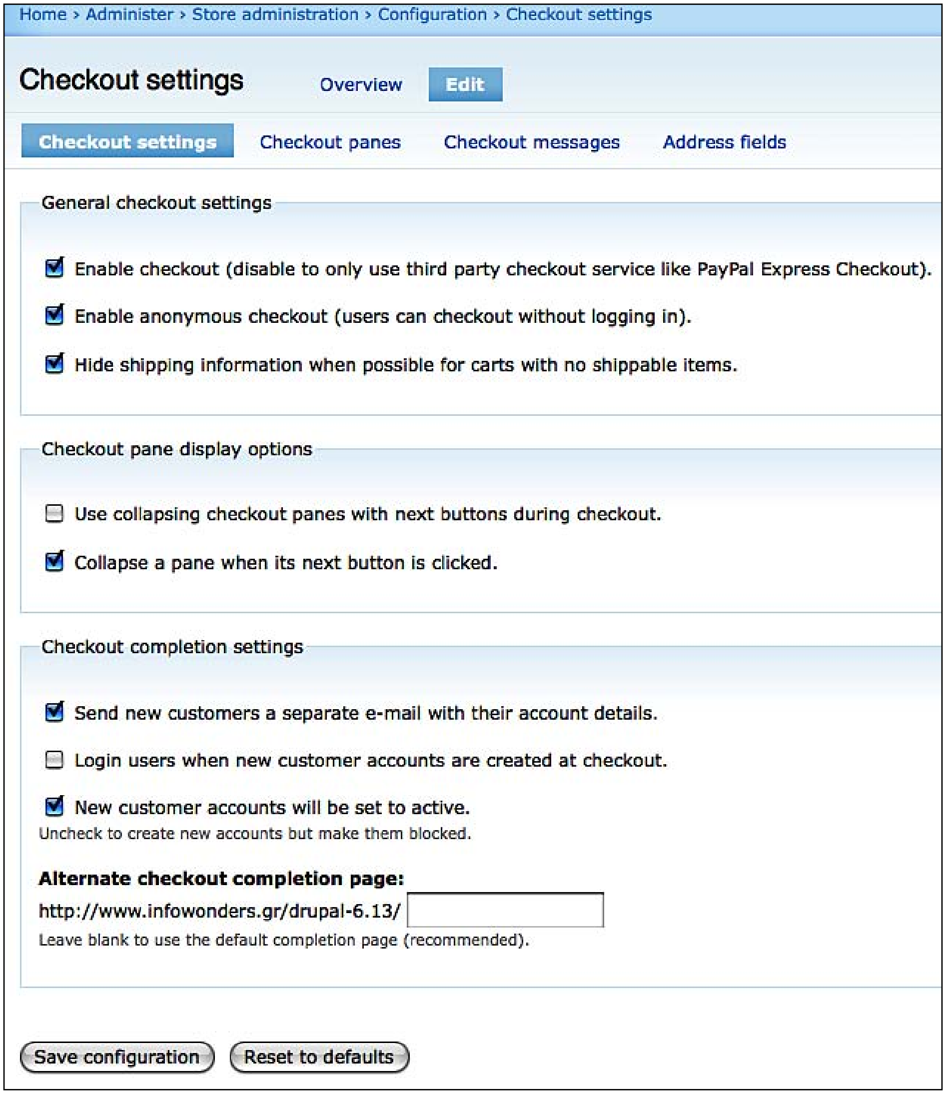


- Now click on **Product grid settings** tab. With these settings we can further customize our product catalog.
- In the product grid settings, first you have to select whether you need to display the products of the catalog in a grid. Select the number of columns in the grid and the fields that will be displayed in your product catalog and click on **Save configuration**.

Checkout settings

Using the **Checkout settings** you can change the look and feel of the checkout page. It has four tabs. Click on **Edit** to enable them.

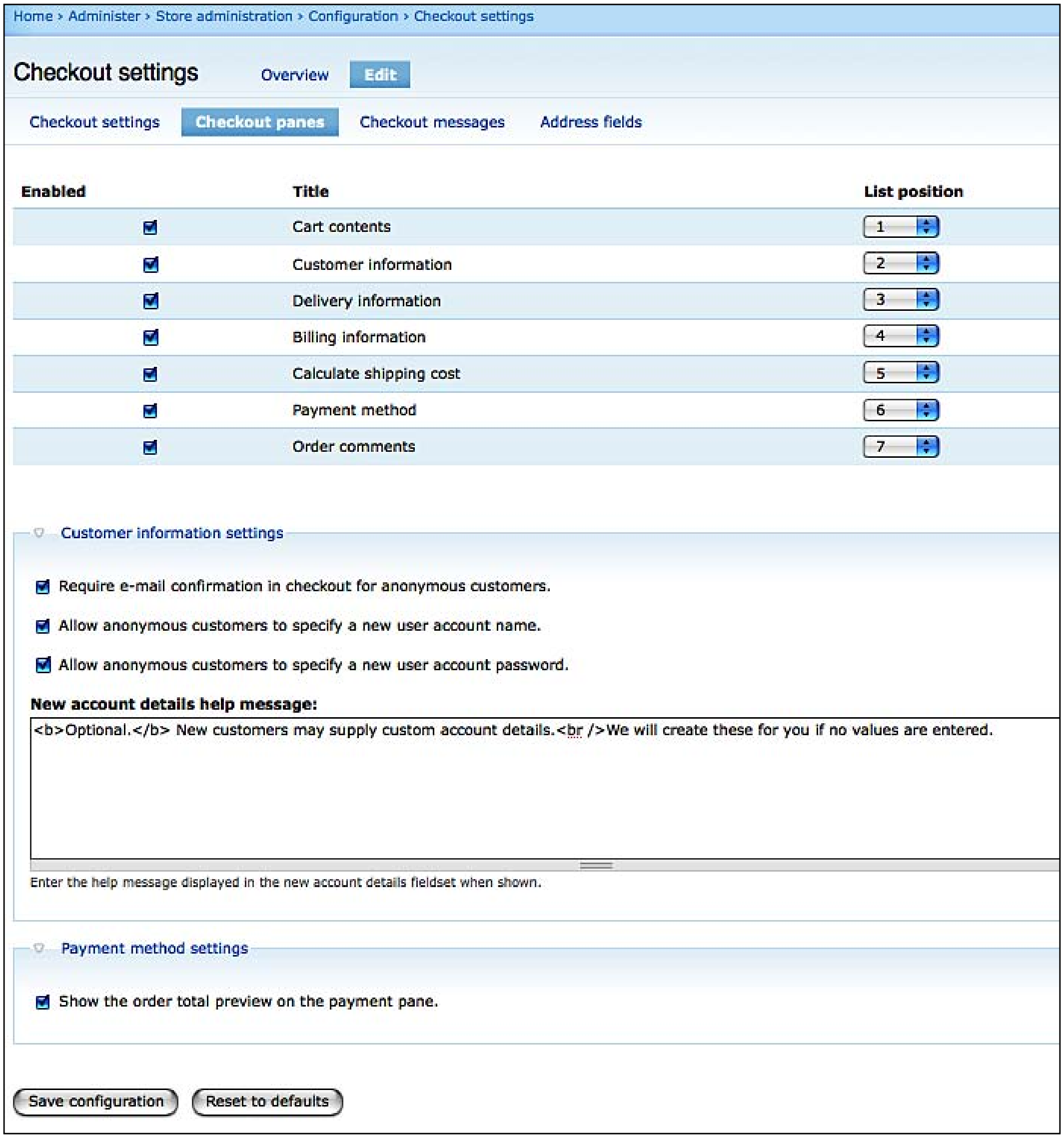
- In the **Checkout settings** page, there are several checkboxes that let you enable or disable basic features of the checkout page. If you are using a third-party checkout service, you can disable the checkout page provided by Ubercart; for this you need to uncheck the first checkbox. We are not going to use such service, so leave it checked.
- The second checkbox enables or disables anonymous checkout. We don't want to allow users to check out who haven't logged in, so uncheck it.
- The next checkbox hides the shipping information panes, when the order has only non-shippable options. It's a very useful feature, so leave this checked.
- In **Checkout pane display options**, you can select whether to use collapsible checkout panes or to collapse a pane when its **Next** button is clicked. The default settings are just what we need, so proceed to the next checkboxes.
- The last group of checkboxes lets you decide how your store is going to handle a new customer account. First, you have to decide whether you want to send new customers a separate e-mail with their account details. It's very useful to inform your customers about their accounts, so leave this checked.
- If you check the next checkbox, Drupal will automatically log in users when new customer accounts are created at checkout.
- The last checkbox lets you set new customer accounts to active. It is recommended to leave this checked, because if you don't, you'll have to activate your clients account manually on a daily basis.
- In the last field you can specify an alternative checkout completion page. Leave this blank, because we are using the default Ubercart checkout page.
- Click on **Save configuration** to store your settings.



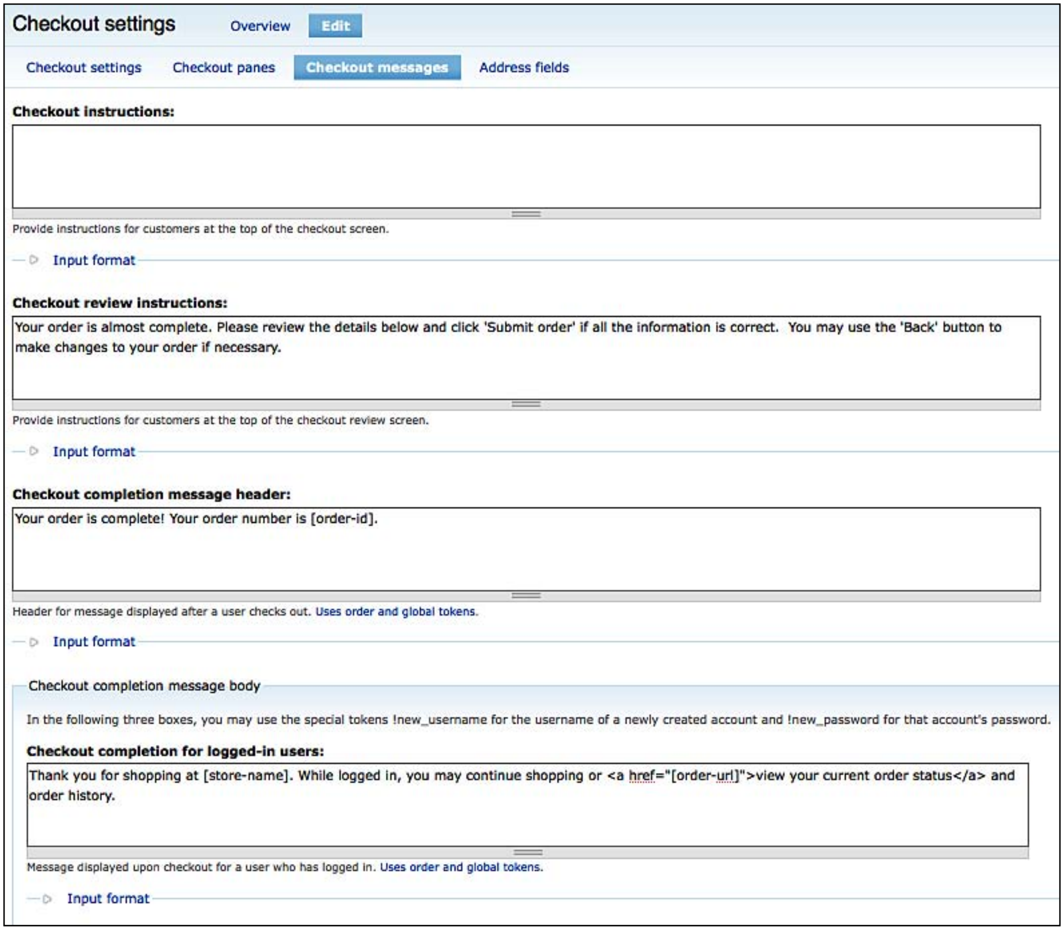
Now click on the **Checkout panes** tab. It has three option groups:

- In the first group you can enable or disable the different panes of the checkout page. All the panes are useful, so leave all of them checked.
- In the **Customer information settings** group, you can decide how you're going to handle the creation of user accounts by anonymous customers. If you want an e-mail confirmation from anonymous customers, and you want to allow them to set their own user names and passwords, you have to check the next three checkboxes.

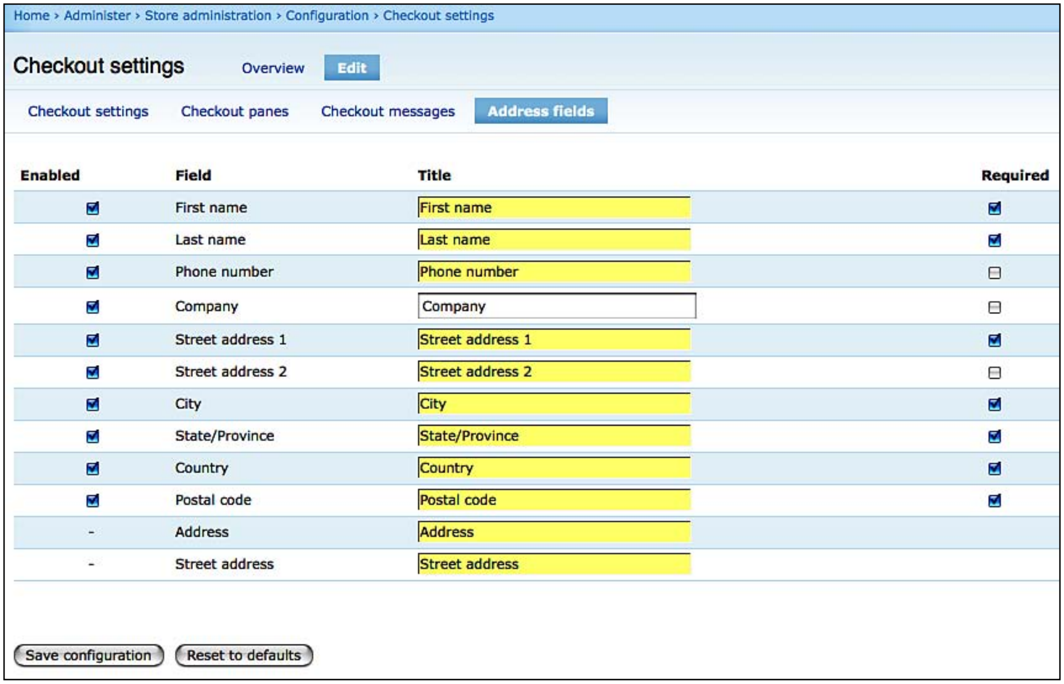
- If you don't want to change the default help message, and want to show the total order preview on the payment pane, just click on **Save configuration**.



Next is the **Checkout messages** tab. This page enables you to view and change the several instructions and messages shown to the customers during checkout. We are going to keep the default settings in this page.



Now click on the last tab to configure the customer address fields. In this you can enable or disable them, change their title, and decide which fields are required.



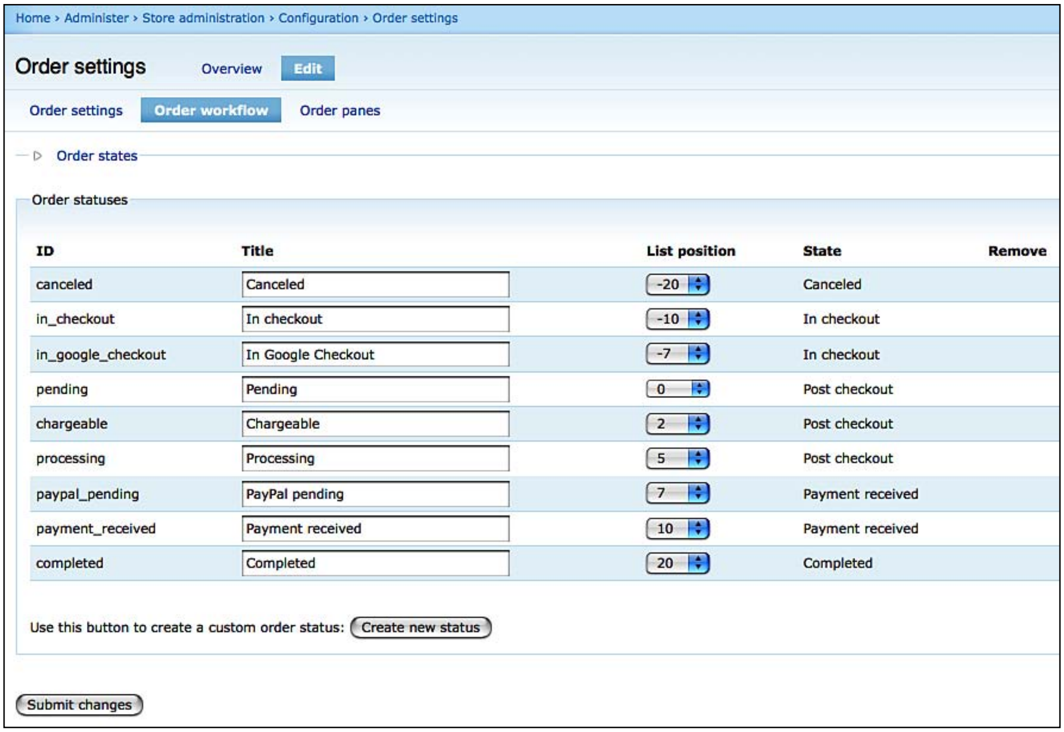
We are going to keep the default settings in this page also.

Order settings

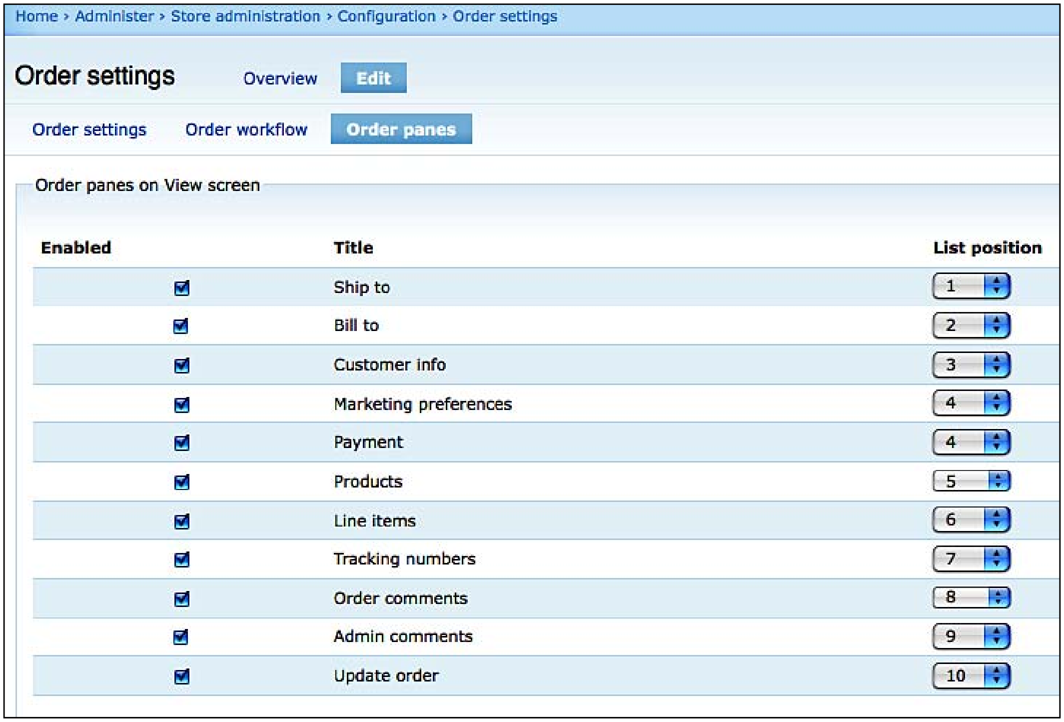
The **Order settings** page has three tabs. Click on **Edit** to enable them.

- In the **Admin settings** section you can set how many orders are shown on the **Overview** screen in the admin page, enable order logging, and capitalize the address on order screens.
- In the **Customer settings** section you can decide whether to allow your customers to view order invoices from their order history and which invoice template to use.

- Now click on the **Order workflow** tab. Here you can see all the **Order statuses**, rename them, change their position in the workflow, or create new ones. The available order statuses are very well planned so, at least for now, there is nothing to change here.



- Now click on the last tab—**Order panes**. In this page you can check the order panes on the view screen, on the edit screen, on the invoice screen, and on the customer screen. All the panes are enabled by default and we are going to keep it this way.

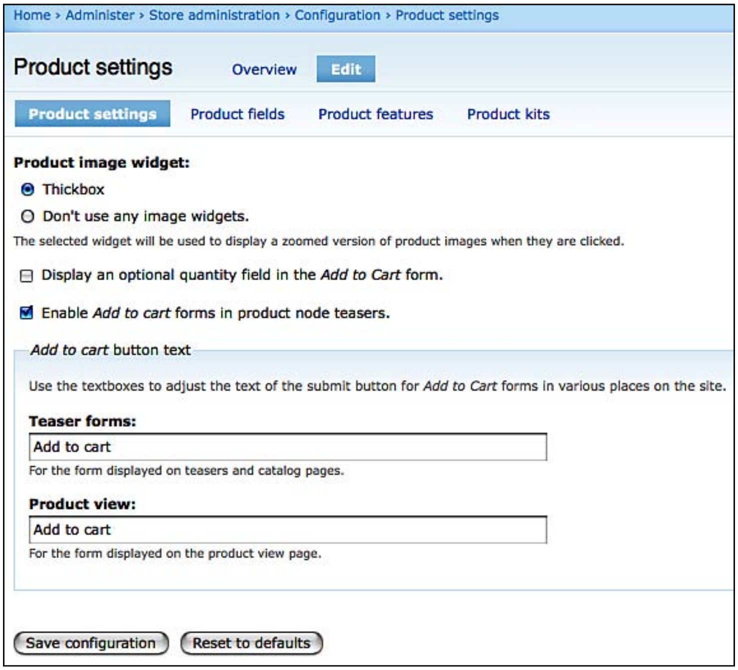


Product settings

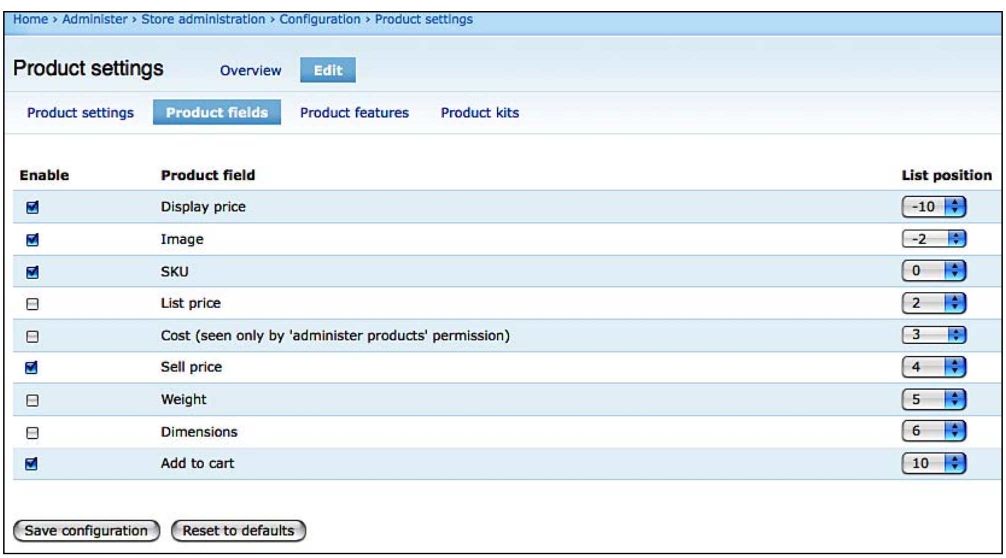
The **Product settings** page has four tabs. Click on **Edit** to enable them.

- In the **Product settings** page, the first option lets you select which product image widget we are going to use. The default widget is **Thickbox**, which allows the customer to see the product images in larger size by clicking on them. It's a very convenient feature, so leave this selected.
- By checking the **Display an optional quantity field in the Add to Cart form** checkbox, you let the customer to enter a quantity value next to the **Add to cart** button, thus adding more than one item of the same SKU instantly.
- The next checkbox, **Enable Add to cart forms in product node teasers**, creates an **Add to cart** button not only on the product page, but also in the category list or in custom lists that you will create in the future.

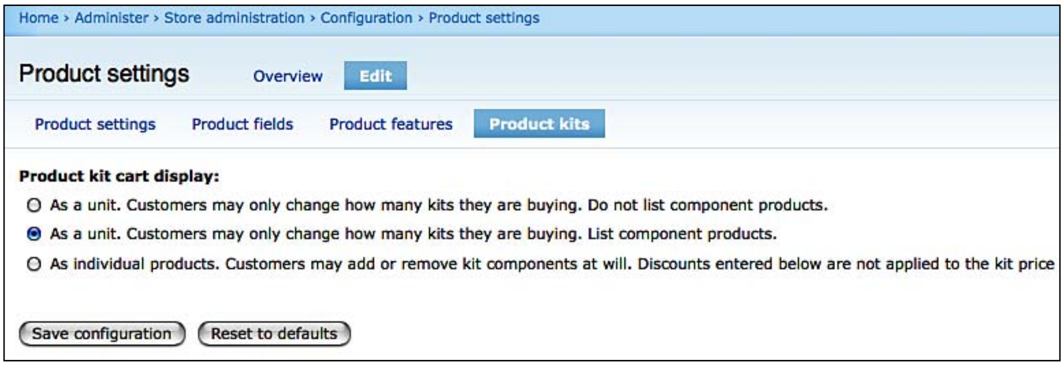
- Finally, you can change the text shown in the **Teaser forms** and **Product view**.



- Next is the **Product fields** tab. In this tab, you can enable, disable, or change the position of several fields that create the product page.



- The last tab is the **Product kits** tab; in this page you decide exactly how product kits are displayed in the cart. There are three options: As a unit without listing the individual products, as a unit showing the individual products, or as individual products. Usually, the second choice is the most convenient.



Store settings

The **Store settings** page has three tabs. Click on **Edit** to enable them.

- In the **Contact settings** page, you enter the contact information of your store.
- In the **Display settings** page, you select the display type for the main store administration page, primary customer address, and a footer message for store pages.
- In the **Format settings** page you have to choose a **Currency format**, a **Weight format**, a **Length format**, and a **Date format**.

Summary

In this chapter we examined together, step by step, the main configuration options of Drupal and Ubercart. We analyzed the main administration page of Drupal and its sections such as content management, site building, and site configuration. Then we moved to the store administration pages, where we showed how to customize every detail of your shop functions to your personal needs. Now that the boring work is over, we can proceed and start creating our categories and products in *Chapter 4, Managing Categories, Products, and Attributes*.

4

Managing Categories, Products, and Attributes

In this chapter the action begins! We are going to see how to manage categories using the Drupal taxonomy, how to create new products or import them from external files, and how to use the attribute system of Ubercart. The topics that we will discuss in this chapter are:

- Creating categories and subcategories
- Creating products
- Importing products from text files
- Creating product attributes
- Managing the product catalog

Creating categories and subcategories

Before we start working on our Ubercart site, it's better to spend some time planning. It's completely understandable that you just can't wait to create your products, complete your store, and start selling. However, don't underestimate the importance of proper planning.

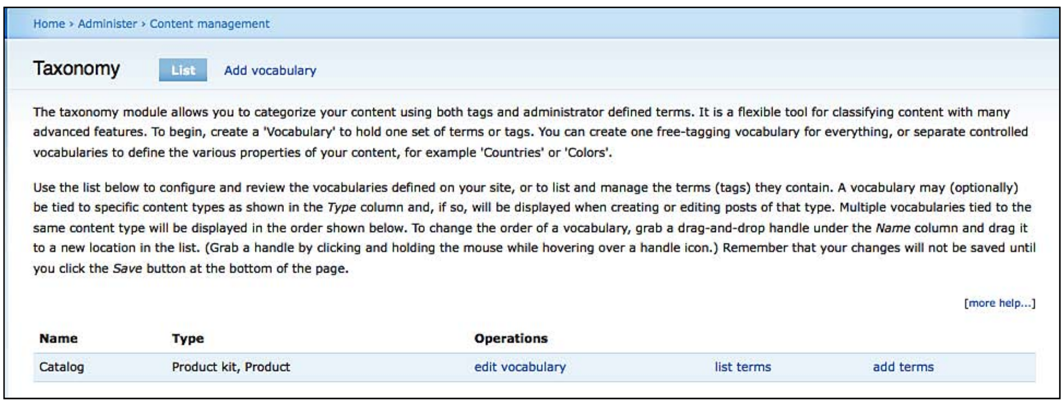
For a small store with a few products, organizing the content in categories and subcategories is an easy and straightforward procedure. On the other hand, for a big store with many sectors and thousands of products, it could be a time-consuming task.

How many times have you found yourself searching for a specific item in a big e-shop among complex and meaningless categories? User experience has to be our first priority, so we not only have to give meaningful names to our categories, but also have to build a simple and logical structure. We don't want to have many main categories. However, we have to keep in mind that using too many subcategory levels may be confusing for our visitors.

Configuring product categories

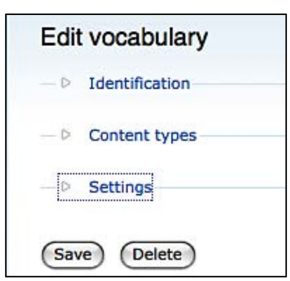
So, let's start and create the basic categories and subcategories of our store. We've already written the basic structure on a piece of paper, so it will be easy to replicate this structure in Ubercart.

Go to **Home | Administer | Content management | Taxonomy**.



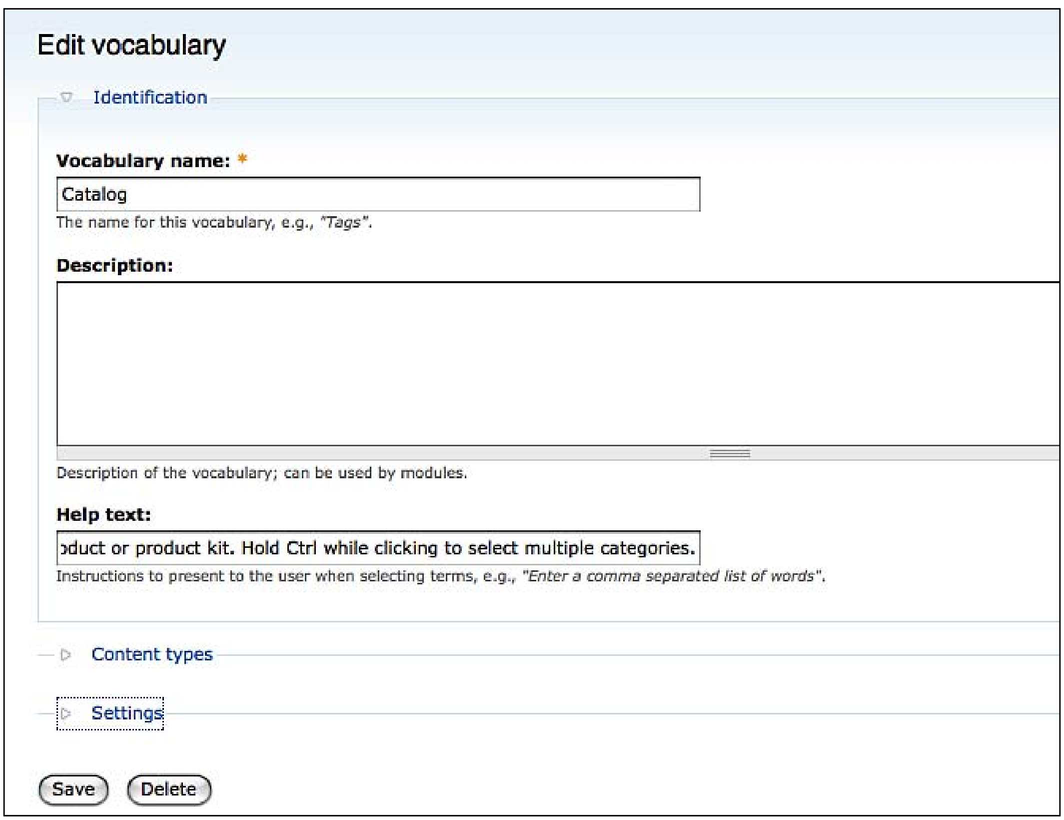
As you can see, there is only one vocabulary available, named **Catalog**. This was created by Ubercart, when we enabled the catalog module and it's used by products and product kits.

Let's take a look at the basic properties of catalog. Click on **edit vocabulary**. There are three main groups of properties: **Identification**, **Content types**, and **Settings**.

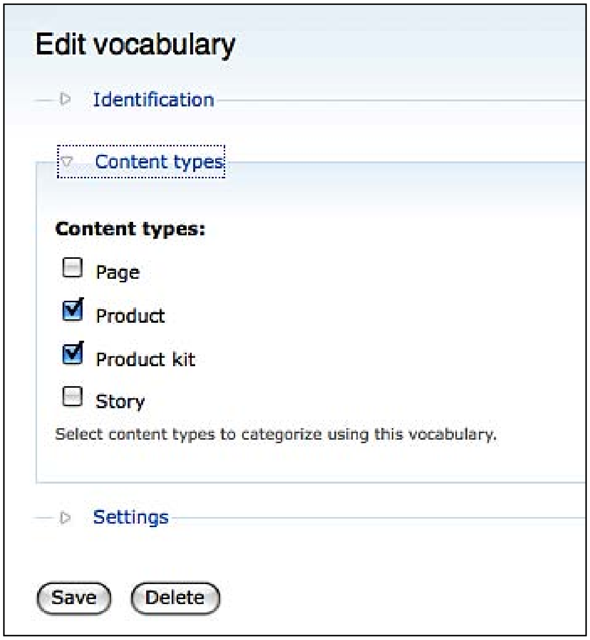


Here is the explanation for these properties:

- **Identification**: This group has three fields:
    - **Vocabulary name**: There is no need to change the name of the vocabulary, so leave this as it is.
    - **Description**: A description for this vocabulary is useful if there are other administrators, so write inside this field: "Catalog is the vocabulary created by Ubercart. It is used to organize products and product kits into categories and subcategories."
    - **Help text**: The help text is available to the users of the website, when they create or edit products. It's shown right beneath the Catalog field. Change the text to "Select a category for your product or product kit. Hold Ctrl while clicking to select multiple categories."

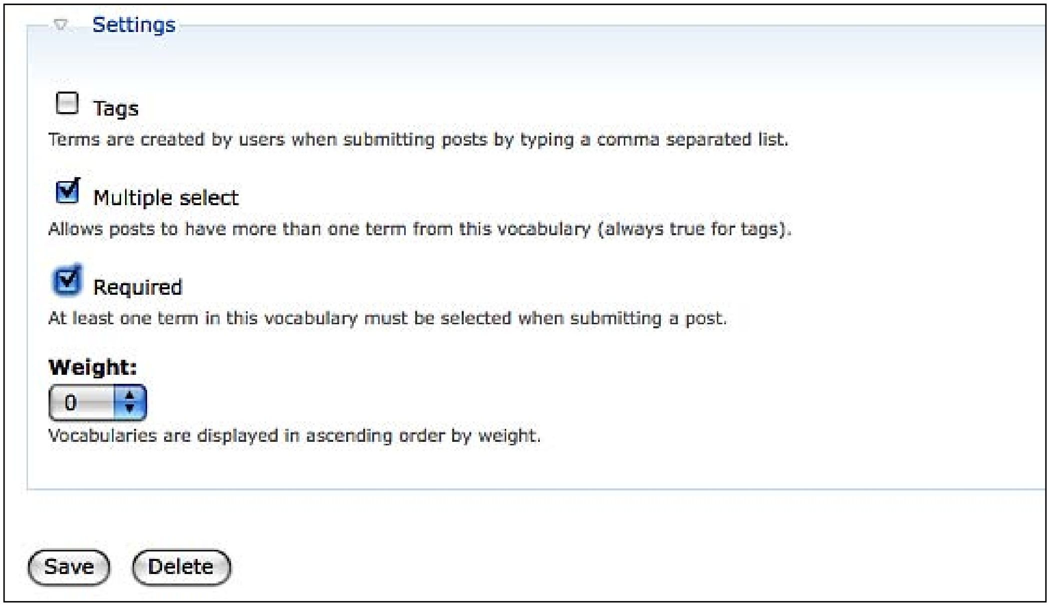


- **Content types**: There are only two content types selected for Catalog: **Product** and **Product kit**. Theoretically, you can create new content types and assign them to Catalog, but for now we're going to use only these two default types.



- **Settings**: This group has four fields:
    - **Tags**: Select this box only if you are not going to use structured organization for your content, but rather free tagging. For the Catalog vocabulary, leave this unchecked.
    - **Multiple select**: You may need to put some products in more than one category, so leave this box checked.
    - **Required**: You don't want to have orphaned products outside the product catalog, so you have to check this box. This ensures that whenever you submit a product, you'll also enter the appropriate category.
    - **Weight**: If you have more than one vocabulary, this sets the order in which they are displayed. The lighter ones are displayed first and the heavier ones are displayed last.

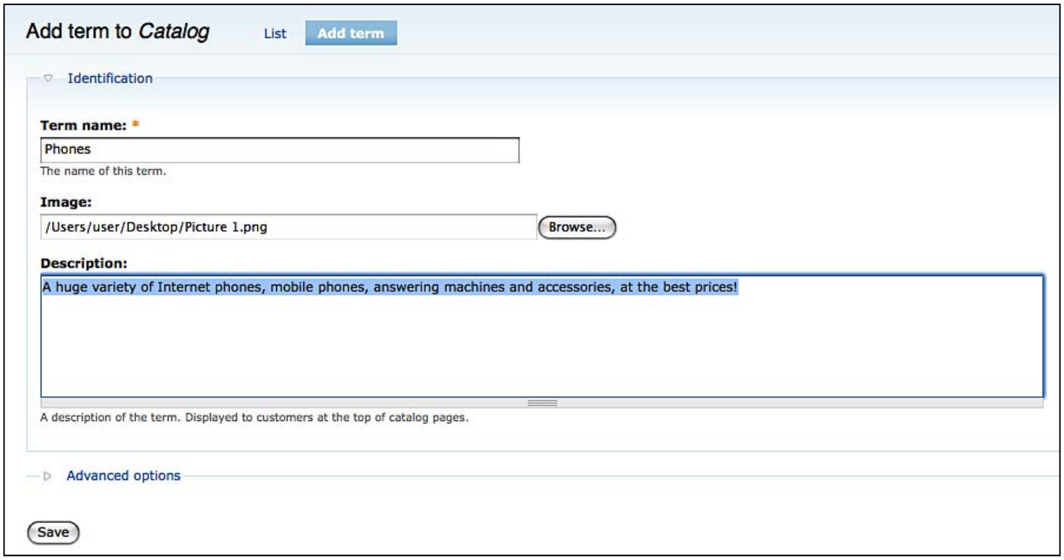
Click on **Save** to store the changes and return to the main taxonomy page.



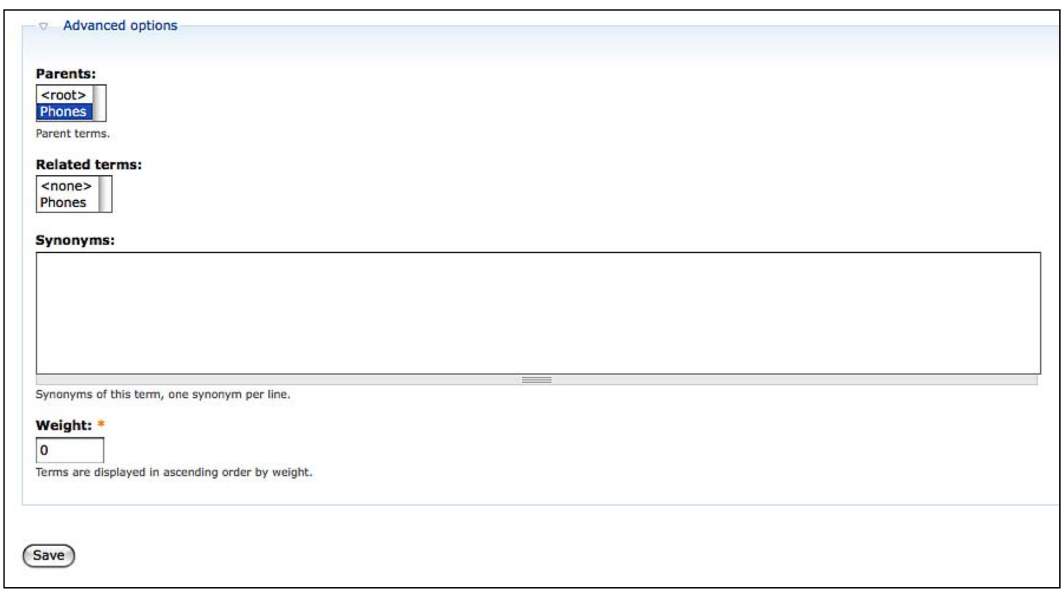
Now, let's create the structure of your product catalog, by adding your categories and subcategories. Click on **add terms** and you'll be transferred to a new page, where you can add terms to the Catalog vocabulary. The following are the fields that you have to fill here:

- **Term name**: This is the name of your category. Enter the name as **Phones** for your first category.
- **Image**: Select an image for this category that will be displayed when your visitors are browsing the store catalog.
- **Description**: Finally, enter a description for your category. The description is visible on the page of the category.

Skip the **Advanced options** section, and click on **Save**. Your first category is created!



Now, let's create some subcategories for the **Phones** category. Enter **Corded Phones** for term name. Skip the **Image** and **Description** fields. Open the **Advanced options** group and select **Phones** for **Parents**. Click on **Save**.



Following the same procedure, create the following structure at your store:

- Phones
    - Corded Phones
    - Cordless Phones
    - Mobile Phones
    - Internet Phones
    - Accessories
- Music Players
    - iPods
    - MP3 Players
    - CD Players
    - Accessories
- Cameras
    - Digital Cameras
    - Camcorders
    - Memory Cards
    - Accessories
- Games
    - Consoles
    - Video games
    - Radio-controlled
    - Accessories

The preceding structure is shown in the following screenshot:

Name	Operations
Phones	edit
Corded Phones	edit
Cordless Phones	edit
Mobile Phones	edit
Internet Phones	edit
Accessories	edit
Music Players	edit
iPods	edit
MP3 Players	edit
CD Players	edit
Accessories	edit
Cameras	edit
Digital Cameras	edit
Camcorders	edit
Memory Cards	edit
Accessories	edit
Games	edit
Consoles	edit
Video Games	edit
Radio-controlled	edit
Accessories	edit

Let's say that your catalog structure is different, and you have to re-arrange some categories and subcategories. In the old days, the only solution would be to edit each one of them manually and change their weight or assign them to a different parent category. Not anymore! Drupal 6 offers drag-and-drop capabilities, so you can do your corrections very quickly. So, if you want to put the **MP3 Players** category above the **iPods** category, you just click on it and drag it. Then you have to click on **Save** to store the new structure. Using the same method, we can assign a subcategory to a new main category, and also transform a main category to a subcategory or vice versa.

This can be done as shown in the following screenshot:

Your product catalog structure is completed, but how can you navigate through your categories and subcategories? Ubercart gives you such functionality, but you have to enable the catalog block first. Go to **Home | Administer | Site building | Blocks**. Find a block named **Catalog**, select **Left sidebar** from the drop-down menu, and click on **Save blocks**. Now, you can see a new menu called **Catalog** in the left column of your site. Keep in mind, though that you won't be able to see the categories, until you add at least one product in them. By default, Ubercart does not show empty categories.

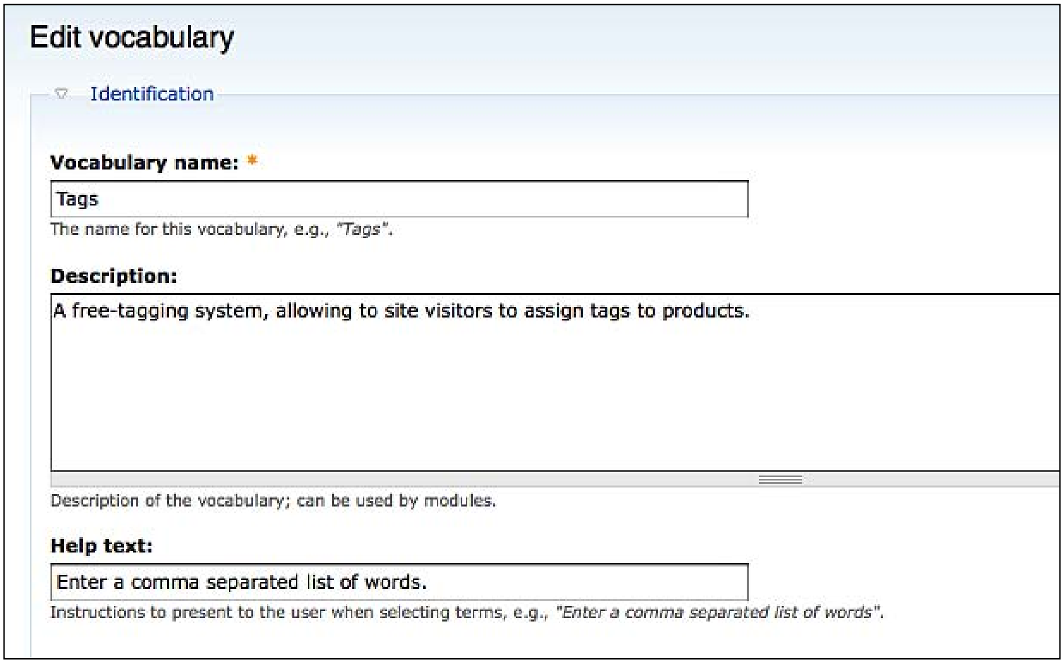
Creating user-generated free tagging

The aim of your catalog is to help users navigate with ease inside your store and quickly find whatever they search for. But, if you build a huge product catalog, with thousand of products and hundreds of categories, navigation becomes complicated and difficult.

Search is useful if someone knows exactly the name, SKU, or description of the product, he or she is looking for, but what if someone is looking for "lightweight MP3 player, ideal for running"? If these characteristics are not written in the product description, he or she won't receive any results, even if you actually have a "lightweight MP3 player, ideal for running" in your store. So, how can you attach extra information on your products? The answer is **Free tagging**.

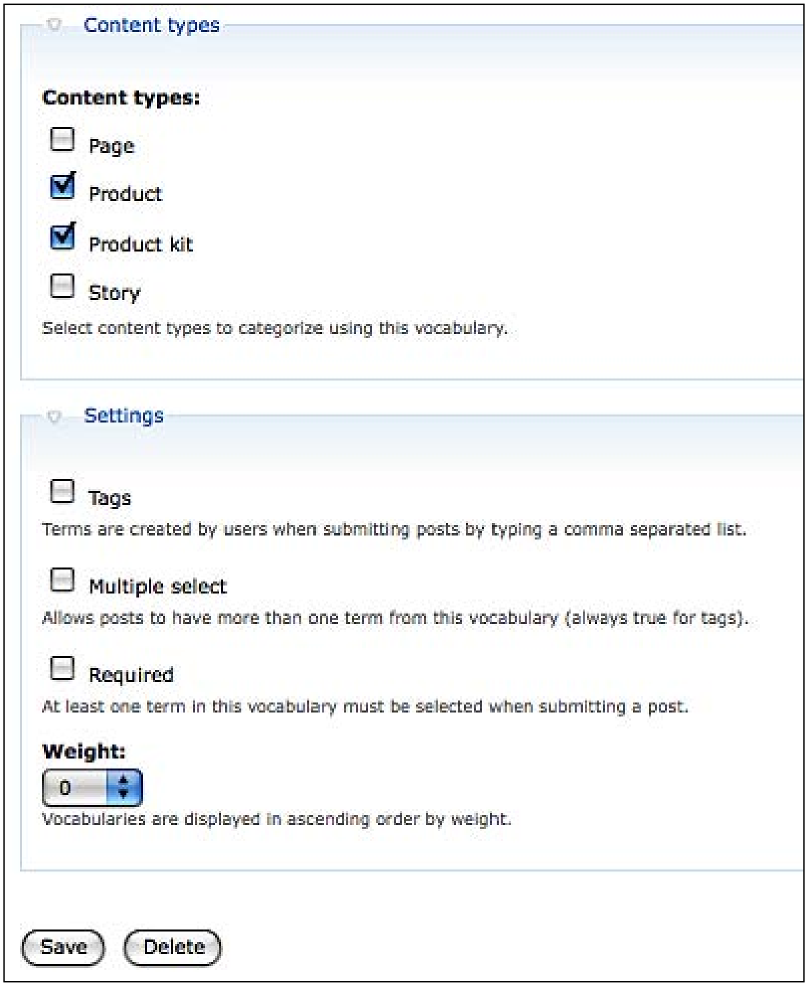
Free tagging allows us to describe our products better, using keywords (tags). One way to do this would be to insert tags to our products when we create them. But we believe that this is not the best solution. We want to trust our visitors and let them assign user-created tags to our products. This tagging system, called **folksonomy**, is very common nowadays among popular websites, and it's proven to be very reliable. So, let's go again to **Home | Administer | Content management | Taxonomy**. Click on **Add vocabulary** and complete the following fields:

- **Vocabulary name: Tags**
- **Description**: A free-tagging system, allowing the site visitors to assign tags to products.
- **Help text**: Enter a comma-separated list of words.



- **Content types**: Check **Product** and **Product kit**

- **Settings**: Check **Tags** and **Multiple select**

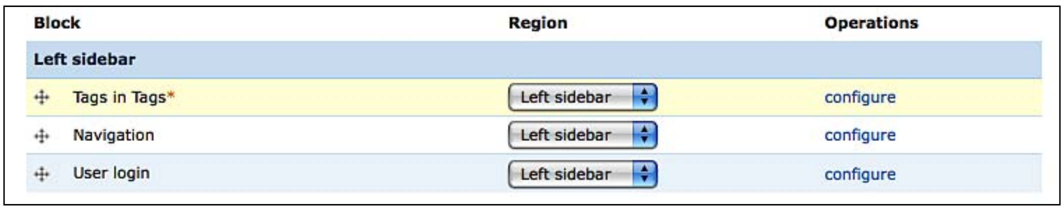


- And now click on **Save** to create your new vocabulary. You don't have to add any new terms, because every time a user adds a new tag, a new term is automatically created by Drupal.

Now, your tagging system is ready, but you face a problem. You want to allow each user to add tags, but in Drupal taxonomy management, it is an administrative-only process. You also want to create a block in order to display the most popular tags, in the form of a tag cloud. There's no need to write any code for this, but you have to install two third-party modules: **Community Tags**, which allows each user to add tags to content and **Tagadelic**, which creates a "tag cloud" page and block.

So, let's first browse to `http://drupal.org/project/community_tags` to download the **Community Tags** module, and then to `http://drupal.org/project/tagadelic` to download the **Tagadelic** module. Select the latest version for Drupal 6.x. Unzip these files and copy them into the `/sites/all/modules` folder of the Drupal site. Then go to **Home | Administer | Site building | Modules**, enable these two modules, and click on **Save configuration**.

The final step is to enable the Tag Cloud block. Go to **Home | Administer | Site building | Blocks**. Find a block named **Tags in Tags**, select **Left sidebar** from the drop-down menu, and click on **Save blocks**.



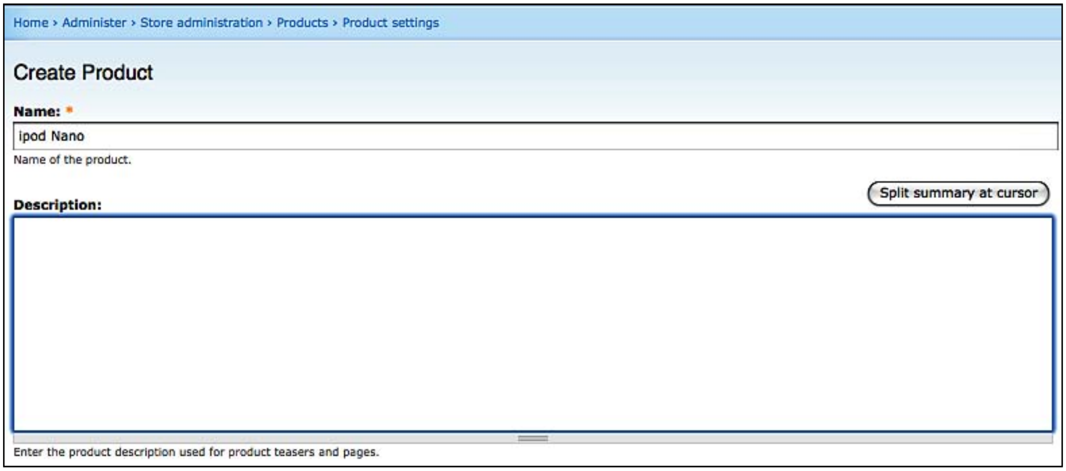
The new block is activated, but there's still nothing to see, as you haven't inserted any tags yet. You have to wait until the creation of the first product to actually see this block, but you also have to insert several tags before you can see a really meaningful tag cloud.

Creating and managing products

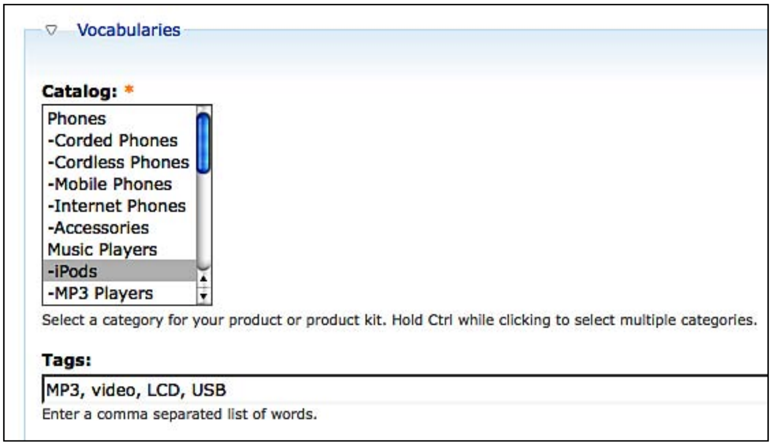
So, now that you have completed the customization of your vocabularies and inserted your terms, you are ready to create your first product. For Drupal engine, new content types are nothing more than nodes with extra fields; in this approach a new product is only another content type, with some special properties created by the Ubercart module. So, if you have previous Drupal experience, creating a new product is just a piece of cake, but if you don't, just carry out the following steps carefully:

1. Go to **Home | Create content | Product**. This opens the page for the creation of a new Product. **Name** is obviously the name of our product. Insert **ipod Nano**.

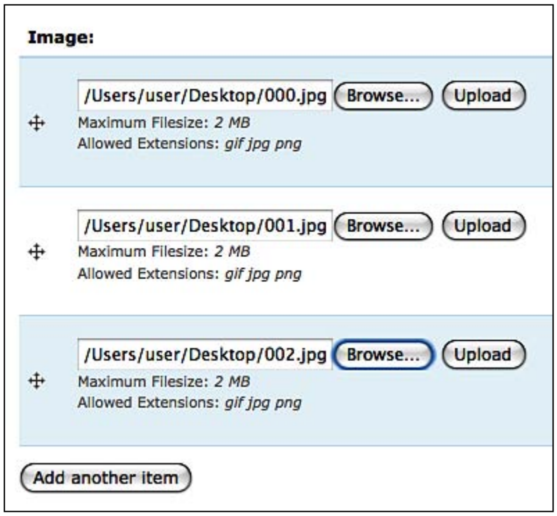
2. **Description** is a detailed description of the properties and characteristics of your product. You can show this field in product teasers or in product pages. Insert "The new iPod nano features a host of features and an elegant new design that beautifully blends a brilliant two-inch display with a gorgeous curved aluminum and glass enclosure" in the **Description** field. Actually, there is no limit to the size of the description—it can be a few words or many pages, and it only depends on what depth of information you want to provide to your clients.

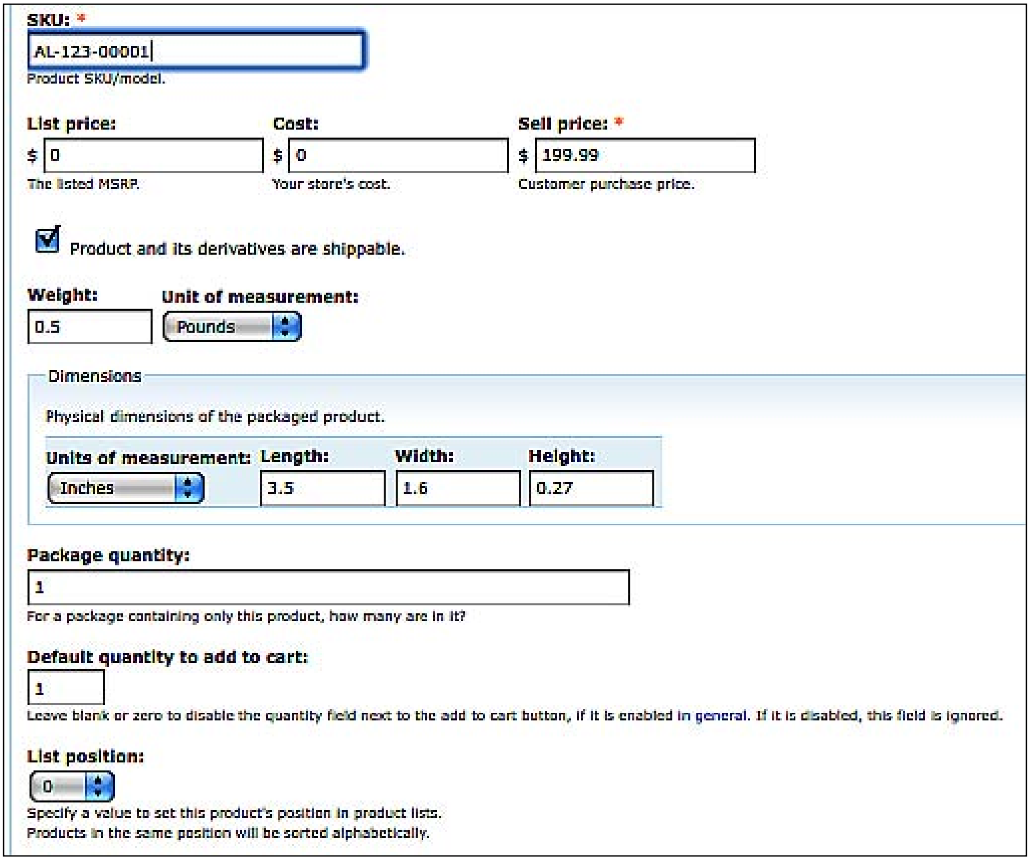


3. Now let's move to the **Vocabularies** section. Here, select **iPods** from the **Catalog** list. Holding down *Ctrl*, you can select multiple categories for your product, but this is not needed here.



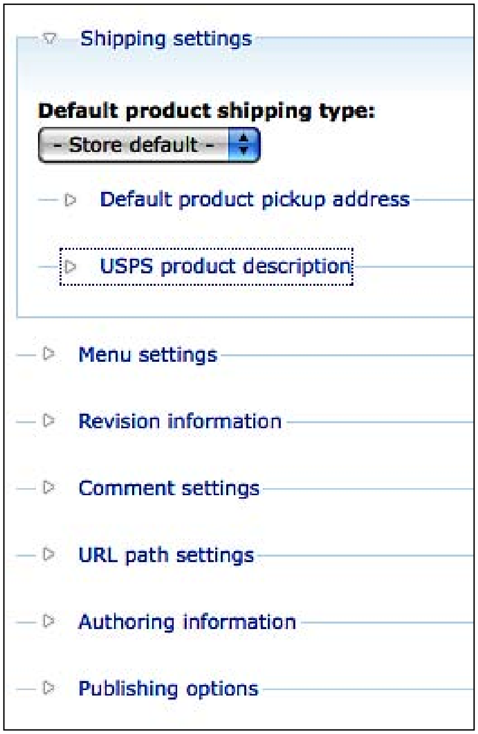
4. Now let's add some tags. Go to the **Tags** field and enter **MP3, video, LCD, USB**.
5. The **Image** field is straightforward. Click on **Browse…** to select the image and **Upload** to upload it to the server. Clicking on **Add another item** allows entering multiple images for the same product.



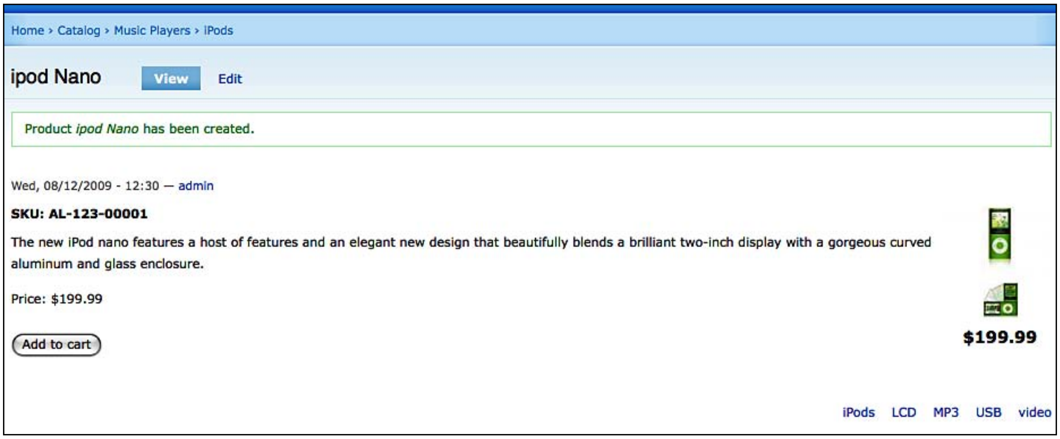


7. The next three fields are **List price**, **Cost**, and **Sell price**. **List price** is the starting price of the product, **Cost** shows for how much you purchased it, and **Sell price** is the price that you actually sell the product for. The only required field here is **Sell price**, so complete only this field.
8. In the next checkbox you have to select whether your product is shippable or maybe just downloadable. Your electronics store has only physical products, so leave this selected for all your products.
9. Enter the **Weight** of the product, and select the **Unit of measurement**. It's not a required field, but we'll use it later to create a shipping quote, so enter a value here.
10. The same applies to the **Dimensions** of the product. They're not required, but some transporters use them to calculate shipping rates, so select the **Units of measurement** and enter the **Length**, **Width**, and **Height** of the product.

11. **Package quantity** shows how many pieces of this product are contained inside every package that you sell. For your store, this value is usually **1**.
12. If you want a quantity field next to the **Add to cart** button, enter the value **1** for the **Default quantity to add to cart**. However, if you don't want this field, enter the value **0**, or leave this field blank.
13. The default sorting in product listings and category pages is alphabetical, so if you want to create a custom sorting and rearrange the position of your products, you can use the **List position** field. Giving a smaller value to this field puts this product closer to the top of the listings. There is no need to make any customizations right now, so leave this field as it is.
14. Leave the default values for **Shipping settings** and all the following settings, and click on **Save**.



15. Your first product is created!



Now, let's explore together the distinct regions of the product page.

- At the top there is breadcrumb navigation. Clicking on a term directs you to the specific catalog page, or the home page.
- Below the breadcrumb there is the name of the product.
- Next to it, there is an **Edit** button. It is visible only to the store administrators, so don't worry, store visitors cannot edit your products!
- The next region shows the creation date of your product, and who the creator was.

- At the center of the page, there is the most important information about the product. First the SKU, then the product description, and the product images at the right-hand side. Clicking on an image opens a new image layer, where you can see all the product images in the original size.

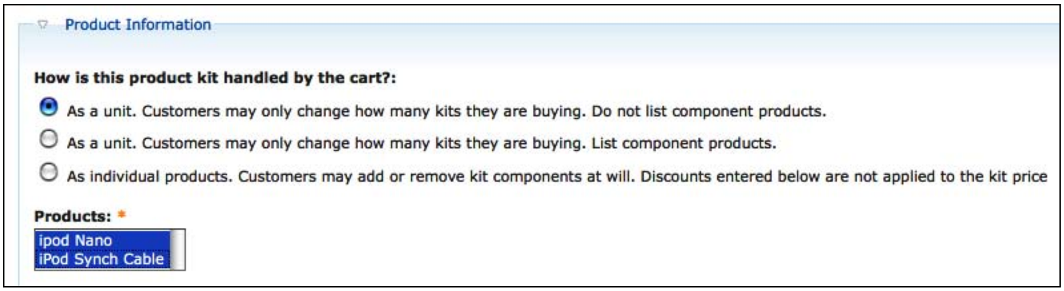
- At the bottom right of the page there is a list of all the related tags and categories. Clicking on a word transfers the visitor to a new page showing all the products that belong to this specific category or have this specific tag attached.
- And finally, there is the **Add to cart** button. Clicking on this button obviously adds this product to the shopping cart.

Creating a product kit

Ubercart automatically creates two content types: Products and Product kits. A Product kit consists of two or more products, grouped together as a single package. Before you can create a product kit, you need at least two products, so let's create one more product for our store, following the procedure that we showed you earlier. So, go to `http://mysite.com//node/add/product`. Enter **iPod Sync Cable** as a name and description. Select **Music Players | Accessories** as a category, and insert an SKU and a sell price. You don't need anything else, so go to the bottom of the page and click on **Save**.

In order to create the Product kit go to **Home | Create content | Product kit**.

If you explore this page, you'll see that there are many similarities between this and the single product creation page. You need a name, a description, and a category and you can add images and tags. There is no need for selling price, because Ubercart automatically adds the prices of the single products. So, let's add **iPod Nano** as name and **Synch Cable** as description, select **iPods** as the category and add the tags **MP3, video, LCD, USB**. The next two fields are the ones that make all the difference between a single product and a product kit:



The first one asks us to decide **How this product kit handled by the cart?** You have to select between three options here:

- Display the product kit as a single unit, without showing the single products. The client has the perception that he or she buys a single product with two components.
- Display the product kit, but also display the single products. The client understands that he or she buys two separate products, packed together as a single unit.
- Display the products separately. Using this method, the client can actually break the product kit, and add or remove separate components at the shopping cart.

For your product kit, select the first method.

The second field displays all the available products. You have to select which products you are going to include in your product kit. We have only two products, so select both of them. Finally, go to the bottom of the page and click on **Save**. Your product kit is created.

Importing products from text files

Inserting products by hand is an acceptable method if you have only a few products in your store. But what happens if you have hundreds, or even thousands of products? This method is time-consuming and not very productive for large stores. Consider that, if you have 10,000 products and it takes only five minutes to insert a single product, you need about five working months to complete the procedure. And you'll also have to face problems such as slow Internet connection or typing errors. That's why the best solution is to import the products from a text file. You can import products from three types or files: Comma Separated Values files, Semicolon Separated Values files, or Tab Separated Values files.

A **Comma Separated Values** (**CSV**) file stores the contents of a table or a list in a file format where a line in the CSV file represents a row of the table and fields are separated by commas, each field belonging to one table column.

For example, if you have the following table:

Name	Category	SKU	Price
iPod Nano	iPods	100001	199.99
iPod Classic	iPods	100002	249.99
iPod Shuffle	iPods	100003	79.99

The following is the CSV representation of this table:

"Name","Category","SKU","Price"

"iPod Nano","iPods",100001,"199.99"

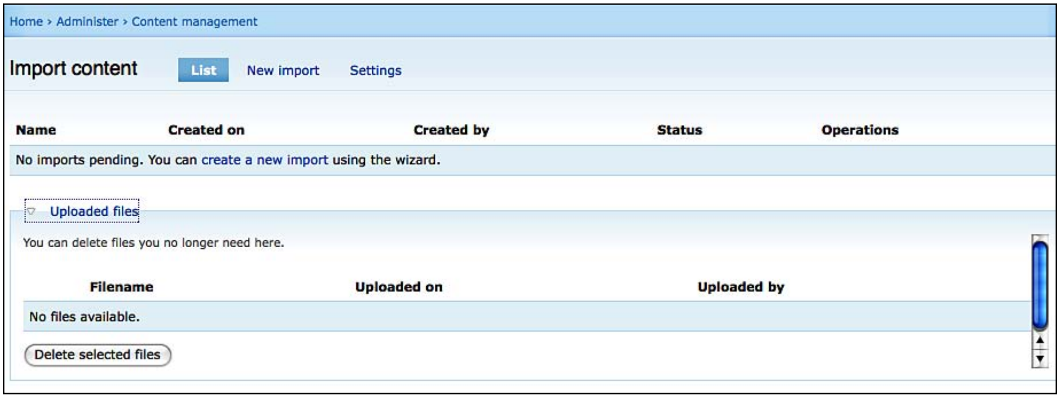
"iPod Classic","iPods",100002,"249.99"

"iPod Shuffle","iPods",100003,"79.99"

In **Semicolon Separated Values** (**SSV**) files, you replace commas with semicolons, and in **Tab Separated Values** (**TSV**) files you replace commas with tabs. We'll use CSV files in our example, but the procedure is identical for the other types of files.

Before you can import CSV files to Ubercart, you first have to install the Node import module. To do so, follow these steps:

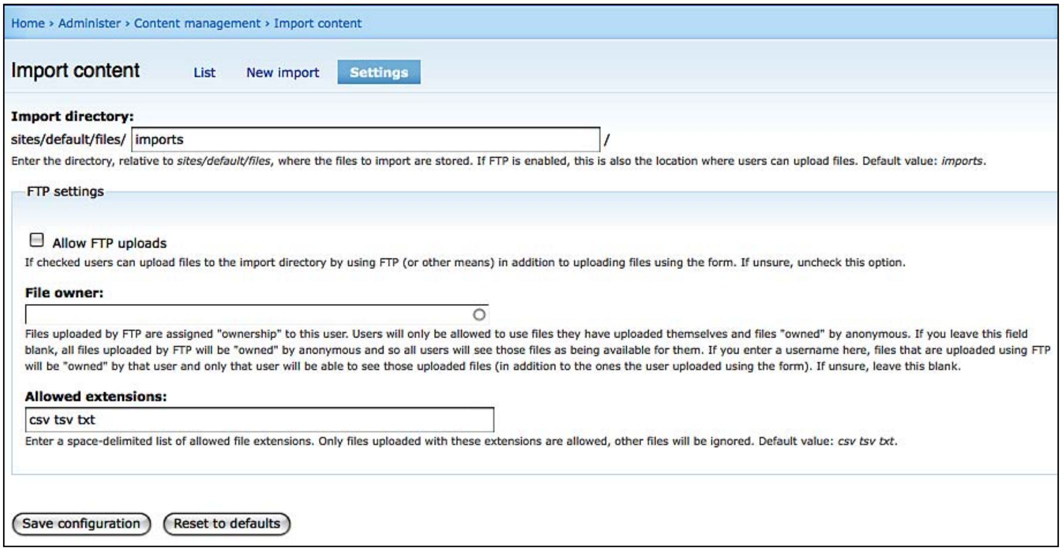
1. Browse to `http://drupal.org/project/node_import` and download the Node import module. Select the latest version for Drupal 6.x. If you take a moment and read more about the Node import module, you will see that this module depends on Date API and Advanced Help modules. Therefore, in order to enable the Node import module you need to download these two modules. Go to `http://drupal.org/project/date` to download the Date API module, and then to `http://drupal.org/project/advanced_help` to download the Advanced Help module.
2. Unzip these files and copy them into the `/sites/all/modules` folder of the Drupal site.
3. Now go to **Home | Administer | Site building | Modules**, enable this module, and click on **Save configuration**.
4. The import content page is available at **Home | Administer | Content management | Import content**.



There are three tabs in this page:

- **List** shows all the created imports and displays all the uploaded files. You haven't created anything yet, so all the listings are empty.
- **New import** opens the new import wizard. We'll take a look at it in a while.
- **Settings** opens the configuration page.

Before you start your first import, click on **Settings** to take a look at the configuration page.



- **Import directory** is the directory where you can upload your CSV files, using FTP or the import wizard.
- If you want to upload files using FTP, you have to check **Allow FTP uploads**.
- For security reasons, if you allow FTP uploads, you also have to insert a file owner in the next field. If not, the uploaded files will be available for anyone, even for the anonymous users.
- In the **Allowed extensions** field, declare which file extensions are allowed. All the others are simply ignored.

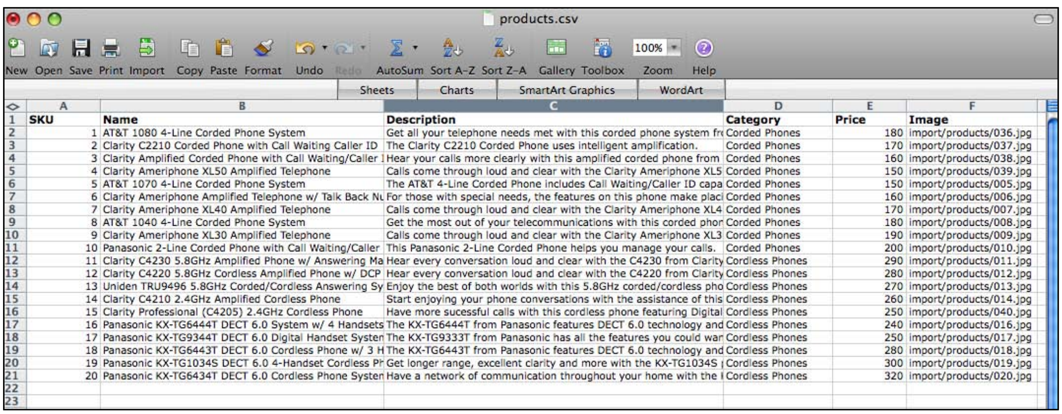
For your store, you don't have to change anything, so just click on **Save configuration** to return to the previous page.

Before we begin exploring the import wizard, let's take a closer look at the CSV file, and see what is the proper structure for product imports and how you can create such files.

The easiest way to create a CSV file is using a Spreadsheet Application such as Microsoft Excel, Open Office, or Numbers. All these applications store their files in a default format, but they can easily export them as CSV files. Inserting files in a spreadsheet is a much faster process than working directly in Ubercart. Of course, the biggest advantage is that if you have your product database ready in another application, you can export it as a spreadsheet, convert it to a CSV file, and import it into Ubercart. Using this method, hard work that would require several months, becomes an easy job that takes only a few hours!

In a proper spreadsheet file, every row represents a product, and every column represents a property of the products. Remember the product insertion form? Actually, for every field of this form, you have to create a separate column. For your store you don't need all the fields, so you are going to use only the most important ones.

As an example, we created the following spreadsheet:

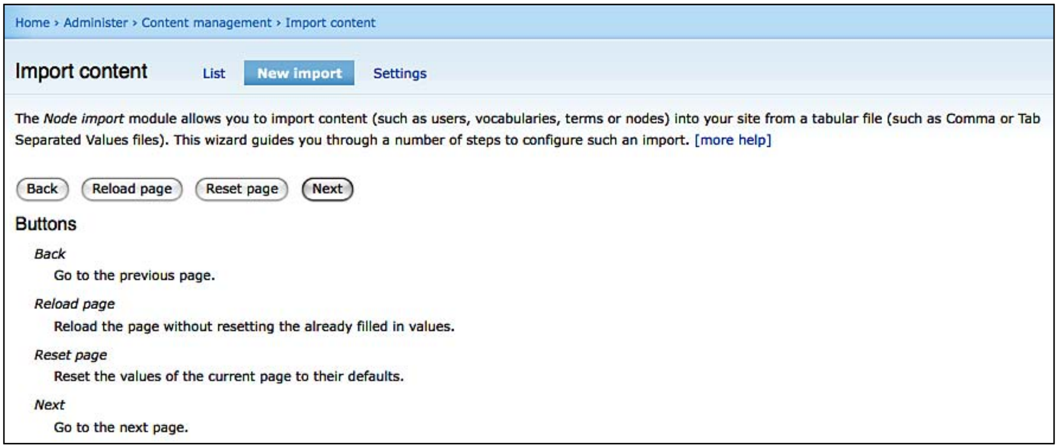


	A	B	C	D	E	F
1	SKU	Name	Description	Category	Price	Image
2	1	AT&T 1080 4-Line Corded Phone System	Get all your telephone needs met with this corded phone system fr	Corded Phones	180	import/products/036.jpg
3	2	Clarity C2210 Corded Phone with Call Waiting Caller ID	The Clarity C2210 Corded Phone uses intelligent amplification.	Corded Phones	170	import/products/037.jpg
4	3	Clarity Amplified Corded Phone with Call Waiting/Caller I	Hear your calls more clearly with this amplified corded phone from	Corded Phones	160	import/products/038.jpg
5	4	Clarity Ameriphone XL50 Amplified Telephone	Calls come through loud and clear with the Clarity Ameriphone XL5	Corded Phones	150	import/products/039.jpg
6	5	AT&T 1070 4-Line Corded Phone System	The AT&T 4-Line Corded Phone includes Call Waiting/Caller ID capa	Corded Phones	150	import/products/005.jpg
7	6	Clarity Ameriphone Amplified Telephone w/ Talk Back Nu	For those with special needs, the features on this phone make placi	Corded Phones	160	import/products/006.jpg
8	7	Clarity Ameriphone XL40 Amplified Telephone	Calls come through loud and clear with the Clarity Ameriphone XL4	Corded Phones	170	import/products/007.jpg
9	8	AT&T 1040 4-Line Corded Phone System	Get the most out of your telecommunications with this corded phor	Corded Phones	180	import/products/008.jpg
10	9	Clarity Ameriphone XL30 Amplified Telephone	Calls come through loud and clear with the Clarity Ameriphone XL3	Corded Phones	190	import/products/009.jpg
11	10	Panasonic 2-Line Corded Phone with Call Waiting/Caller	This Panasonic 2-Line Corded Phone helps you manage your calls.	Corded Phones	200	import/products/010.jpg
12	11	Clarity C4230 5.8GHz Amplified Phone w/ Answering Ma	Hear every conversation loud and clear with the C4230 from Clarity	Cordless Phones	290	import/products/011.jpg
13	12	Clarity C4220 5.8GHz Cordless Amplified Phone w/ DCP	Hear every conversation loud and clear with the C4220 from Clarity	Cordless Phones	280	import/products/012.jpg
14	13	Uniden TRU9496 5.8GHz Corded/Cordless Answering Sy	Enjoy the best of both worlds with this 5.8GHz corded/cordless pho	Cordless Phones	270	import/products/013.jpg
15	14	Clarity C4210 2.4GHz Amplified Cordless Phone	Start enjoying your phone conversations with the assistance of this	Cordless Phones	260	import/products/014.jpg
16	15	Clarity Professional (C4205) 2.4GHz Cordless Phone	Have more sucessful calls with this cordless phone featuring Digital	Cordless Phones	250	import/products/040.jpg
17	16	Panasonic KX-TG6444T DECT 6.0 System w/ 4 Handsets	The KX-TG6444T from Panasonic features DECT 6.0 technology and	Cordless Phones	240	import/products/016.jpg
18	17	Panasonic KX-TG9344T DECT 6.0 Digital Handset Systen	The KX-TG9333T from Panasonic has all the features you could war	Cordless Phones	250	import/products/017.jpg
19	18	Panasonic KX-TG6443T DECT 6.0 Cordless Phone w/ 3 H	The KX-TG6443T from Panasonic features DECT 6.0 technology and	Cordless Phones	280	import/products/018.jpg
20	19	Panasonic KX-TG1034S DECT 6.0 4-Handset Cordless Pr	Get longer range, excellent clarity and more with the KX-TG1034S	Cordless Phones	300	import/products/019.jpg
21	20	Panasonic KX-TG6434T DECT 6.0 Cordless Phone Systen	Have a network of communication throughout your home with the	Cordless Phones	320	import/products/020.jpg
22						
23						

It has eight columns: SKU, Name, Description, Category, Price, Image, Tags, and Weight. You can create the same structure in your favorite application and save the file as CSV.

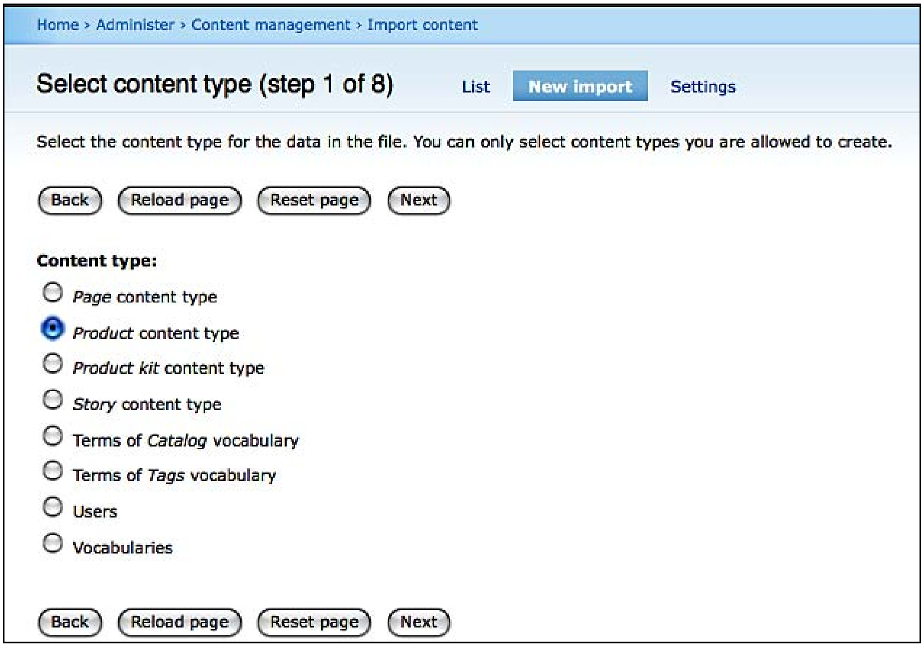
The final step is to upload the images of your products to the server. Using an FTP application, upload all the files into the `sites/default/files` folder. As you can see in the spreadsheet, we are using a new folder named `import` to separate these images from all the other images of the site. You can use the same convention and keep another name for your folder. Just keep in mind that you have to update the Image column also. By default the images root folder is `sites/default/files`. For instance, if you add your images in the `products` folder inside the `files` folder, then you need to change the image path column by adding the appropriate path prefix, in our case the value would be `/products/036.jpg`.

After you have saved the file and uploaded the images, return back to your Ubercart store and click on the **New import** button. The import wizard begins.

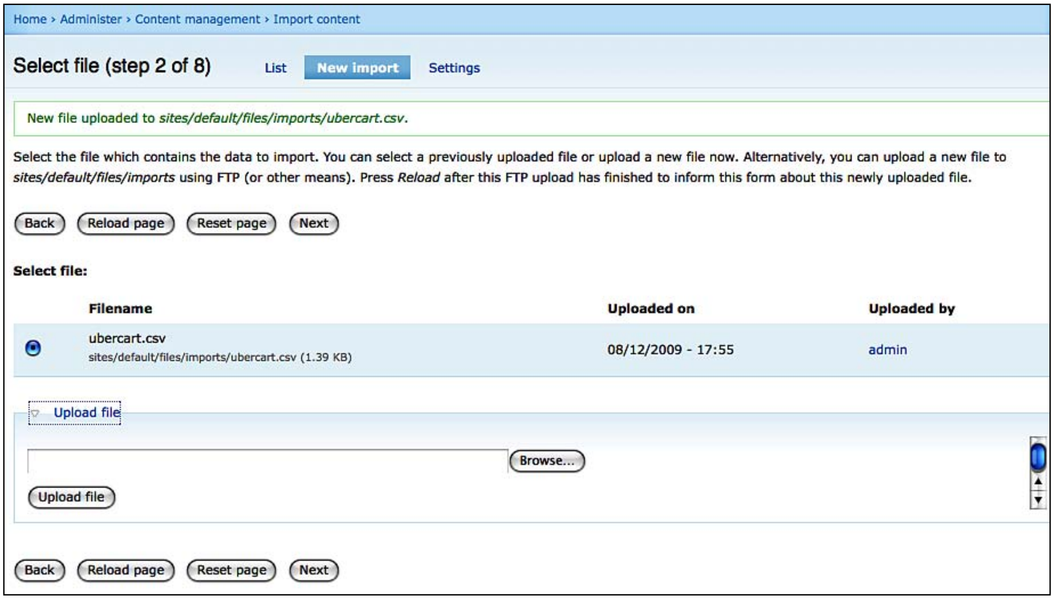


The introductory page shows basic help about this procedure and shows the main buttons. There's nothing to do here, so click on **Next**.

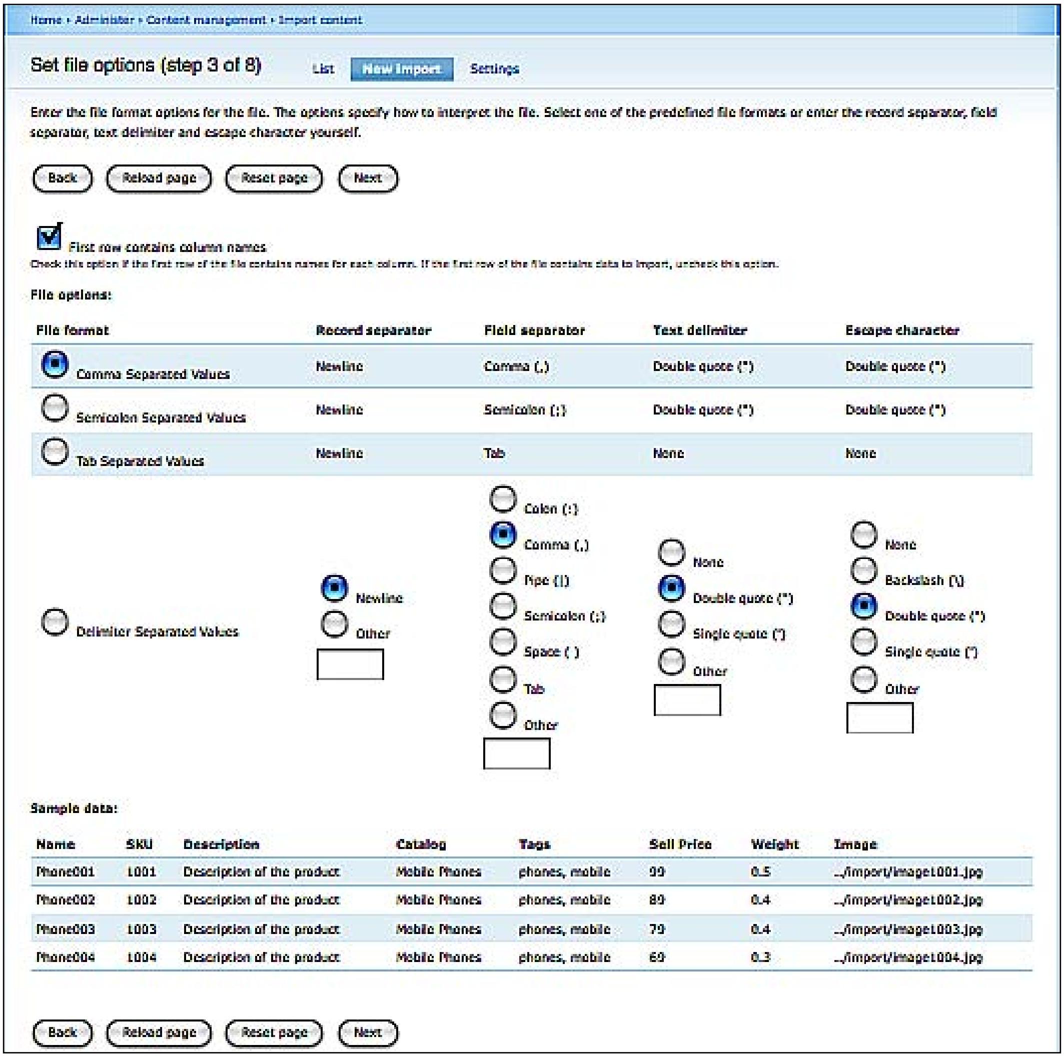
1. In the first page of the wizard, you have to select what kind of content type you want to import into your site. Select **Product content type** and click on **Next**.



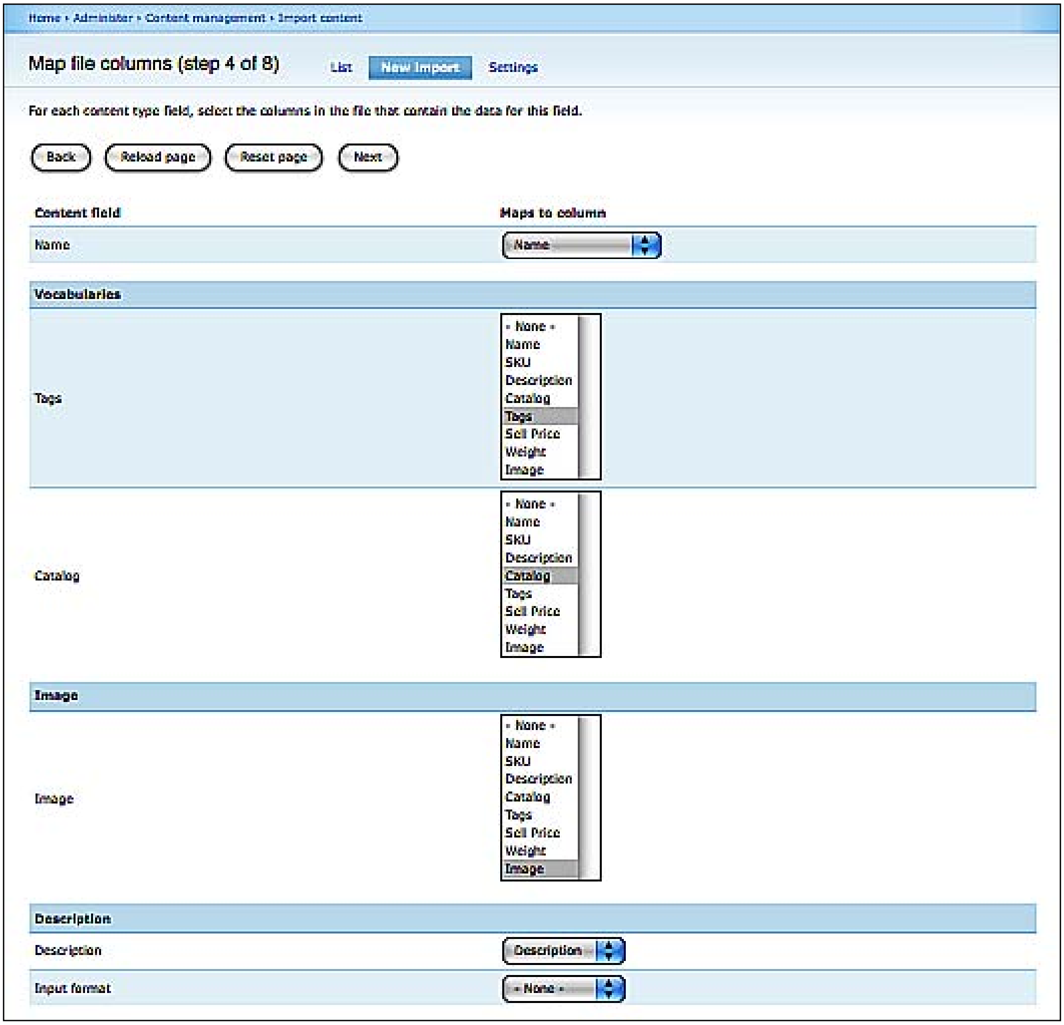
2. In the second page, you have to select and upload the CSV file. If you have already uploaded any files, you can see them as a list and select which file you want to use. If you want to upload a new file click on **Browse**, select the file, and click on **Upload file**. After you have uploaded and selected the CSV file, click on **Next**.



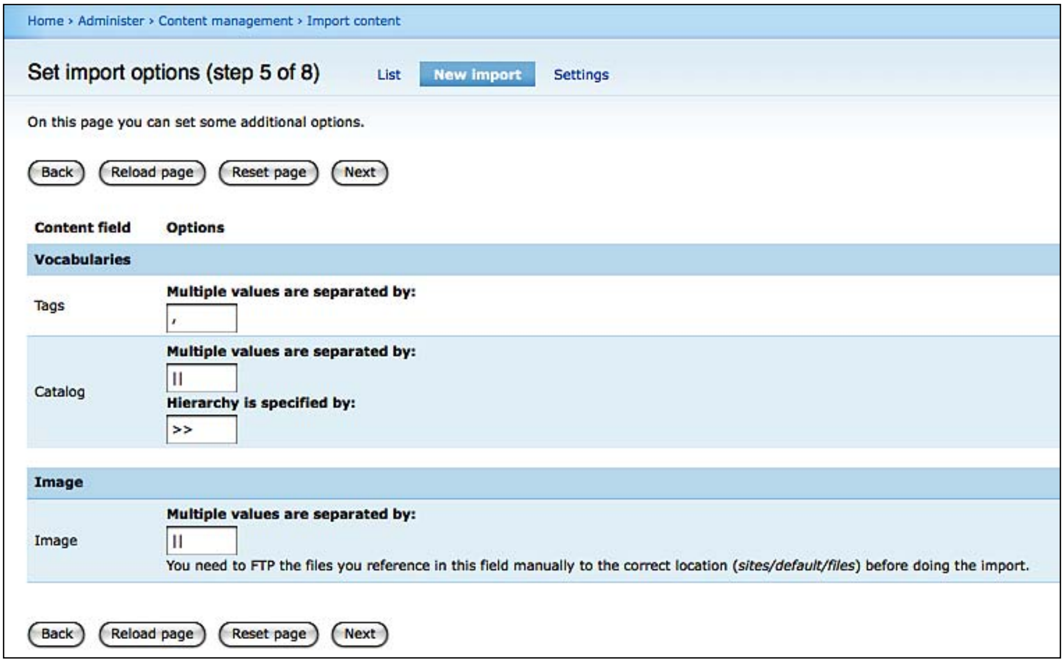
3. In the third page you can make some basic customizations related to your file. First you have to select whether the first row contains column names. This is the case here, so leave this field checked. Then, you have to select the file format. You have created a CSV file, so you don't have to change anything. At the bottom of the page, you can see some sample data, created from the four top rows of the file. So, just click on **Next**.



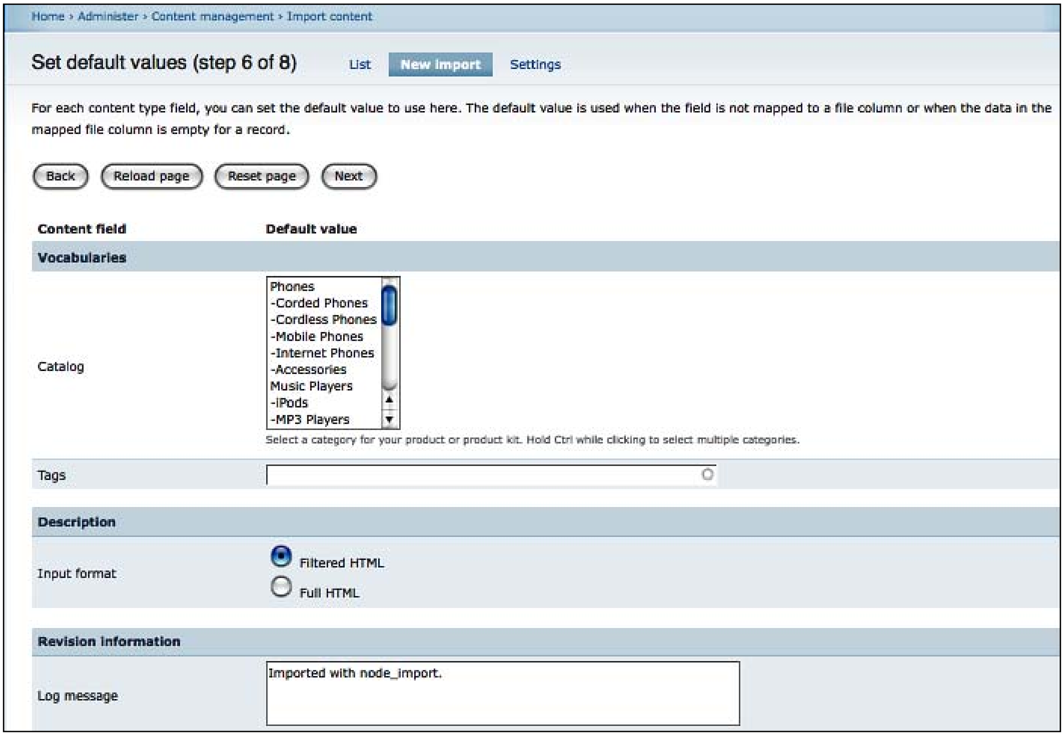
4. In the fourth page you have to match the columns of the file with the fields of the products. If you used the headers that we showed to you in the sample file the node import module is smart enough to automatically map the columns. So, **Tags** columns matches with **Tag** field, **Catalog** column matches with **Catalog** field, and so on. Of course, you have to check every field just to be sure that the mapping is 100 percent accurate. If something is wrong, you can select a different column from the drop-down menus at the right. When everything is right, go to the bottom of the page and click on **Next**.



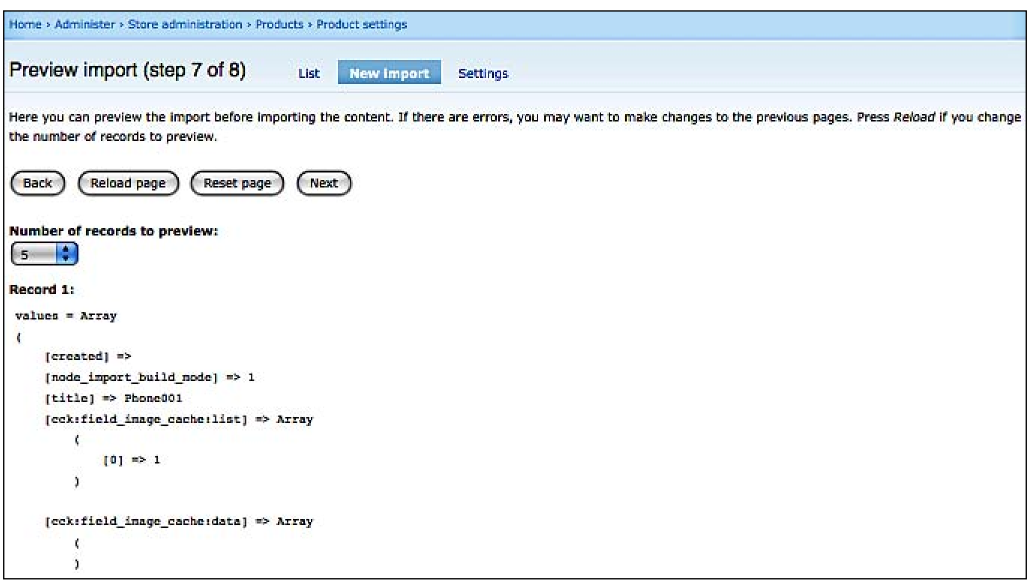
5. In the fifth page you can set some additional options. There's nothing to change here, so click on **Next**.



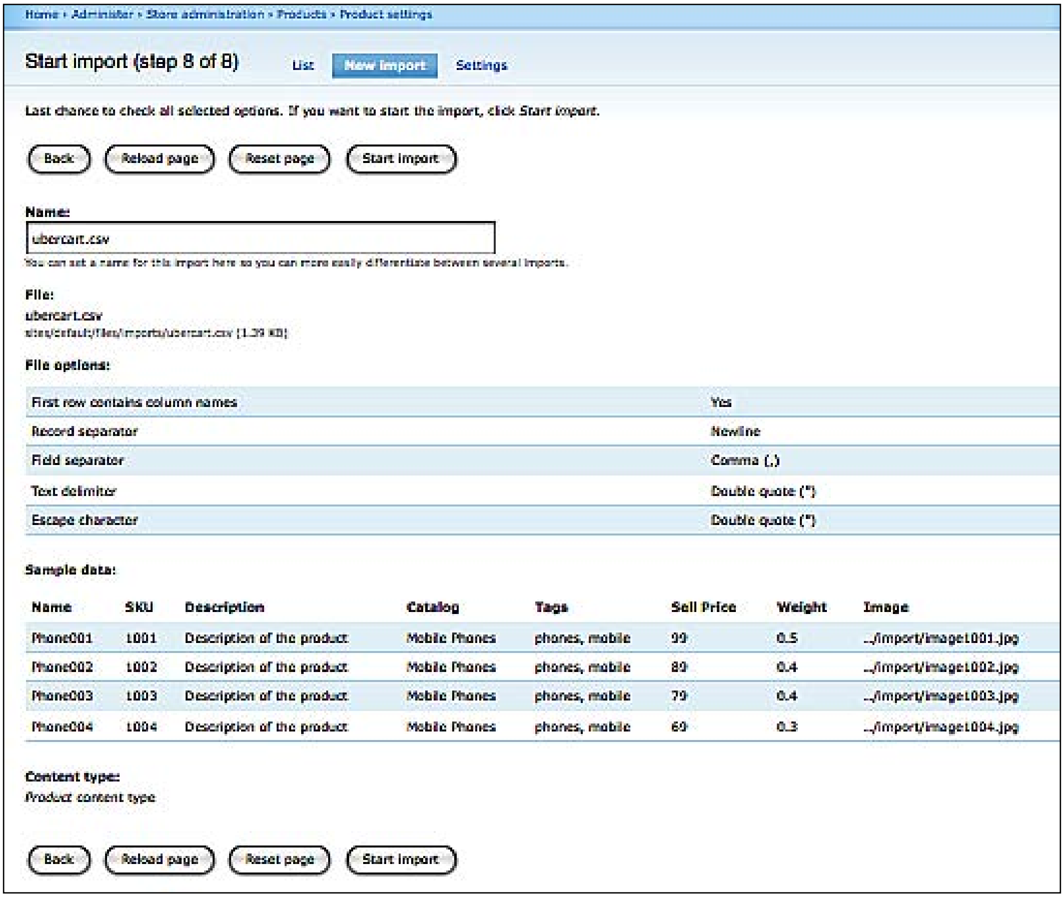
6. In the sixth page set the default values. They are used when the field is not mapped to a file column or when the data in the mapped file column is empty for a record. You don't have to enter default values, because the file is complete and all the columns are mapped, so just click on **Next**.



7. In the seventh page you can preview the import. Take a close look at these queries to see if there are any errors. If everything is right, click on **Next**.



8. Now we reach the eighth page, which is the last page before the final import. You can take a look at the basic options and see some sample data. Click on **Start import** to continue. When the procedure is complete, you can see a report about the imported rows, and the rows with errors. If you don't need the CSV file anymore, it's better to click on **Delete** and remove it from the server. Your new products are now imported.



You can use the same process, no matter whether you want to import 10 or 10,000 products! (Of course the latter takes a little longer.)

Creating product attributes

In Ubercart you can manage attributes at two levels:

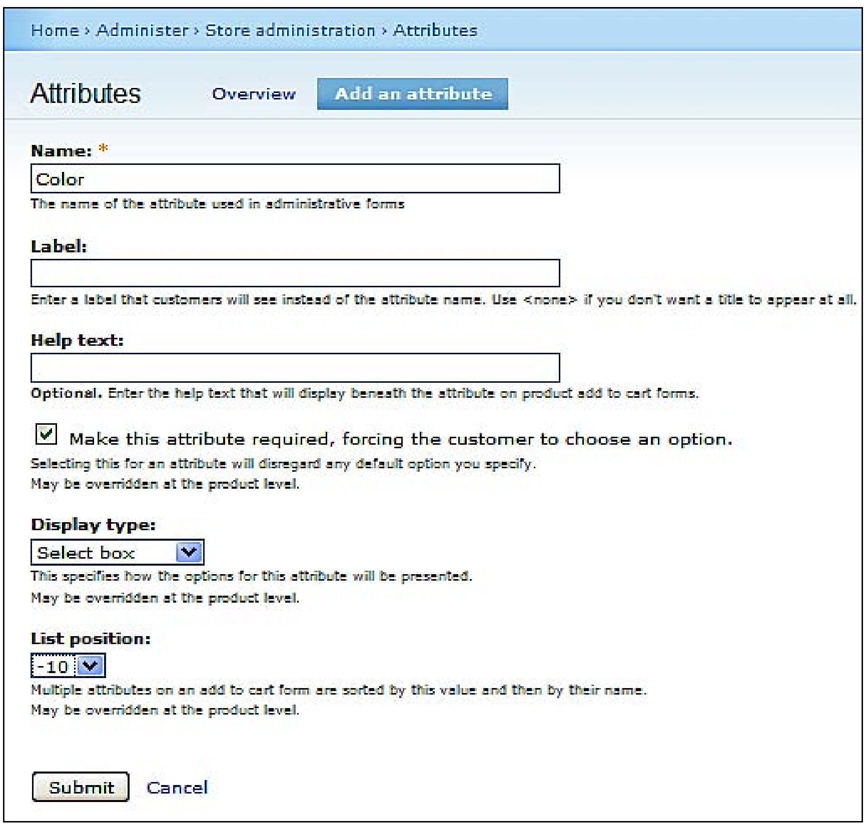
- Using the store settings, you can create and customize attributes, and give them default values. If you have specific groups of products with different attributes for each one, you can create product classes.
- When a new product is created, the default attribute settings are applied to it. After the creation, you can modify them separately for each product.

So, let's start by exploring the default attribute settings for your store. You are going to create two new attributes, **Color** and **Memory**.

1. Go to **Home | Administer | Store administration | Attributes**.
2. At the main attributes page you can see an overview of the created attributes. You haven't created any attributes yet, so click on **Add an attribute** to add a new one.

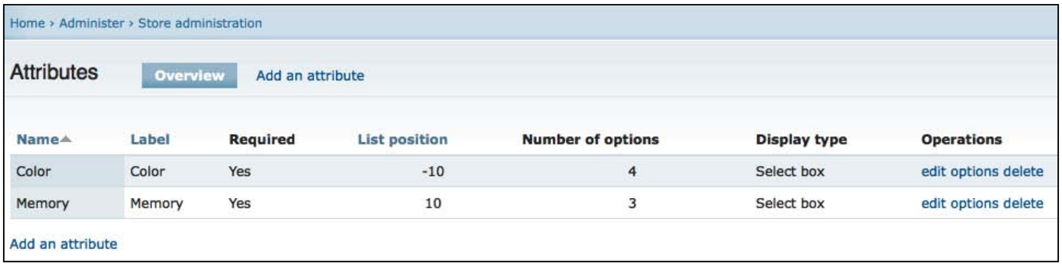
3. The page for the creation of a new attribute opens. Here, you have to enter the basic settings for the attribute:
   - **Name**: The name of this attribute appears at all the places where it's used, such as administrative forms or product pages. For the first attribute enter **Color**.
   - **Label**: If you want to overwrite the attribute name in customer pages, you can use this field. Using the word color everywhere is fine, so leave this field blank.
   - **Help text**: If the name of the attribute is not straightforward, you can create a custom description, visible on cart forms. For this attribute, leave the field blank.

- **Make this attribute required, forcing the customer to choose an option**: You want the customer to select a color before adding a product to the cart, so check this field.
- **Display type**: Leave this as **Select box**.
- **List position**: If you have multiple attributes, you can alter their position by giving them different values at this field. Usually, attributes with smaller values are placed closer to the top. We always want **Color** to be displayed first, so give it a value of **-10**.
- Click on **Submit** to create the attribute.



4. Following the previous procedure, create a new attribute with the name **Memory**, which is required, displays as a **Select box**, and has a list position of **10**.

Now, your two attributes, **Color** and **Memory**, are ready. Before customizing your products, you have to enter the attribute options. These are the different available values for each attribute, for example, all the available colors for the products in your store.

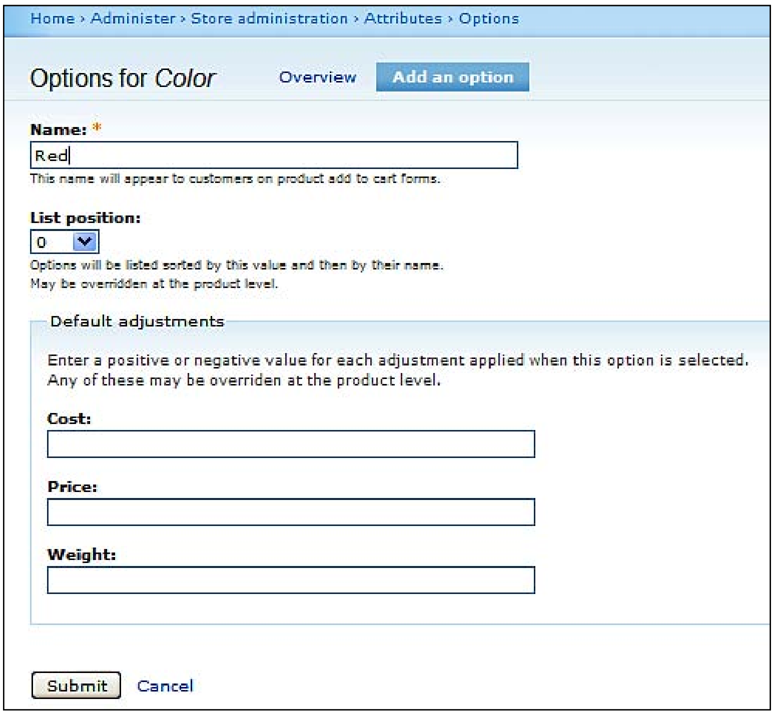


Carry out the following steps:

1. Click on **options** to manage the options for Color.
2. At the main options page, you can see an overview of the created options, and you can create a new option. You haven't created any yet, so click on **Add an option** to add a new one.

3. The page for the creation of a new option opens. Here, you have to enter the basic settings for your attribute.
    - **Name**: The name of this option appears on product **add to cart** forms. For the first option enter **Red**.
    - **List position**: This allows you to rearrange the options. You don't have to change this now.

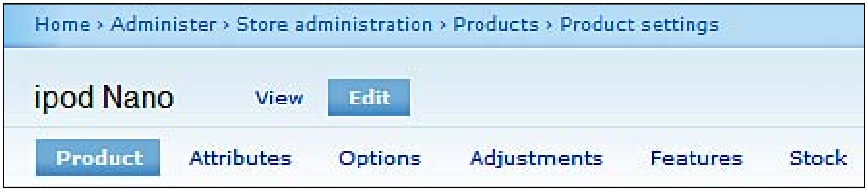
- **Default adjustments**: These are useful if you have a specific difference in cost, price, or weight for this option, compared to the default values of the product. For example, if you would like to sell the red version of your product at $10 more than the default version, you should enter $10 in the price field. This is not the case here, so leave these fields blank.
- Click on **Submit** to create the new color.



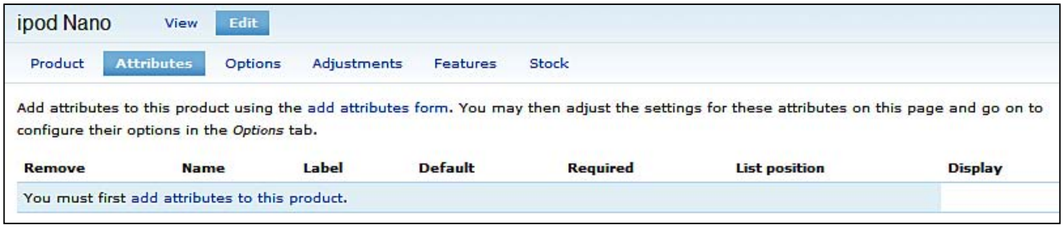
4. Following this procedure, create three more colors: **Green**, **Blue**, and **Yellow**.
5. Now click on **options** for the Memory attribute and create three options **16MB**, **32MB**, and **64MB**.

Now it's time to go to a specific product and assign attributes and options to it. To do so, carry out these simple steps:

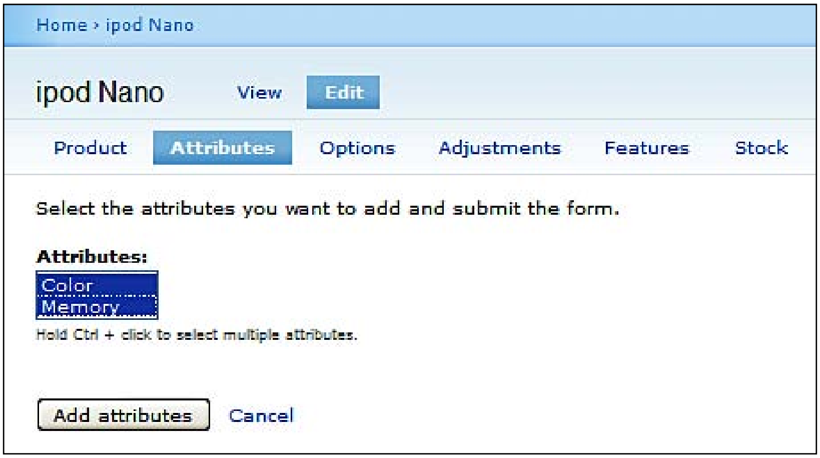
1. Using the **Catalog** block at the left, browse to the **iPod Nano** product that you created earlier and click on **Edit**.
2. Just below the product name, there are several tabs. Click on the **Attributes** tab to modify the attribute properties for this specific product.



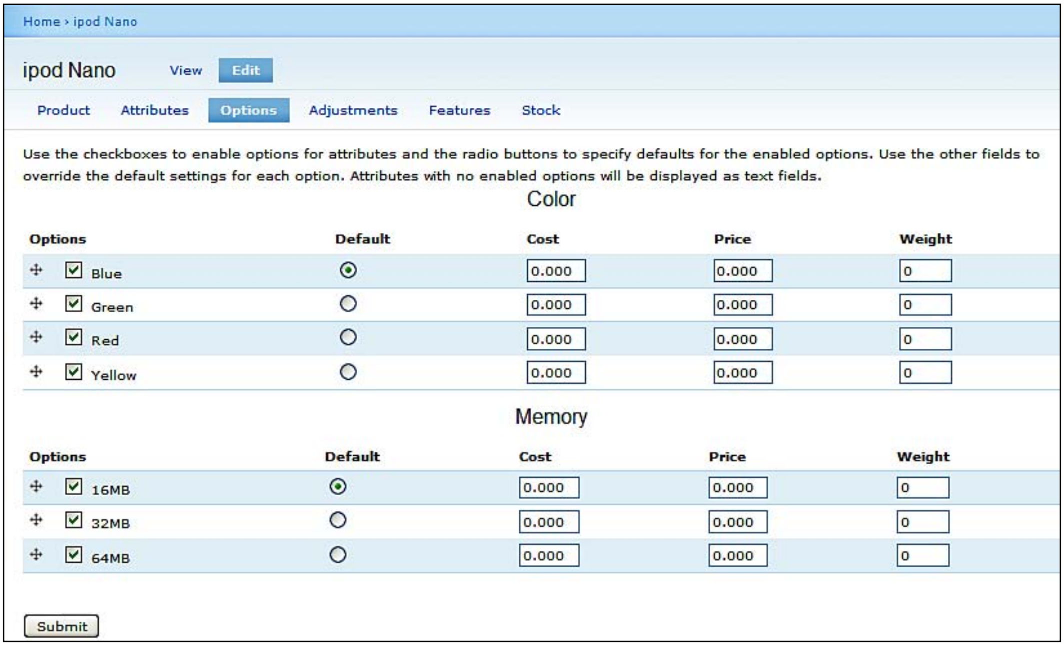
3. There are no selected attributes for this product, so click on **add attributes to this product**.



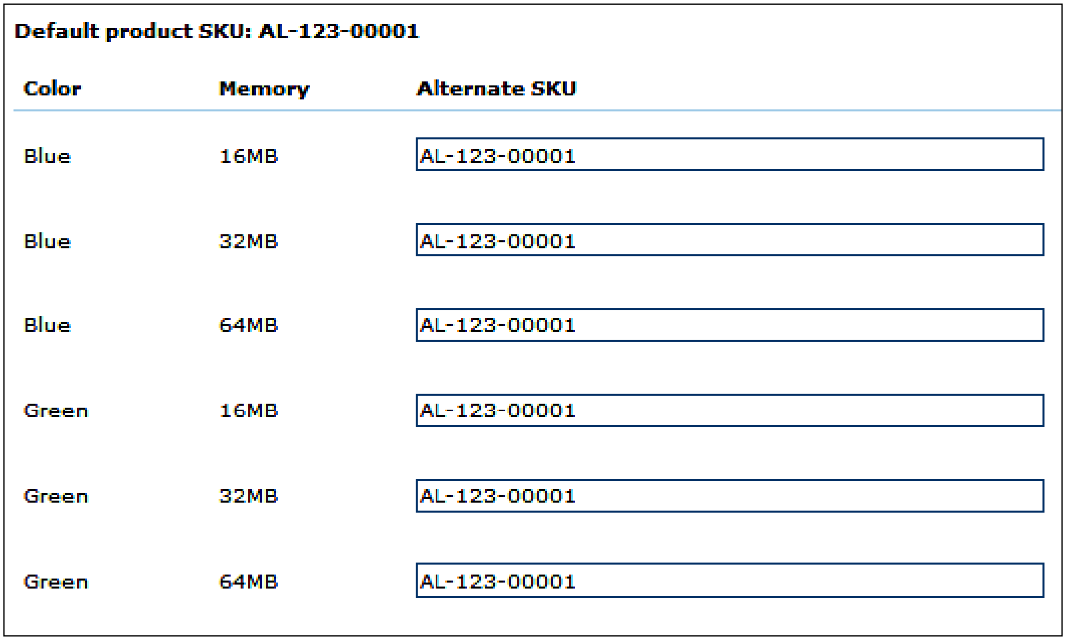
4. You can see a list of the attributes that you created at the store settings page. You can select one or more attribute for this product, holding down *Ctrl*. So, select **Color** and **Memory** and click on **Add attributes**.



5. Next, click on the **Options** tab. In the options page, each attribute has all the options selected, using the default options. Here, you can choose only specific options for your product, select the default options, and make adjustments to the cost, price, or weight for each option. For this product, just leave the default values, and click on **Submit**.

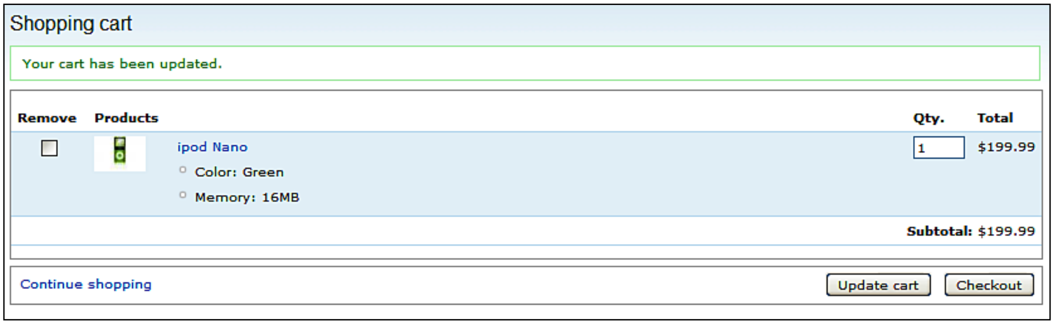


6. Finally, click on the **Adjustments** tab. Depending on how you manage your stock, you may want to give different SKUs for each specific set of options that is chosen. This gives you more information about your products, but it also adds some complexity. It's better to keep things simple, so leave the default SKUs and click on **Submit**.



7. If you select the **View** tab, you'll see that there are two new select boxes in the product page, one for the **Color** and one for the **Memory**, containing all the options that you created. If you try to add this product to the cart without selecting the color and memory, you'll get an error message, because they are required fields. So, select a value from each select box and click on **Add to cart**.

8. If you examine the shopping cart page, you'll see that below the name of the product, there are the selected options for each available attribute. This information helps the customer to remember what he or she ordered and helps the store manager to know, exactly what products he or she has to deliver.



Product classes

Before we proceed, let's study one more concept: **Product classes**—they allow you to transform almost any Drupal content type into Ubercart products. So, how this is related to attributes? In a big store, usually there are product categories with different attribute groups. If you use only the default product content type, you have to assign all the available attributes to it, and select the correct attributes for each product separately. This is time-consuming and unproductive work, so using product classes you can create custom content types, and assign to them only selected attributes.

Let's take a look how we can do this:

1. First, create a new class at **Home | Administer | Store administration | Products | Manage classes**. Enter **tshirts** as **Class ID**, and **T-shirts** as **Class name**, and click on **Submit**.
2. Now, if you click on **edit**, you can add attributes and options for this product class. Let's select only the **Color** and keep all the available colors.
3. Go to `http:/www.mysite.com/node/add/tshirts` to create a new product belonging to the t-shirts product class. After the creation of this product you'll see that it has by default only the attributes and options that we selected in stage two, so we don't need to customize this section at all.

Product classes are a powerful feature of Ubercart. If you combine them with the two most popular third-party Drupal modules, `CCK` and `views`, you can add custom fields to them. You can also create custom views and display each product class in a different way.

Managing the product catalog

After the creation or import of all your products, it's sure that you will have to make minor customizations on a daily basis. New colors are coming for a specific product, price changes are required, or you just want to add a more detailed description to a product. You have to know the most effective way to browse your catalog and manage your products. Actually, there are three ways to do this.

- **Using the catalog block**: You can browse the catalog using your categories and subcategories, find the product that you are looking for, and edit it. This is the simplest method, but it's not very fast in bigger stores.
- **Using the product browser of Ubercart**: Go to **Home | Administer | Store administration | Products | View products**. There is a list of all the products of your store. You can select a product and then edit it. It provides basic functionality and it's not very useful for a large number of products.
- **Using the content management feature of Drupal**: Go to **Home | Administer | Content management | Content**. Here, you can see not only your products, but all the content of the site, including pages, blog posts, and so on. You can filter the results by status, content type, or category and use combined filters to refine your search. This is probably the most effective way to browse your content and manage the product catalog.

Summary

In this chapter, you learned how to create categories and subcategories using the Drupal Taxonomy, and how to create products manually. Then, using the import wizard we showed you the procedure of importing products from a CSV file. Finally, we examined product attributes and gave you some ideas on how to manage your product catalog.

After we have imported and configured our online store product catalog, next on our list is to be prepared to handle shipping and packaging of our future orders. This will be the subject of *Chapter 5, Managing Shipping and Packaging*.

5
Managing Shipping and Packaging

Until now, you have carried out the basic configuration of Drupal and Ubercart, you created the categories and subcategories, and you have inserted all the products. Now, you have a fantastic product catalog with amazing product descriptions, great photos, and multiple attributes. The store visitor can spend hours browsing all these products, checking all the details, and absorbing all this information. But you didn't build this shop just to allow the site visitors to stare at the products, and you want to give them the opportunity to buy the products.

The customers can add products to their shopping cart, but the ordering process is incomplete. We need to implement two more procedures: Shipping quotes, packaging, and close supervision of these complex and important procedures. In this chapter, we'll help you to learn how to configure them. So, in this chapter we will learn to:

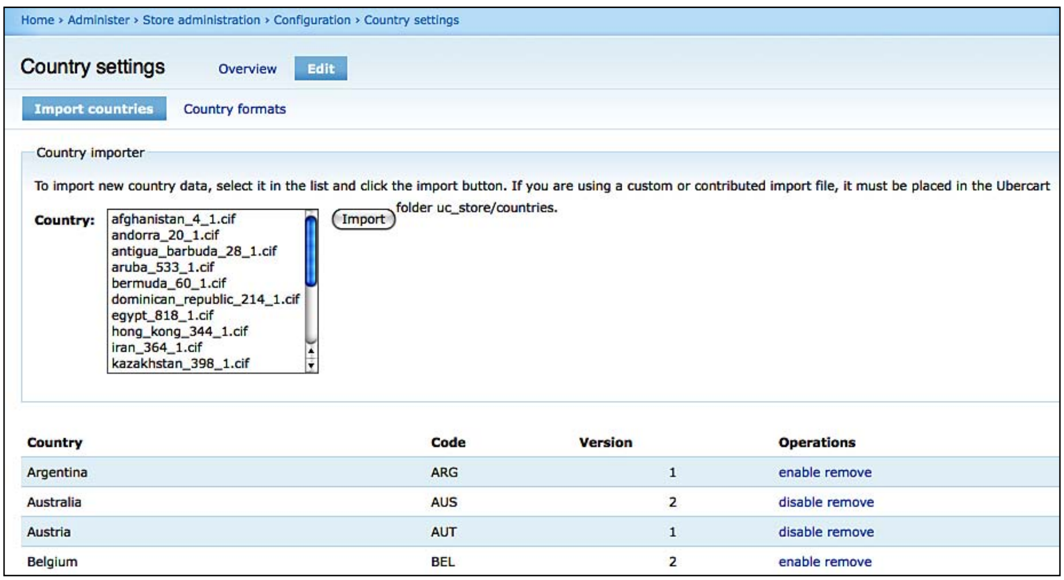
- Create and configure product shipping quotes
- Administer shipping and choosing your courier service
- Create shipping quotes

Sending a product from your company to the customer's door and charging him or her for this seems like a straightforward procedure. But when you start to analyze all the options and parameters of it, you realize that it can become one of the most complex procedures of your store. We have to build shipping quotes that calculate the cost with accuracy, based on the customer's choices. As we want to provide top-class service, and we try to have only satisfied customers, we have to work very carefully in this stage and try to avoid any errors.

Now the good news: usually, you make this customization only once, when you are building your store. After that, you need to make small alterations only when you start to ship to a new country, or when you want to change the shipping charges.

Deciding where to ship products: Country settings

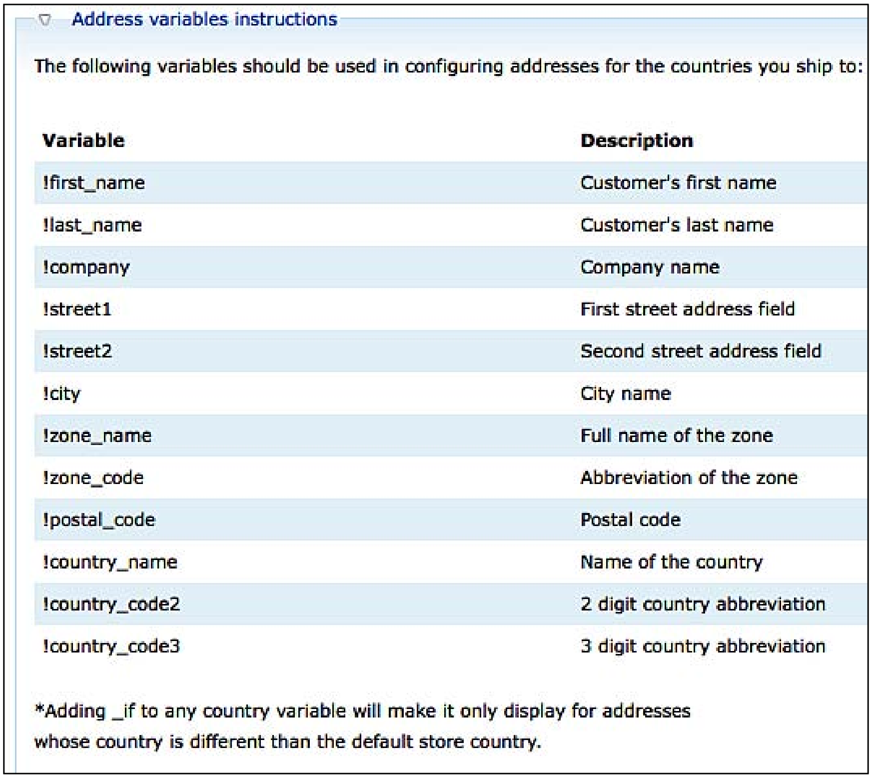
Go to **Administer | Store administration | Configuration | Country settings**. Country settings have two tabs, **Import countries** and **Country formats**. The country importer allows you to select the countries from which you are going to accept orders. The default countries are US and Canada. The interface allows you to import new countries, and disable or delete the existing countries.



If you want to import a new country, you first have to create a Country Import File (`.cif`) with all the country details, and copy it to the `ubercart/uc_store/countries` directory on your server. A `.cif` file contains information about the regions of the country and the address format. Ubercart already contains `.cif` files for about 65 countries. As the Ubercart community is very active, new countries are added constantly, so if the country you're interested isn't on the list, check the **Contribution** section in `http://www.ubercart.org`.

If you want to create a `.cif` file by yourself, the best way is to copy a file for another country, study it carefully, and make all the required corrections. Before you start, you have to know the ISO 3166 codes for this country, found at `http://en.wikipedia.org/wiki/ISO_3166-1`. If it is necessary, you can then add all the regions of the country and change the address format. If you put the file in the `ubercart/uc_store/countries` directory on your server, you'll see it it in the import list.

The **Country formats** section allows you to alter the address variables of every imported country. The first collapsed box shows all the available variables, and instructions about them. After that, there is a collapsed box for each country that allows you to make detailed customizations.



Address variables instructions

The following variables should be used in configuring addresses for the countries you ship to:

Variable	Description
!first_name	Customer's first name
!last_name	Customer's last name
!company	Company name
!street1	First street address field
!street2	Second street address field
!city	City name
!zone_name	Full name of the zone
!zone_code	Abbreviation of the zone
!postal_code	Postal code
!country_name	Name of the country
!country_code2	2 digit country abbreviation
!country_code3	3 digit country abbreviation

*Adding _if to any country variable will make it only display for addresses whose country is different than the default store country.

Understanding and configuring conditional actions

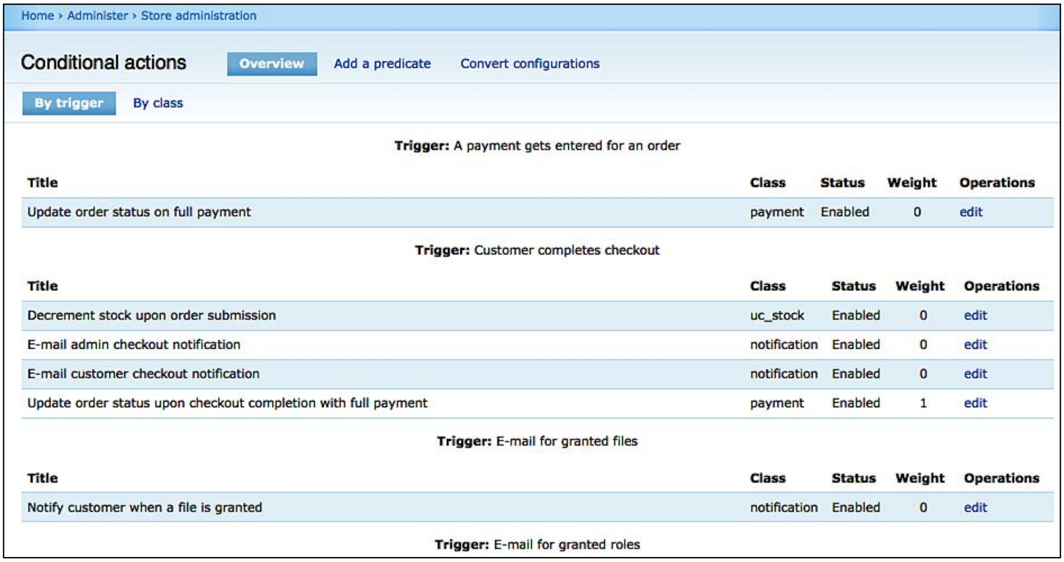
Conditional actions is an Ubercart core module, and is required to create shipping quotes and tax rules. It's also used to update order status automatically, when a payment is received, and to send automated notifications. When you ship products to different regions of the world, or you have different shipping charges, depending on the total weight, you need to implement some business logic in your store. Then, the store can automatically decide what happens when you ship X products to Y region with Z weight.

A conditional action has three elements: a trigger, the conditions, and the actions. This combination is called a **predicate**.

A trigger is actually an event and it's usually enabled when the customer completes an operation in the store. During the second step, the system checks the conditions and, depending on them, decides what action it has to perform. In the final step, the system completes one or more action, and gives feedback to the customer or the administrator about the results.

For example, let's consider that we create the following conditional action: for orders over $100, shipping is free, but for orders under $100, shipping rate is $10. The trigger is enabled when the customer decides to complete the order and goes to the checkout page. Then, the store checks the conditions: is the order over or under $100? Depending on the conditions, our system automatically makes the right decision, and makes the right action. It calculates the shipping rate and informs the customer about the order total.

The main advantage of conditional actions is that you can build all the logic without writing even a single line of code. You just have to spend some time getting used to the user interface. If you go to **Administer | Store administration | Conditional actions**, you can see an overview of all the default predicates, which are created during Ubercart installations.



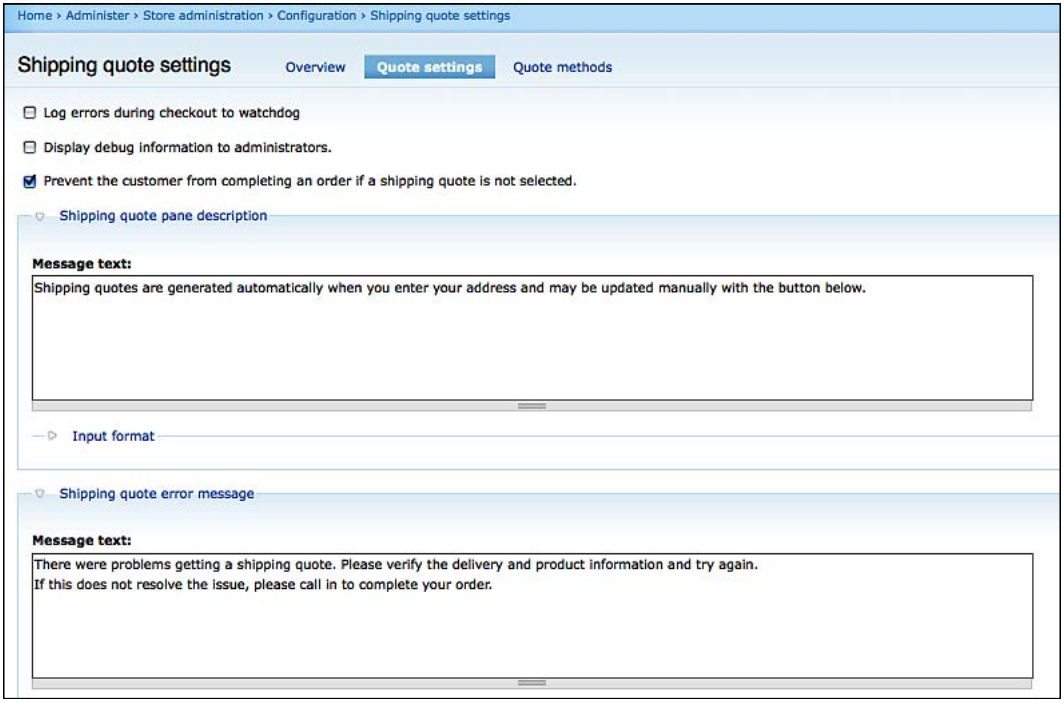
You can edit them to change the basic information, the trigger, the conditions, and the actions, or you can create a new predicate.

Shipping quote settings—overview

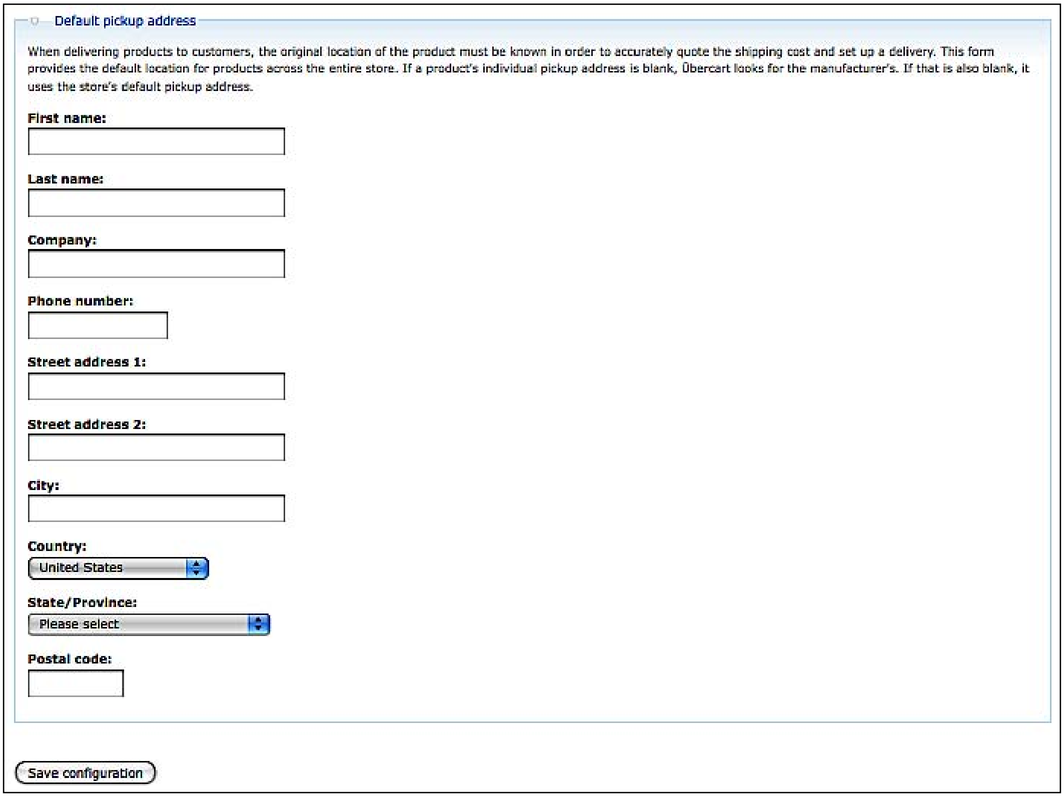
To administer the shipping quote settings, go to **Administer | Store administration | Configuration | shipping quote settings**. This page has three tabs:

- **Overview**: This first tab is just a quick overview of the shipping quote settings and shipping quote methods.
- **Quote settings**: In the quote settings page, we can do some basic customization of the shipping procedure of our store. At the top of the page there are three checkboxes followed by some other settings:
    - The first two, **Log errors during checkout to watchdog** and **Display debug information to administrators** are useful to developers, when they want to check whether there are any errors in the checkout process. We don't need this functionality, so leave them unchecked.

- The third option, **Prevent the customer from completing an order if a shipping quote is not selected** is obvious. We leave this checked, thus forcing the customer to select a shipping quote at the checkout page.

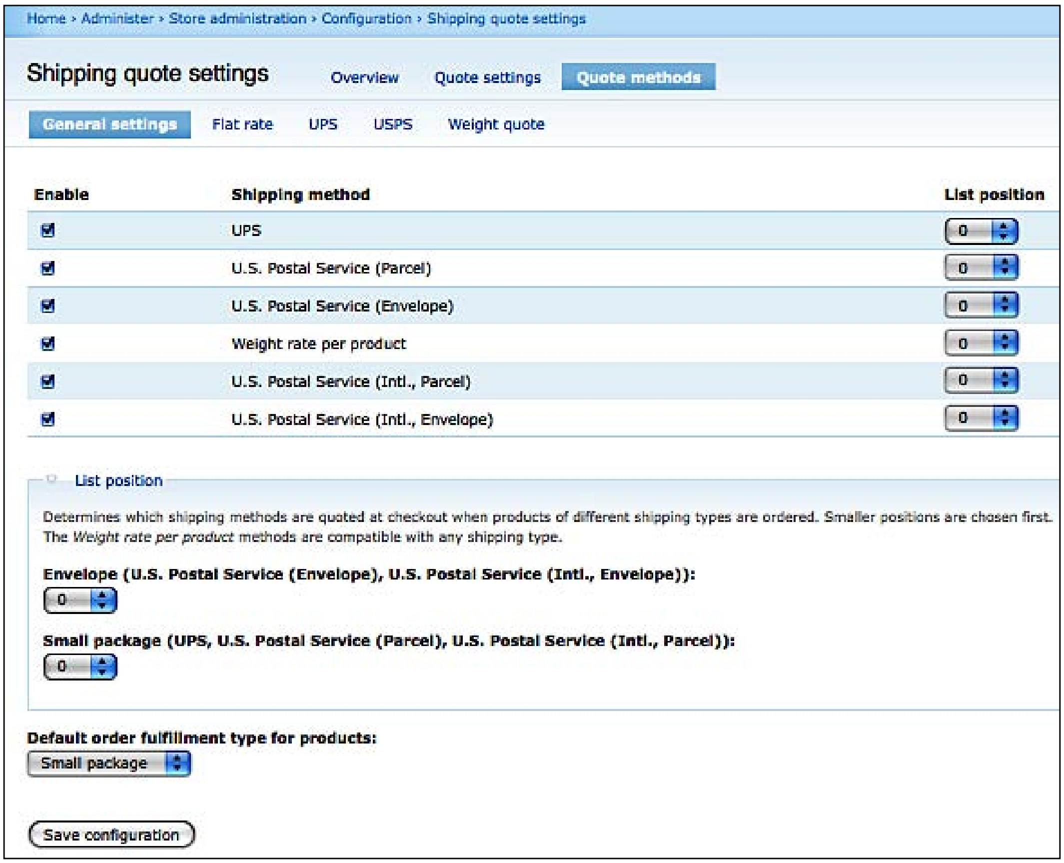


- The **Shipping quote pane description** allows us to write a custom description for the shipping quote of the checkout page. There is no need to change anything here.
- The **Shipping quote error message** shows the error message that is presented to the customer when something goes wrong during the calculation of the shipping quote. It's a good idea to add a phone number or an e-mail address here, giving the opportunity to your customers to contact you immediately, and keeping them from abandoning their cart.
- The **Default pickup address** is necessary in order to calculate the shipping cost, especially when you cooperate with transport companies. So carefully complete your name, company name, phone number and address, and click on **Save Configuration**.



- **Quote methods**: This page starts with the **General settings** section. There, you can see all the available shipping quote methods, enable or disable them, and determine their list position. Ubercart has four available shipping modules, grouped under the name **fulfilment** in the modules page.
    - **Flatrate**: Assigns a flat shipping rate to products
    - **U.S. Postal Service**: For integration with US Postal Service rates
    - **UPS**: For integration with UPS rates
    - **Weight quote**: Assigns a shipping rate to products based on weight

If you cannot see all the four shipping quote methods, please go to the modules (**Administer | Site building | Modules**) page, and enable them. Then return to the quote methods page, enable them too, and click on **Save configuration**. Of course, when your store goes live, you don't have to use all these methods simultaneously, because probably you would confuse your customers. We are going to show them all to you and then you'll have to decide what fits best to your needs.



Now, at the top of the page, there are four more tabs, one for each available quote method. We are going to explore all of them one by one in the following sections.

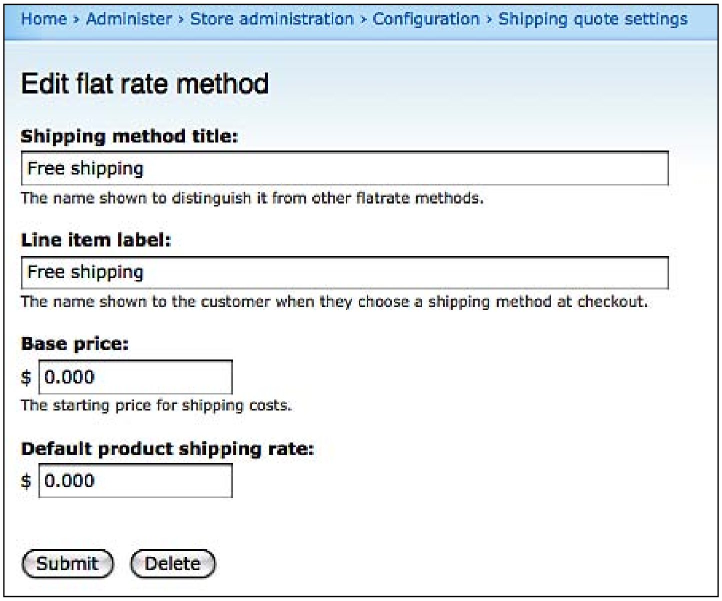
Flat rate

Click on the **Flat rate** tab. There are no flat rates created yet. We want to create a standard shipping method, charging $10 per order and $1 per item. We also want to create a free shipping method, for orders over $250.

To do so, carry out the following steps:

1. Click on **Add a new flat rate shipping method**.
2. The **Shipping method title** is the name shown to the administrator. Enter **Free shipping**.
3. The **Line item label** is the name shown to the customer during checkout. Enter **Free shipping**.
4. The **Base price** is the starting cost for the calculation of the total shipping cost. Enter **0**.
5. The **Default product shipping rate** is the default shipping cost for every product of our store. Enter **0**.

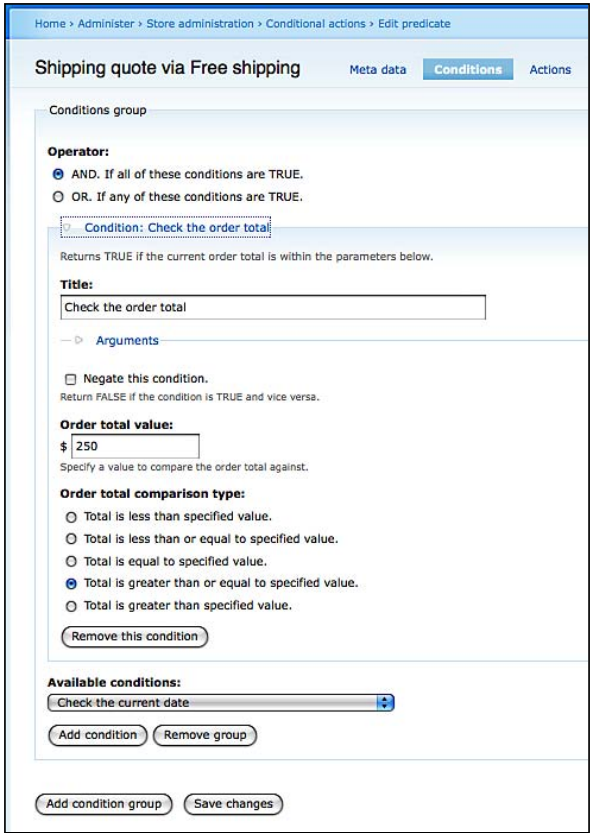
6. Click on **Submit** to create the first flat rate method.



Now, again click on **Add a new flat rate** to create the second flat rate method. Enter **Standard Flat Rate**, **Standard Flat Rate**, **10**, and **1** in the four fields, and click on **Submit**.

So, now that we have created the two flat rates, how can we determine which one is activated, depending on the order total? That's exactly the point where we need conditional actions to proceed. Click on **conditions** for the **Free shipping** method to configure them.

If you check the breadcrumb trail, you'll see that we've actually been transferred to the conditional actions settings page. In this page, by following simple rules, we can activate the free shipping method if the conditions that we set are met. In our example, we are going to set only one condition, but we can implement more complex logic by creating multiple conditions, and organizing them in condition groups. So just click on the **Available conditions** list, select **Check the order total**, and click on **Add condition**. In the new fields that open, enter **250** in the **Order total value**, and select **Total is greater than or equal to specified value** for the **Order total comparison type**. Click on **Save changes**.



For the **Standard Flat Rate**, we are going to follow exactly the same steps, with only one difference. We want to activate this method only when the order total is less than $250. So, click on the **Available conditions** list, select **Check the order total**, and click on **Add condition**. In the new fields that open, enter **250** in the **Order total value**, and select **Total is less than the specified value** for the **Order total comparison type**. Click on **Save changes**.

> For every new flat rate that we create, a new field is created in the **Edit products** page under the **flat shipping rates** group. There we can override the default settings, and set a different shipping rate for every product.

Our flat rates are set up and are working as planned. If you do some tests, you'll see in the checkout page that for orders over $250, only the free shipping choice is available, and for orders less than $250, only the standard flat rate choice is available. Well done!

UPS

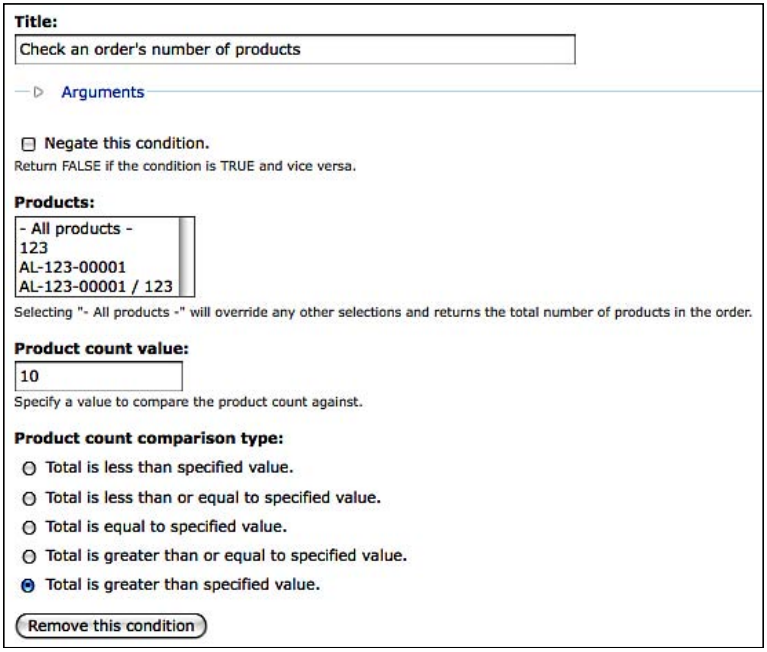
The **United Parcel Service** (**UPS**) shipping method is a very powerful tool, but you have to spend some time on it. If you click on the UPS tab, you'll see that there are 15 different fields to complete. You'll also have to open an account in the UPS site. Now, you have to make a decision here. If you decide to have a simple shipping rate policy, you can create a group of flat rates. As you just saw, it's a very simple procedure, as it allows you to build a flat rate in a few minutes.

Keep in mind though that flat rates have a big disadvantage because they are not accurate. The transfer companies usually won't charge you with a flat rate; they'll charge you with a different rate every time, depending on the size or the weight of the package and the final destination. This means that even if you calculate a "shipping rate average" and you use this for your flat rates, sometimes you'll have a profit from this process, but sometimes you'll have a loss.

The truth is that nowadays most companies use the flat rate solution, because it's simpler, easier to implement, and it doesn't break the operation of your site when it cannot communicate with the UPS site. But if you want to be 100 percent fair to your clients and to be able to calculate precise shipping rates for every kind of package, we believe that you should invest some time, and enable the UPS method.

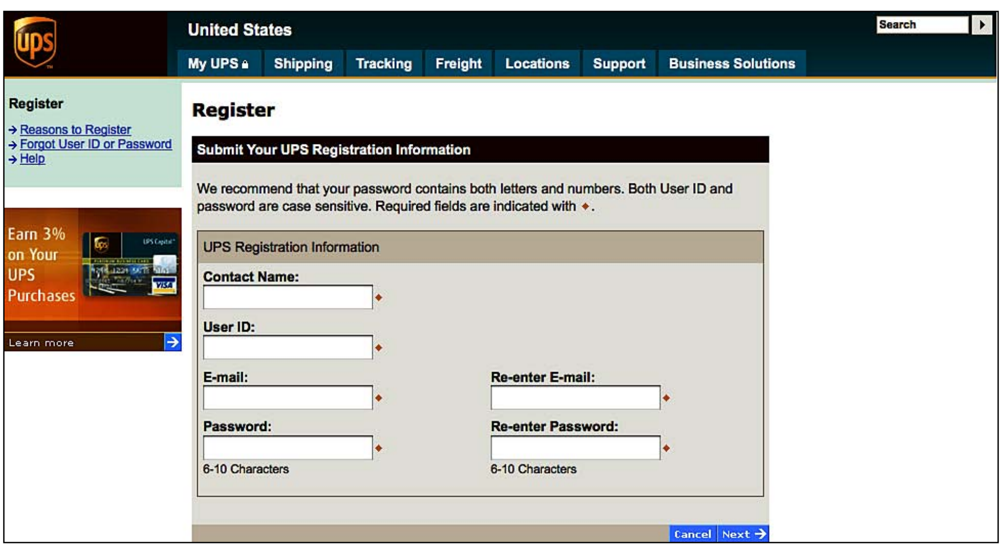
Let's see together what the required steps are:

1. Click on **Set the conditions that will return a UPS quote**. This will open the conditional actions dialog. As we saw in the *Flat rate* section, here you can customize the conditions under which this method will be enabled. For our example, we want to activate this method when the customer has more than 10 items in his or her order. Of course, you can use the same method to activate it for specific countries or for specific dates. So, from the **Available conditions** list select **Check an order's number of products** and click on **Add condition**. In the **Product count value** enter the value **10**, and from the **Product count comparison type**, select **Total is greater than specified value**. Now click on **Save changes** to store the conditions for this quote.

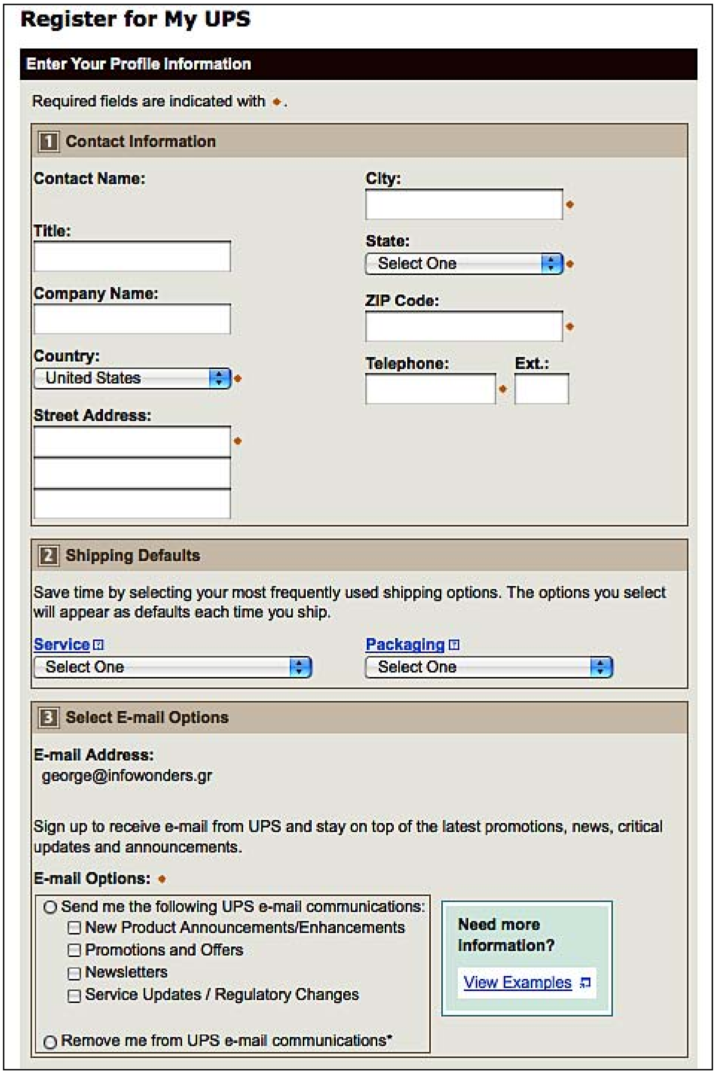


2. The next step is to set up an account with UPS. To do so, carry out the following steps:

   i. Go to `http://www.ups.com/e_comm_access/gettools_index`. Click on **Register**. This opens a new page where you have to complete your account details. When you finish, click on **Next**.



ii. In the next page, you have to complete your **Contact Information**, **Shipping Details**, **E-mail Options**, **Payments Defaults**, and your **Primary Role** in the company. After completing these fields agree to the terms and conditions of use. Click on **Next** to complete the registration. Soon you'll receive an e-mail, confirming your registration.



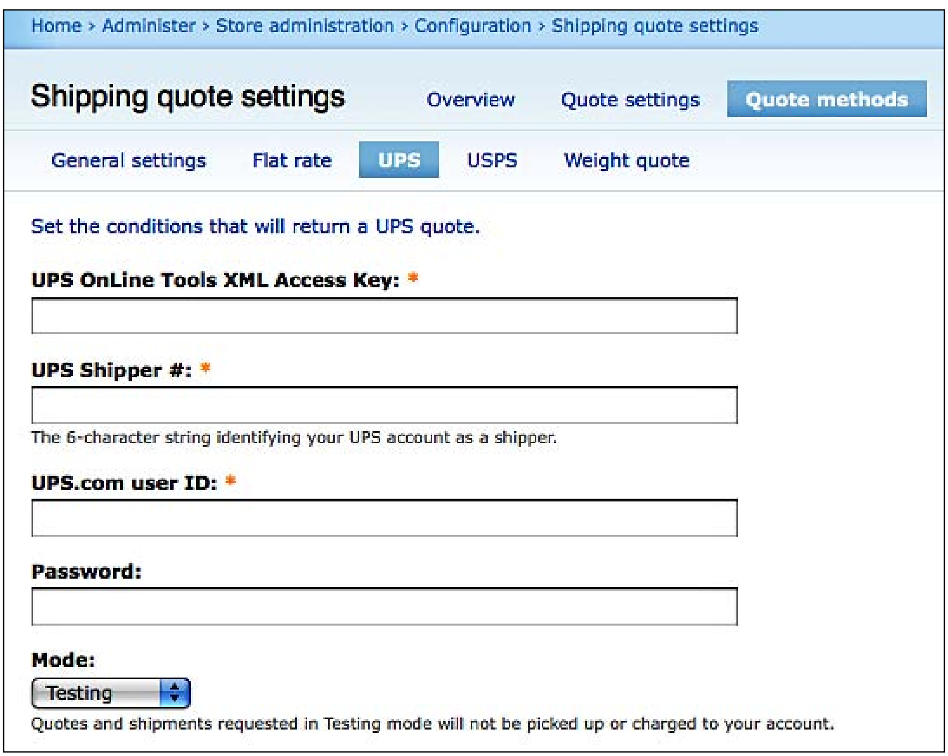
iii. Unfortunately, this is not enough, we also need an XML access key. This allows our site to have access to and receive data from the UPS site.

iv. Click on **Log-In** and enter your **User ID** and **Password**.

v. Apply for a developer's key and then request an XML access key.

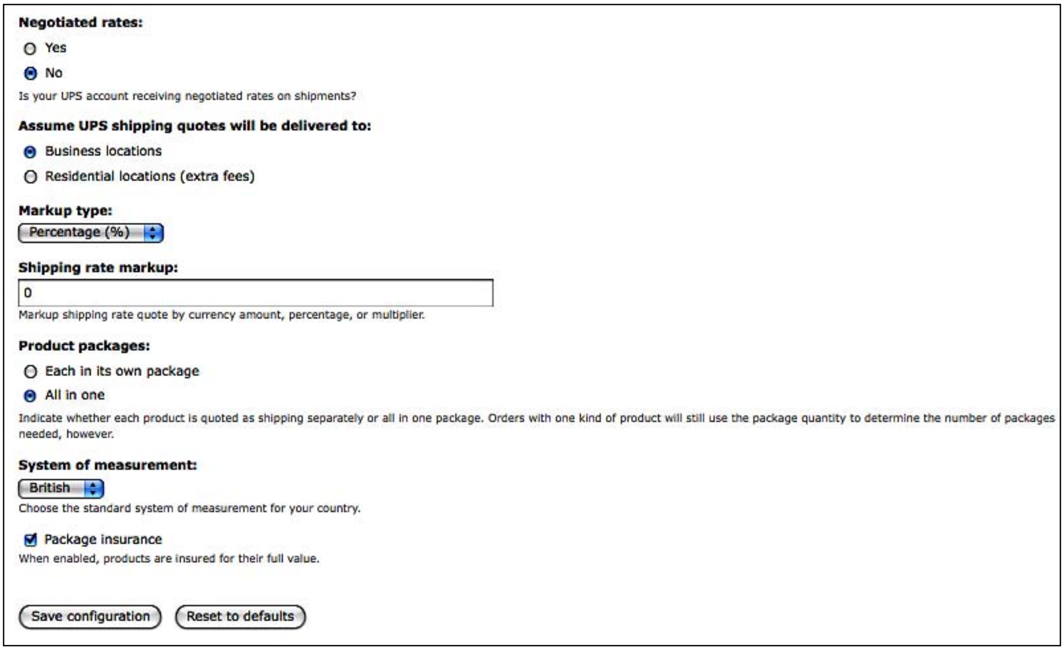
Now, let's go back to our store settings and start completing the required fields.

3. In **UPS OnLine Tools XML Access Key** insert the XML access key, as you received it from UPS.
4. In the **UPS Shipper #** enter the six-character string that identifies your UPS account as a shipper.
5. **UPS.com user ID** is the user name that you entered when you created your account at `http://www.ups.com`.
6. **Password** is the one that you entered when you created your account at `http://www.ups.com`.
7. The **Mode** field has two options: **Testing** and **Production**. During the development phase, we are going to use the testing mode. When we finish our tests and we are sure that everything is working perfectly, and we don't receive any errors, we can switch to the production site. When you do it, pay attention, because there will be no more room for tests. For every order that you submit, you'll be charged by UPS, so be careful!



8. UPS offers a huge selection of transport services. From the **UPS services**, select the services that you want to offer to your clients. It's of course possible to include all the services, but it's better to keep things simple and select only four or five choices. Experience shows that too many choices don't help the clients, they actually confuse them.
9. **Pickup type** presents **the various pickup options that UPS offers. Select the** one that is more convenient for your company. Consult your local UPS office for information regarding the various pickup types and their cost.
10. The **UPS customer classification** also affects the total cost of shipping. This option is related to the previous one. For daily pickups the default is **Wholesale**, for customer counter pickups the default is **Retail**, for other pickups the default is **Occasional**.
11. In the **Negotiated rates**, select **Yes** if you are able to negotiate your UPS rates; if they are standard, just select **No**.
12. In the **Assume UPS shipping quotes will be delivered to** field, you have to choose between **Business locations** and **Residential locations**. For the second choice, you'll be charged with an extra fee.
13. The **Markup type** and **Shipping rate markup** fields allow you to define how you're going to mark up the shipping rates that UPS charges you. For example, if we want to mark up the shipping rates by 20 percent, we select **Percentage** at the first field, and **20** at the second.
14. In the **Product packages** field, you have to select whether you're going to ship each item separately, or all in one package. If you make a big package with all the items of your order, the shipping rate is going to be lower.
15. In the **System of measurement** field, select whether you prefer the **British** or **Metric** system depending on your country's standards.

16. Select **Package insurance** if you want the products of your shipment to be insured.

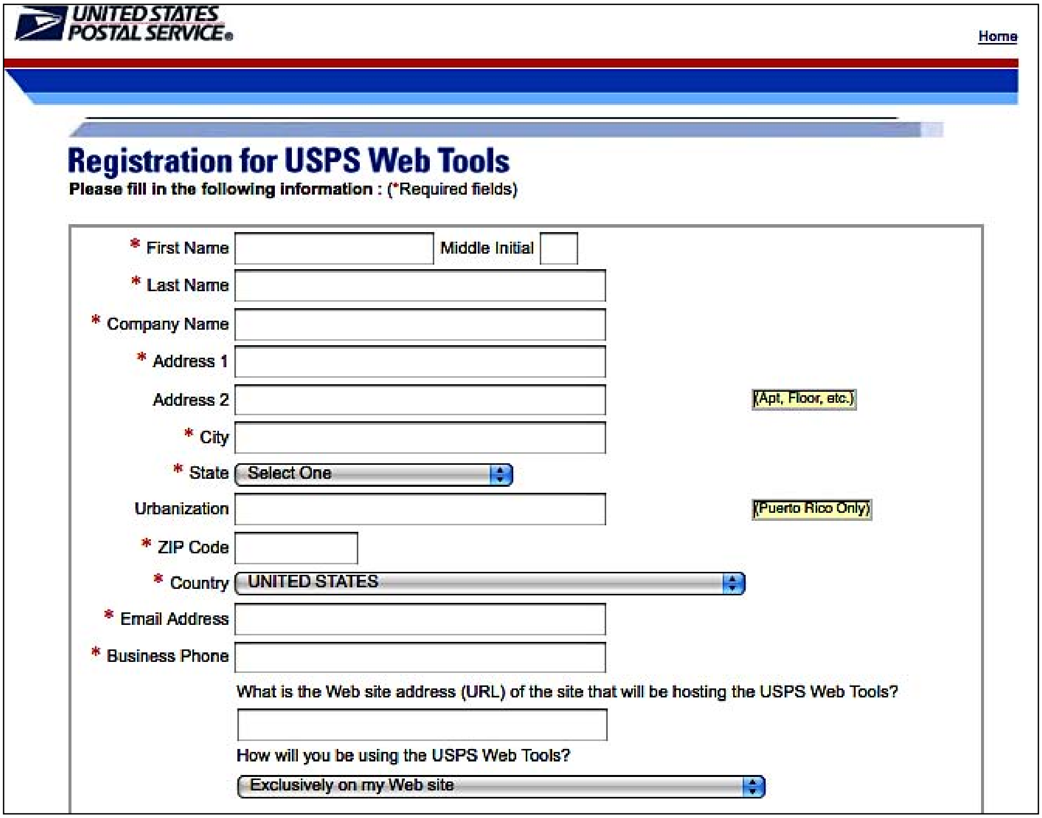


17. Finally, click on **Save configuration** to save the UPS shipping quote with all your adjustments.

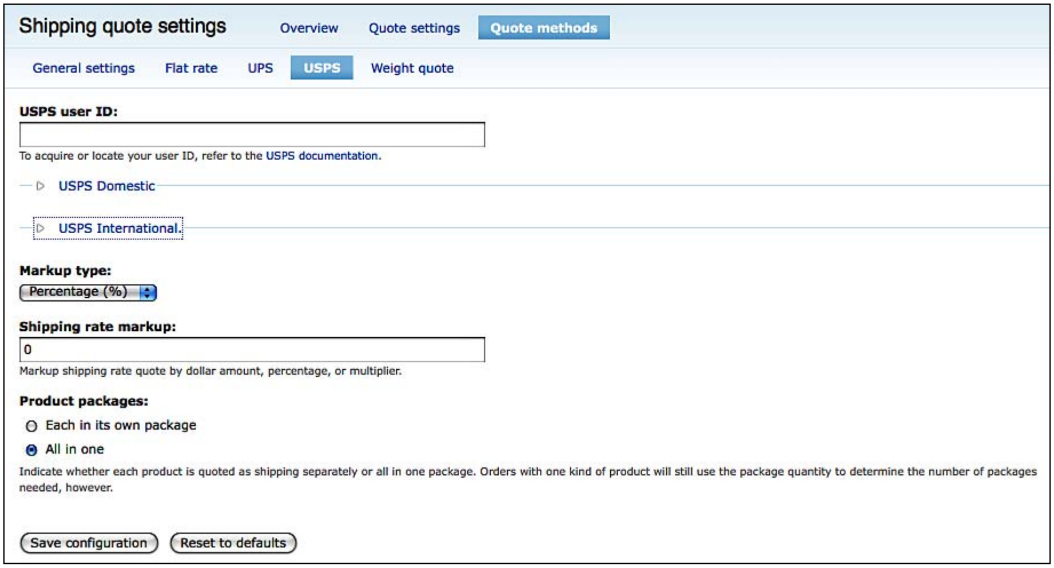
USPS

The **United States Postal Service** (**USPS**) quote works similarly to the UPS quote that we examined in the previous section. Actually, it's much simpler as it has fewer options, but you still have to follow some required steps:

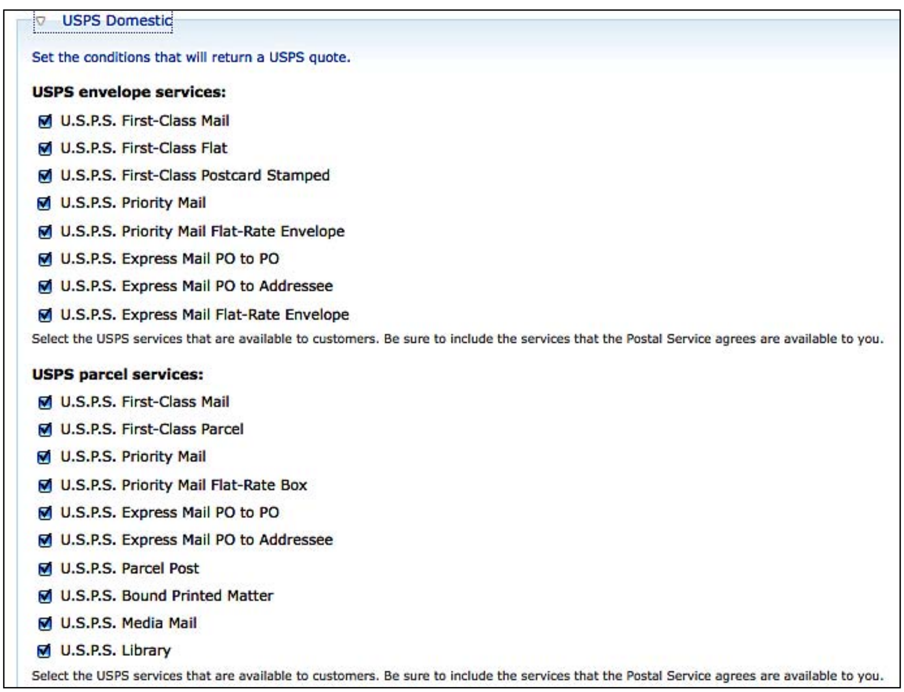
1. Go to `https://secure.shippingapis.com/registration/` to register for the **USPS Web Tools**. Soon you will receive an e-mail with a user ID and a password.



2. Send an e-mail to `icustomercare@usps.com` and ask access to their production server, indicating that you're using Ubercart.
3. Now it's time to complete the fields in the USPS configuration page of our store. First, insert your **USPS user ID**.



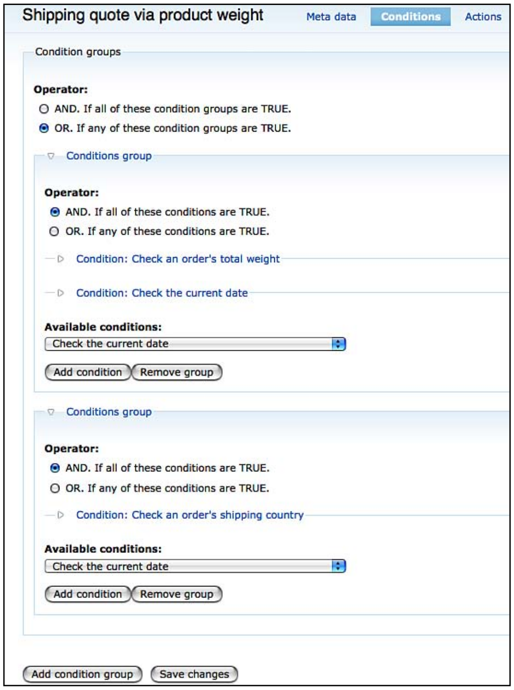
4. Click on **USPS Domestic**, and select the appropriate **envelope services** and **parcel services**.
5. Do exactly the same for **USPS International**.



6. The **Markup type** and **Shipping rate markup** fields allow you to define how you're going to mark up the shipping rates that USPS charge you. For example, if we want to mark up the shipping rates by 20 percent, we select **Percentage** at the first field and **20** at the second.
7. In the **Product packages** field, you have to select whether you're going to ship each item separately, or all in one package. If you make a big package with all the items of your order, the shipping rate is going to be lower.
8. Finally, click on **Save configuration** to save the USPS shipping quote with all your adjustments.

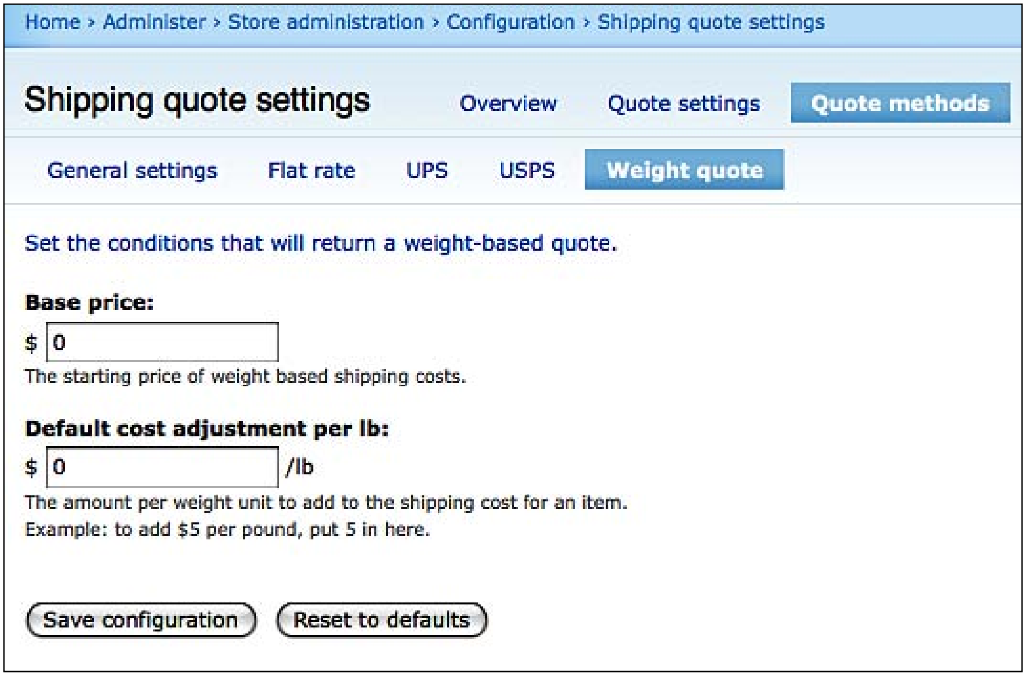
Weight quote

The final method that we are going to examine is the **Weight quote**. Click on **Add a new weight quote shipping method**. This will open the conditional actions dialog. Here you can set the conditions under which the weight quote will be enabled, following the same procedure that we showed you in UPS quote method. You can create one or more condition groups, each one of them containing one or more conditions, connected with an AND or OR operator. Using and combining conditions such as the current date, the total weight of the order, or the shipping country, among others, you can build complex system logic and define exactly when the weight quote will be enabled.



Then select a **Base price** and a **Default cost adjustment per lb**. The total shipping cost will be calculated using the following formula:

Shipping cost = Base price + (Default cost adjustment per lb * package weight in lbs).



Now click on **Save configuration** to store the conditions.

Summary

In this chapter, we have discussed all the best practices and workflows that should be followed in order to achieve higher efficiency for your business back-office administration. Being consistent on your deliveries and your packaging policies will help you build a reliable profile among your customers and always provide high service standards. Keep in mind that shipments is an area most online stores have problems with, so opt for solid returns and lost item policies in advance.

Furthermore, always make sure that you have communicated all the special or additional shipping and handling costs to your end customers, as this will help you build a fruitful professional relationship and help you in your future loyalty programs.

In the next chapter we will discuss shipping taxes and online payments, two very important issues that affect not only the convenience of your clients, but the safety of your store as well.

6

Managing Taxes and Payments

We have reached the point where we can start selling our products and collect payments. This is a turn-key feature of our online electronics shop, as we can now successfully launch our business, utilizing direct online web payments from our customers. This is the point where the game gets serious and we should become cautious about our next steps.

Reliable online payments will be the basis for our reputation management, so that each visitor should be absolutely confident; the visitor's confidence depends on a fault tolerant and trustworthy system. Customers have an intrinsic physiological barrier against providing their credit card number online, so you are here to persuade them that you have a secure and state-of-the-art online shop that can be trusted and let them focus on your merchandise alone.

So in this chapter, we will discuss how to:

- Manage tax rates and settings
- Configure automatic payment gateways
- Create your own payment gateway from scratch

Managing tax rates and settings

In this section we will configure tax rates to which our products are subject. Moreover, we will go through all the extra rate settings, such as shipping insurance or early shipping fees.

Basic tax rate configuration

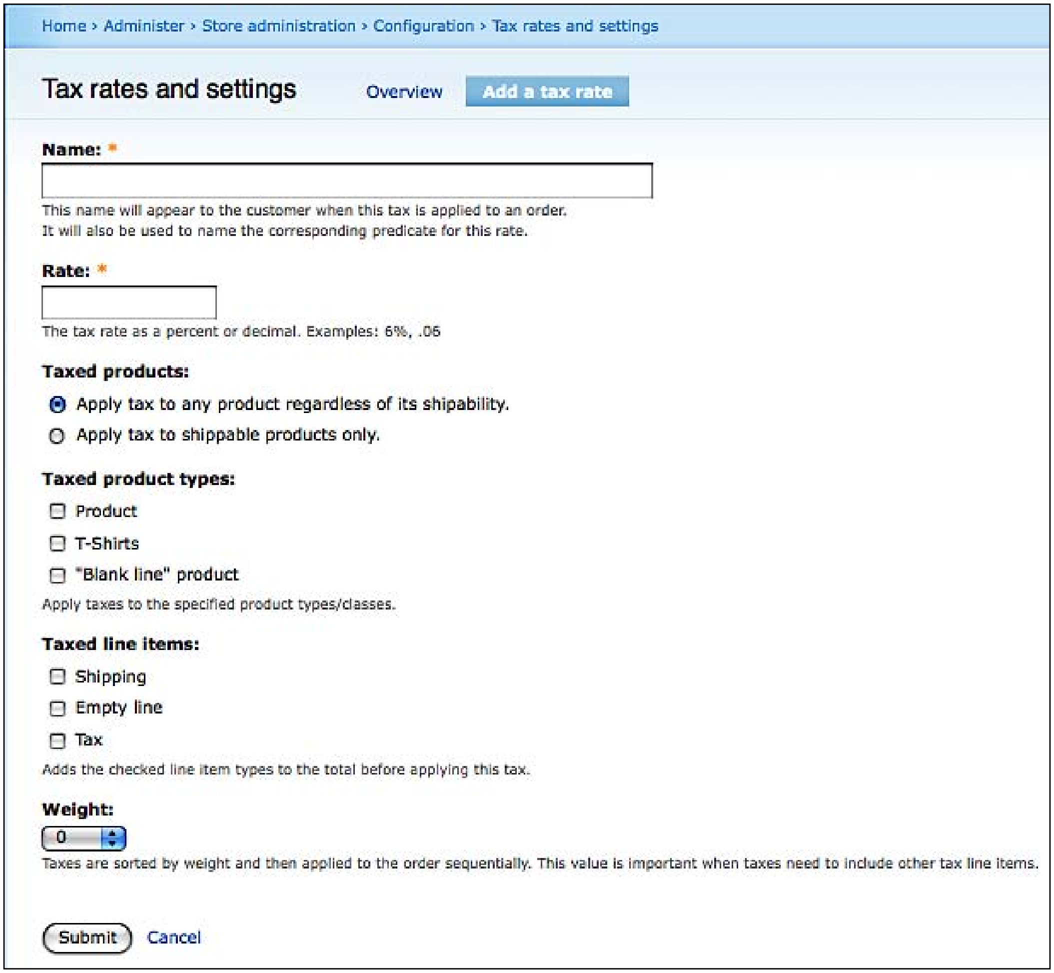
By now you have a fully functional online shop, very competitive prices for your merchandise, and a business plan that predicts orders from all over the world. You also went through all the shipping configurations, so the next step you have to take is to start dealing with tax rates. When it comes to setting up rate calculation for your online store, you should take into account a great many factors that can integrate with your existing business logic. Fortunately, Ubercart from version 2.0 onwards, provides this functionality in its core, encapsulating these configuration settings in Ubercart conditions, a powerful and extensible tool for integrating business logic. This way you can handle complex state sale taxes based on customers geographic locations (billing address), shipping date, or the type of products shipped. You will also be able to make dynamic taxation procedures based on either shipping country or customer's billing country.

Adding a new tax rate

After having concluded the various rates and taxes for each region, you plan on providing shipping services, so you should start the configuration of these tax options. In the United States, tax policies are very diverse and even in the same state goods can be subject to different tax policies!

> Before you start your shop tax configuration, make sure to put all your taxation data in the following form:
>
> Country, State, Region, Postal Code, Percentage
>
> Also take a note to spend some time on a little research. Valuable data can be found on tax office sites and e-government portals all around the world. In addition, try discussing each individual area concerning the features you wish to provide your clients, such as shipments and bill, with your tax consultant.

When you have all the supporting tax charges, you will have to configure Ubercart tax administration to comply with your shop's policies. Go to **Home | Administer | Store administration | Configuration | Tax rates and settings**. On the overview screen you can see some policies already configured and you can also add new ones. Click on **Add a tax rate**; you will see the tax rate creation screen.



Here, you should give some self-explanatory name to each tax policy, which will help you to review and change them in the future. After that you have to give a value for percentage that will be added as tax. Here, you can notice that you can assign specific tax rates for each product category, something very useful, as cases like this are pretty common in tax legislation. Product categories such as books and primary need consumables are subject to lower tax costs than electronics for instance.

After creating a new tax policy, you should proceed to configuring the conditional actions associated with this particular policy. On the tax administration screen, click on the **Conditions** link next to the tax definition you just created.

You are now on the tax condition configuration screen where the observant reader should notice that actually the edit environment is the same as the condition statements, discussed in the previous chapter. Note that you can configure complex policies with either of the conditional logics: AND or OR.

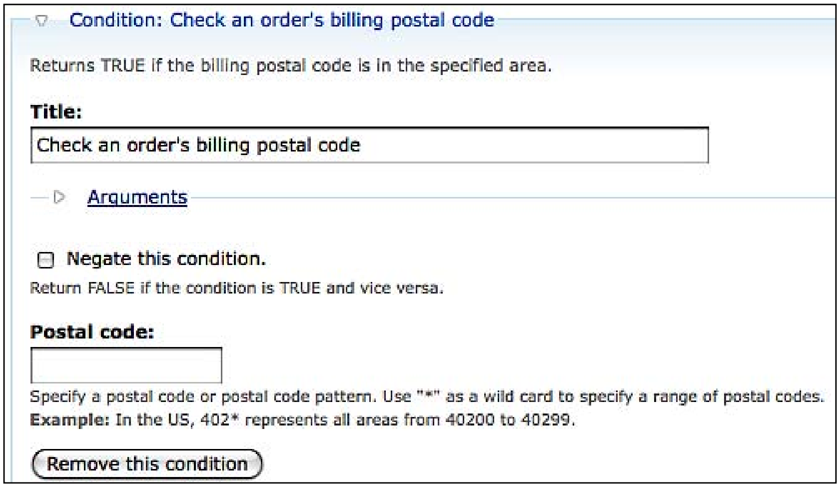
Having all the flexibility that the action module provides to configure our tax policies we can go with some of the options provided in the drop-down menu. There are some actions that are of particular interest in tax calculations, such as **Check the current date**, **Check an order's billing State/Province,** and so on.

Select **Check an order's billing State/Province** and click on the **Add condition** button. A new options group will be displayed, so click on it to expand it. As you can see, all the possible countries with the corresponding states or administrative regions are displayed. This way you do not need to research or perform complex data entries for each region individually. Select the region to which you want to associate the tax policy you just created, and click on **Save changes**.

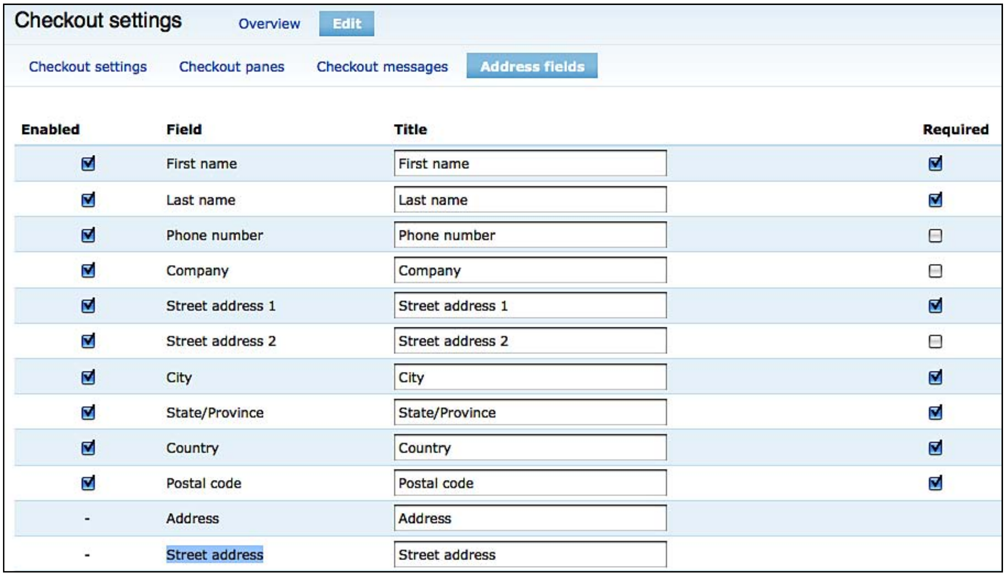
> Do not forget that you can also clone your configured tax policies. So, you can save time and avoid replicated actions.

Congratulations! You have created your first tax policy for your online shop. You can continue until you finish configuring all the other policies you have collected.

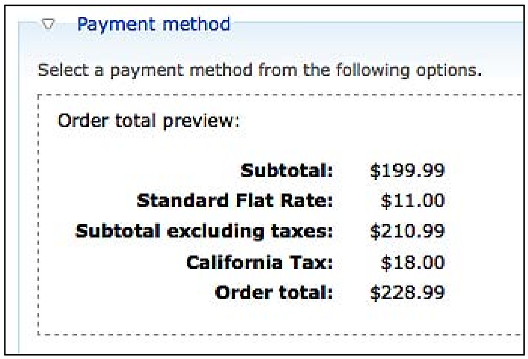
Other useful policies are based on country or postal code as displayed in the following image:



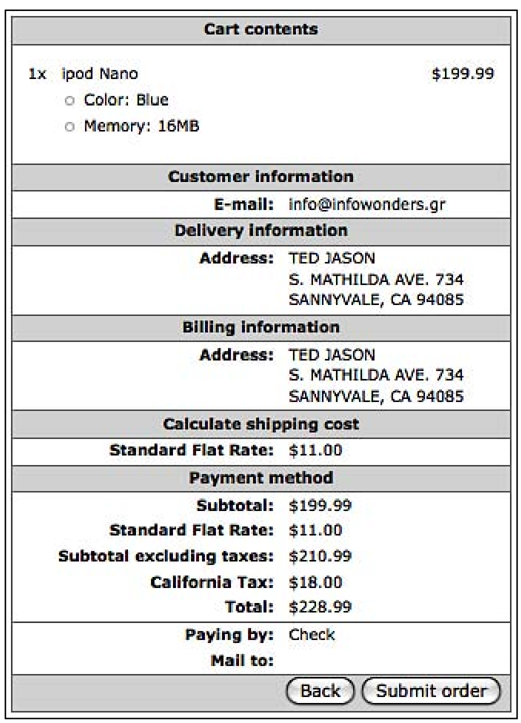
While configuring the tax policies for your store, you should also consider the type of data you will have for the tax calculation procedure. In particular, if you want to configure postal code-aware tax policies, make sure you make the appropriate adjustments on the **Checkout settings** page. You should check all fields you rely on for your tax calculation as **Required**. To do this, go to **Home | Administer | Store administration | Configuration | Checkout settings | Address fields**, and check all the mandatory fields as **Required**.



After the setup of these tax rate policies, you should go and test your configuration. Ubercart taxes are completely integrated with the checkout process, so information for taxes and extra charges will be available to the user in each step for the payment process. When customers provide their billing information, payment description is updated automatically, providing information about the taxation charges for the order.



Eventually, upon order fulfillment, data is added to the final order in the correct sequence, and the electronic invoice is issued by your online shop.

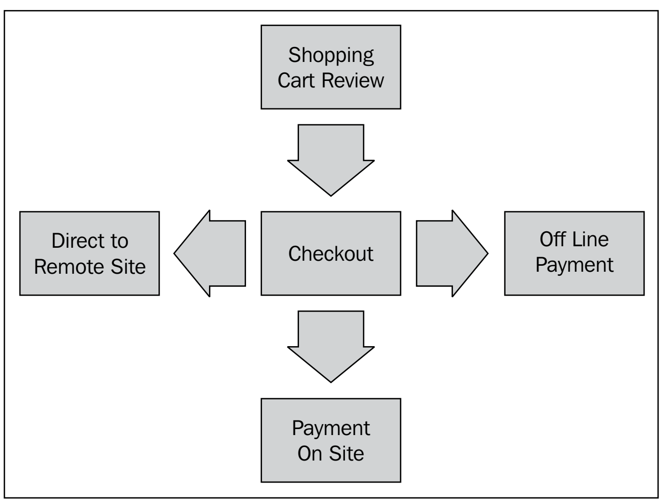


Payment settings

The next step we should go through before releasing our store to the public is the payment gateway configuration. We will discuss the constraints and precautionary measures that one should keep in mind when accepting and storing credit cards information for an online store. We will also discuss the benefits that Internet sales provide. At the end, we will provide step-by-step configuration processes for two of the most popular payment gateways, namely PayPal and Google Checkout.

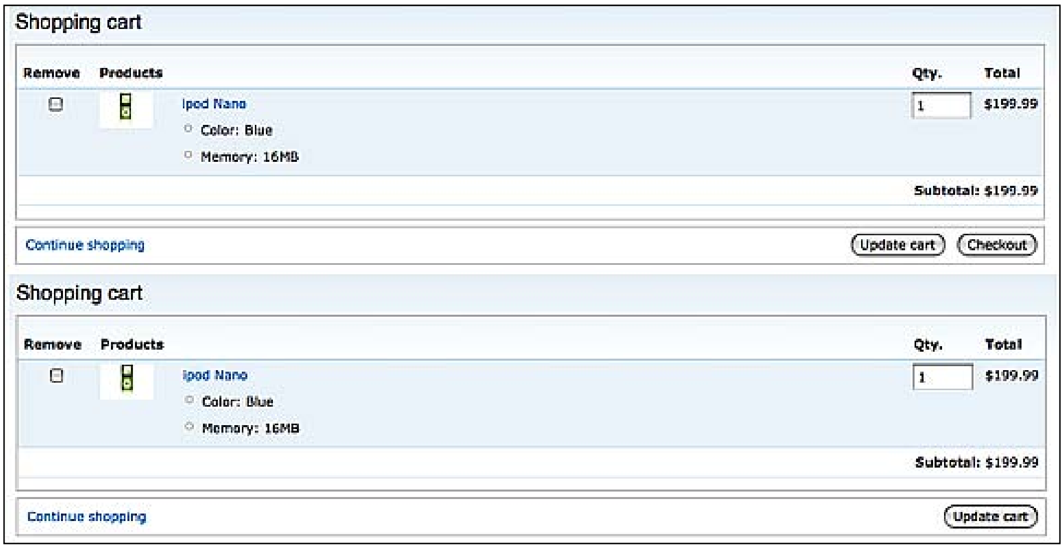
Understanding payment workflow

Before we continue with the configuration, we need to have a high-level understanding of how Ubercart handles payments and order stats. After your client finds the product he or she wishes to buy, either by searching your online catalog as we discussed in *Chapter 4, Managing Categories, Products, and Attributes*, or through a Google search (SEO optimization will be covered in *Chapter 10, Optimizing and Promoting Your Store*), the customer cart is updated with each selection. From this point, you have three alternatives as depicted in the following image. It depends on your merchant plan whether you will accept electronic payments on your site, take the payment process offline via bank transfer or a check, or redirect it to a safe provider location.



Let's discuss these options in more detail:

- **Accepting payments outside the online shop**: You can choose to complete the payment process offline either by accepting money transfers or postal checks. This way you will not be dealing with credit cards and online payments but here you add the drawback of breaking the purchase process by asking the user to perform offline actions; this will interrupt the payment workflow and potentially put off the semi-converted customer. This option of accepting payments outside the online shop is ideal for small stores that do not perform many sales. Below you can see two cart review screens, one having a payment option and one not.



- **Accepting payments by redirection**: This method completes the selling transaction; however, it does not take the payment on your store; instead it redirects the user to the subscribed payment gateway interface. This method is usually asynchronous, but it provides significant advantages, the most important one being that you are not responsible for securing your site for payment acceptance. This alternative significantly reduces the administration costs as well as server hardening and certification purchase costs. It is ideal for medium and large stores.
- **Accepting payment on site**: This is the most non-invasive option and is usually synchronous, when the payment gateway confirms that the payment money has already been captured and the transaction is successful. To implement on-site transactions, one should be PCI compliant, a process that needs real-time monitoring and additional security costs.

Payment Card Industry Security Standards Council (**PCI SSC**) defines the common acceptable industry standards for accepting payments on site. The most important prerequisites are an SSL certification ensuring that the store corresponds to a trusted domain, specific standards for encrypting and persisting data (CCV storage is prohibited), hardening actions, and 24/7 store administration. You can find more information at `https://www.pcisecuritystandards.org/`.

Accepting credit cards on site—pros and cons

Having understood the overall picture of payment configuration, now comes the time to decide which policy you should adopt for accepting payments for your online shop. Following the discussion about accepting credit cards on site, we should provide the facts on both the cases. To help you decide, we will go through the advantages and disadvantages of accepting payments on your site, which requires handling credit card encryption and securing your online shop. The objective is to provide enough information to validate your needs and find pros and cons associated with each choice, questions to ask before signing in, and potential pitfalls. Here are the facts that will help you decide:

Pros

- **Increased sales by accepting credit cards**: Accepting credit cards on site or via redirection increases the probability, speed, and size of customer purchases. Many people refrain from carrying cash nowadays. Others prefer to pay with credit cards because of the ease of making returns or exchanging merchandise. For the business owner, accepting credit cards could increase sales as it allow the customers to make impulse purchases especially when they land on a product page from a Google search or teaser YouTube video. In addition, customers can buy a product even when they don't have cash in hand or they don't have sufficient funds in their current accounts.
- **Improves business cash flow**: Experts advise that accepting credit cards can also improve a business's cash flow, allowing the business to receive the money within a few days or even hours rather than waiting for a personal check to clear or an invoice to come due. Finally, credit cards provide a guarantee that you will be paid, and remove the risks involved in accepting personal checks.

- **Advanced demographics and statistics**: You have the customer's contact information to follow up for sales and marketing purposes. (This is a pro for the merchant but a con from the point of view of many customers, who prefer anonymity.)
- **Expand your client base**: By accepting payments from some of the major payment gateway systems over the world, it follows that your clients are more familiar with the payment process and feel more secure by engaging in financial transactions similar to those they have carried out successfully before. Moreover, modern fraud detection and refunding systems provide the secure and trustful environment your customer wants to be convinced about in order to complete the transaction.
- **Attract more leads**: Raise your Convert Ratio by providing an automated mechanism and selling to the end user at the time he or she is convinced and wants to buy the most.
- **Protect yourself from fraud**: Current payment gateways spend most of their administrative efforts and resources in fraud prevention and detection. By adopting a credit card acceptance system you make sure that you will leave the refund headache behind as most of the services guarantee each transaction that goes through.
- **Lower processing recourses**: This point is pretty much self-evident, as by automating the process a small fraction of your client service department will be needed, along with a huge reduction of your business operational costs.

> Whether you accept credit cards online using an Internet merchant account, through an online payment processor, or both, read all agreements carefully before committing. Do not hesitate to ask a payment processor/provider questions if you find that information is unclear or incomplete. Doing so can help you avoid costly or time-consuming mistakes.

Next in our analysis is some information about drawbacks to accepting credit cards payments on your site. As you will notice the advantages of adopting on-site credit card payment outweigh by far any disadvantages.

Cons

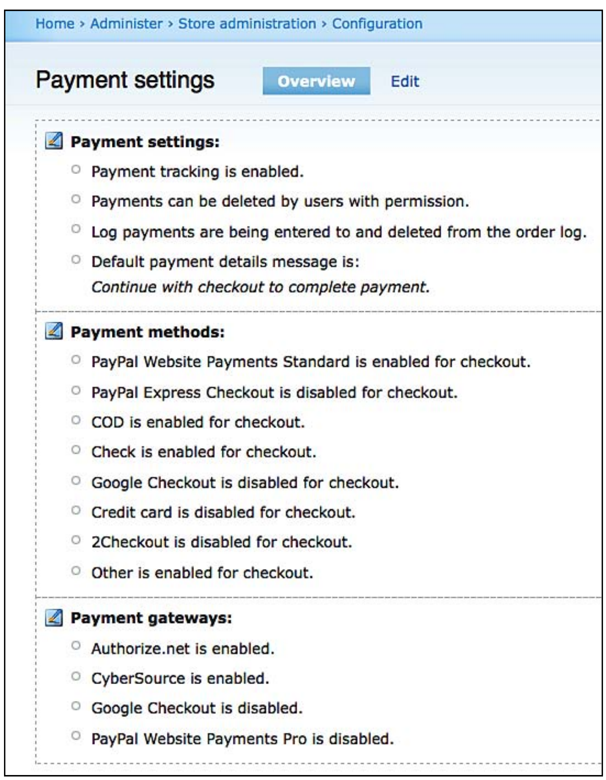
- Clients still have issues in providing financial information over the Internet.
- Not everyone has a credit card but many people recently use prepaid cards only for online purchases.
- This method does not lend itself well to the purchase of downloadable soft goods such as software, art, graphics, and so on. Vendors wanting to sell downloadable soft goods will need to find a way to ensure the product is paid for, once downloaded.

- You will have to deal with commission policies and flat rates. This can be a significant problem if you are selling low cost merchandise.

> PayPal, Google Checkout, and Amazon Payments have a policy of a percentage and a flat rate fee on each transaction. These values are added to the total amount of the transaction. For instance, for a payment of $100 PayPal charges an additional 3 per cent and a flat rate of $0.34, resulting to a total payment of $100+$0.34+100*0.03=$103.34. So, you should take into account this fact during setting up your pricing policy. In special cases and when your monthly, semester, or annual revenue exceeds some quota, most of these online payment gateways can provide negotiable commission fees. If you refund a transaction then commission fees captured are returned to your account.

Analyzing payment settings

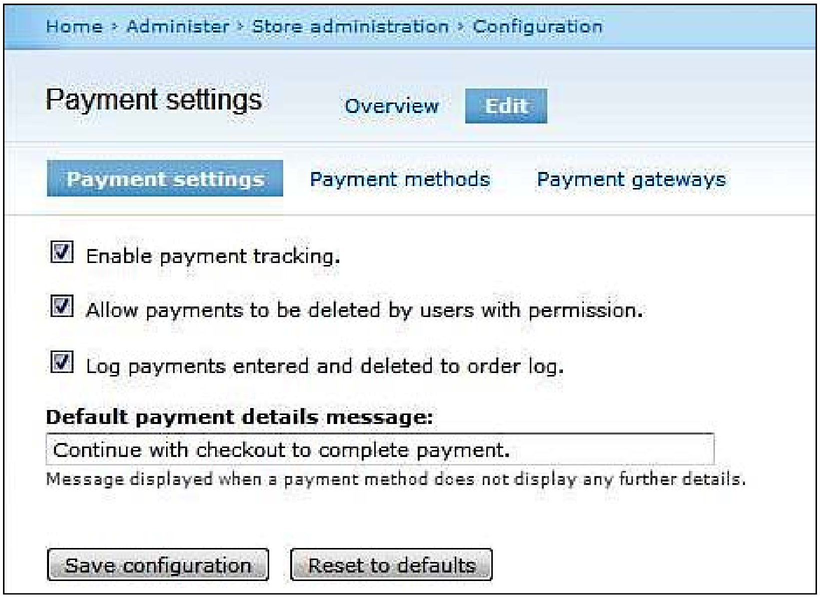
The Payments settings page contains three sections: **Payment settings**, **Payment methods**, and **Payment gateways**, as shown in the following screenshot:



In the **Payment settings** section, you can configure some general settings for the payment systems in your store. In the **Payment methods** section, you can configure some basic payments systems such as PayPal Website Payments Standard or Cash On Delivery. In the last section, **Payment gateways**, you can set up more complex payment systems such as PayPal Website Payments Pro or Authorize.net.

Payment settings

These are the basic settings for your payment system as shown in the following screenshot. The first three checkboxes refer to two systems in place to track payment information. If the first box is checked, a ledger is kept for each order showing payments received/refunded along with the current balance for each order. If the second box is checked, you may grant permission for people to delete payment lines from these ledgers in the event of mistakes. The third checkbox makes sure that payment information is copied to the order log, which cannot be deleted:



Finally, at checkout, when a customer is choosing their payment method, Ubercart automatically loads up extra details that must be filled out by or displayed to the customer. In the event that the payment method requires no further information, this text field lets you specify a default message that gets displayed to the customer so they know it is safe to proceed with checkout.

Payment methods

The table at the top lists the different payment methods available on your site and lets you choose payment gateways to process these types of payment. For the default methods such as **Check** and **Other**, no gateway is required. If you've installed modules that define extra payment methods such as the Credit module, you may also want to install a payment gateway module that will let you process credit cards through your site. The select boxes here allow you to choose the default processor per payment method.

Payment methods require their own settings, some more than others. To properly configure the payment methods in use on your site, click on the title below the main table to expand a group of settings for that method. Be sure to save your changes when you've finished editing them, and be careful that you don't click the reset button accidentally!

Payment gateways

In this tab you can edit payment gateways, and configure API keys and credentials. Each gateway module you install will generate its own configuration screen. Mainly you can configure gateways and choose whether or not to go live or test in the production stage. We will see detailed configuration for particular gateways in the following sections.

Selecting payment methods

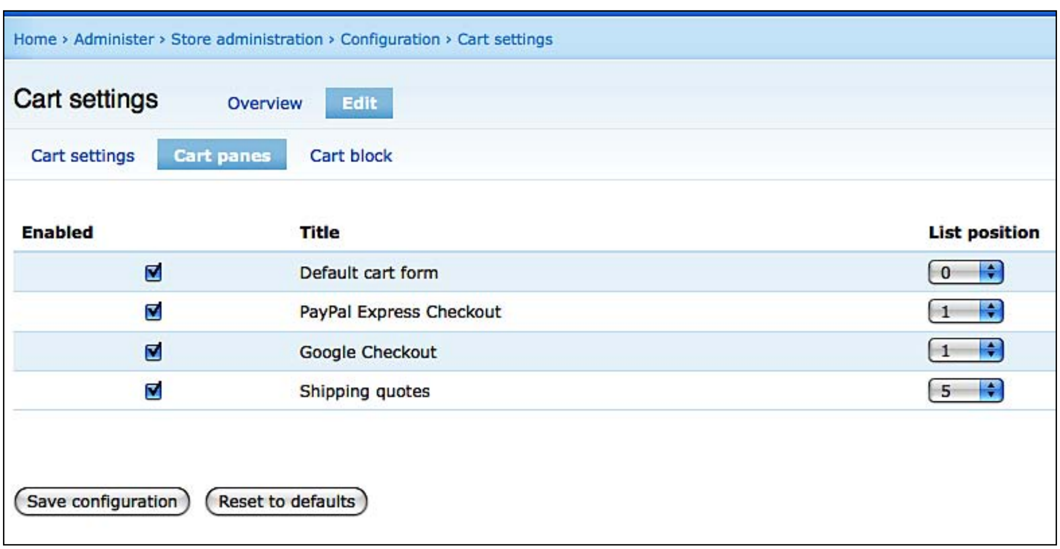
Choosing the appropriate payment method for your business is not a trivial task. The golden rule *the more the better* in most cases can confuse the customer and redirect load to your customer support center. In general, you should use one credit card processing method along with redirect payment methods like PayPal Checkout and/or Google Checkout. For instance, if you are accepting PayPal payments from your PayPal account holders, credit card charging can be handled by your configured account in PayPal Pro.

If you choose to do it on site, which adds value to your online shop, minimizing administration costs for order fulfillment, payments, and dispatching, you should activate this option by navigating to **Home | Administer | Store administration | Configuration | Checkout settings | Checkout settings**. The first checkbox in this screen, marked as **Enable checkout**, is the one you want to activate. This way you will be accepting credit cards on site, approving them, and capturing money via web services gateways. Client experience will be uniform with subsequent steps and payment will be less invasive to the purchase workflow.

The following is the payment gateway selection screen (**Home | Administer | Store administration | Configuration | Payment settings | Payment methods**):

Now, we will discuss in detail the various choices a business owner can make:

- **PayPal Website Payments Standard**: Default PayPal redirection functionality is added to your site.
- **PayPal Express Checkout**: Express redirection to PayPal from the cart page. No user data will be filled out since the customer is being redirected to PayPal for amount capture. You can activate this functionality by clicking the appropriate checkbox on the page **Home | Administer | Store administration | Configuration | Cart settings | Cart panes**.



- **Check**: This is the default Ubercart functionality for check payment. No checkout functionality should be enabled.
- **Credit card**: By enabling credit card payment support you will be accepting card numbers and customer data on your site, and will be storing them in your database. In this case you should also enable encryption, as we will see in detail in *Chapter 10*. Under this option a great number of gateways that provide web service functionality for credit card charging are available in Drupal repositories. For this particular reason a drop-down will be provided in order to choose from all available and configured gateways. Special keys, validation, or configuration details for each gateway will be administrated through screens on the payment gateways page (**Home | Administer | Store administration | Configuration | Payment settings | Payment gateways**).

Ubercart third-party payment modules

One of the most demanding processes during development of a payment-enabled online shop is the integration with a payment gateway, raising issues such as security, development, and testing. Luckily, Ubercart is an open source project with a highly motivated community, a model that minimizes the danger of security holes, and that maximizes feature implementations, since new functionality is returned back to the community. In this context of collaborative development, a significant number of gateway-specific modules exist in Drupal repositories and the Ubercart contributed modules site.

> You may find more payment gateway implementation for Ubercart at `http://twurl.nl/nxdqwk` along with data about the current status of most of the projects that you may find at `http://www.ubercart.org/contrib`.

Especially outside the USA, where there are not so many options and established services, integration with local bank payment systems is vital for local business owners. Ubercart provides many out-of-the-box solutions, but do not forget that if you cannot find a suitable payment module for your payment provider service or your local bank, someone is probably facing the same problem and is willing to help you out.

Some interesting payment gateways besides the one we have already mentioned are:

- **Adyen | Ubercart Payment** (`http://drupal.org/project/uc_adyen`)
- **SPplus payment for Ubercart** (`http://drupal.org/project/uc_spplus`)
- **Ogone | Ubercart payment** (`http://drupal.org/project/uc_ogone`)

- **IDEAL Pro | Ubercart Payment** (`http://drupal.org/project/uc_ideal_pro`)
- **Ubercart eWay Payment Gateway** (`http://drupal.org/project/uc_eway`)
- **EBS Payment Gateway for Ubercart** (`http://drupal.org/project/uc_ebs`)

Integrating with payment gateways

Accepting payments using a preconfigured payment gateway will complete your online shop functionality, and provide a more integrated experience to the end user, as we have discussed in the previous section. This way the purchase workflow will be easily followed by the user and this will avoid any questions or misunderstandings. Moreover, you can provide your customer with an automated purchase process, reducing client support, and communication costs.

When it comes to selecting the payment gateway you will go ahead with, you should keep in mind the following concerns. You are establishing a business partnership and you need the maximum amount of support, service availability, and credibility from your counterpart. Also keep in mind that this gateway is the front line of your store when it comes to credibility and trust issues. You are dealing with your customer's money and this is something vital to the success of your online business. You should not take any chances on that at all.

The following are some points you should consider when you choose your payment gateway:

- Ask for the uptime statistics and availability data of the service.
- Check the status with lost password policies and account administration retrieval. For instance PayPal, due to strong fraud detection policies, has rather strict procedures that can even lead to account suspension.
- If you are not situated in the USA or UK, consider obtaining a USA or UK VAT number. This way, your final gateway candidate pool will grow exponentially.
- Discuss commission policies; some gateways besides the standard commission fees can provide negotiable commission policies, especially if your yearly or monthly revenue exceeds some thresholds.
- Check for the user base of established services. For instance, PayPal has a large number of users that have a habit of performing transactions through the specific payment system. They have income from AdWords or relevant promotion actions and they tend to think of the service as their personal bank. So, in that case your client base will have an extra reason to fulfill the purchase with payment if they feel familiar with the service itself.

Paypal: Create account, customize, connect with Ubercart, manage

PayPal is the most famous payment gateway and has obtained its reputation from the high fraud detection and security level it provides. PayPal is ideal for medium and large online shops, and can also accept PayPal users that handle credit card charges internally. PayPal supports redirection with both PayPal Checkout and web services, that is on site credit cards with PayPal Pro. The only constraint is that PayPal Pro functionality needs a United States address and VAT number, so all of us outside the USA cannot use it.

Furthermore, we will describe the process of setting up an account in PayPal and configuring our online store to accept PayPal transactions step by step. If you already have a PayPal account then go to *PayPal account upgrade instructions* directly and skip the account set up instructions.

PayPal account setup instructions

Here we will set up a verified PayPal Business Account. Store owners who don't have a PayPal account should follow these steps:

1. Go to PayPal at `http://www.paypal.com`.
2. Click on **Sign Up**.
3. Set up an account for business owners. Input your regional data and business information. Do not forget that you should have a US-based address associated with your credit card to go for the PayPal Pro services.
4. Follow the instructions on the PayPal site. At the end, choose a password and two security questions for the account ownership claim. You should be rather careful as PayPal has some strict fraud detection policies. Note down the questions and answers you provide, to avoid headaches if one day you lose your password and are forced to retrieve it. For those who do not have e-mail access and security question answers, PayPal just closes the account and sends only a balanced check.

PayPal account upgrade instructions

To upgrade your PayPal account, follow these steps:

1. Go to PayPal at `http://www.paypal.com`.
2. Click on **Upgrade your Account** link.
3. Click on the **Upgrade now** button.

4. Choose to upgrade to a Business Account and follow the instructions to complete the upgrade. If you haven't already, add a bank account to become a Verified member.
5. Follow the instructions on the PayPal site. This process is simple and PayPal charges you with $1.5 that you will be refunded in your first transaction, in order to provide you with a verification code. The total time to get the transaction in your balance sheet may take two or three business days. Therefore, it would be ideal if you have an e-banking account to monitor your card transactions for the PayPal verification code. In the meantime you can accept money with only the constraint that you cannot put it in a bank account via a credit card. If your transactions exceed $750 per month then PayPal will provide this money via check.

We have gone through an extend discussion about whether or not to accept credit cards in our site. We will now continue and configure our online shop to accept credit card numbers and perform the payment via PayPal's Pro web service.

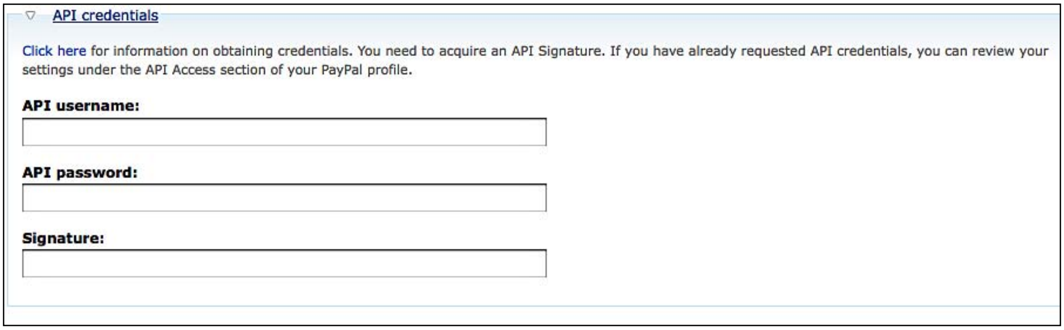
Apply for Website Payments Pro

To apply for Website Payments Pro follow these steps:

1. Go to PayPal at `http://www.paypal.com`.
2. Click on the **Merchant services** tab.
3. Click on **Website Payments Pro (U.S. Only)**.
4. Click on **Sign Up now**.
5. Fill in your information, and submit your application. Approval takes between 24 and 48 hours.
6. Once approved, accept the Pro billing agreement. Check the **Getting started** section on the upper left of your account overview page.

First-party authentication: Register API signature:

1. Log in to your PayPal account and click on the **Profile** sub-tab.
2. Click on the **API Access** link under the **Account Information** column.
3. Click on the **Request API credentials** link.
4. Select the API signature credential.
5. Browse to **Home | Administer | Store administration | Configuration | Payment settings | Payment gateways** and fill in the API credentials information in the PayPal Website Payments Pro settings section of the page.



Your account is now set up. You should activate the security in order to be **Payment Card Industry Data Security Standard** (**PCI DSS**) compliant. We will return to this subject in *Chapter 10*, where we will discuss security issues.

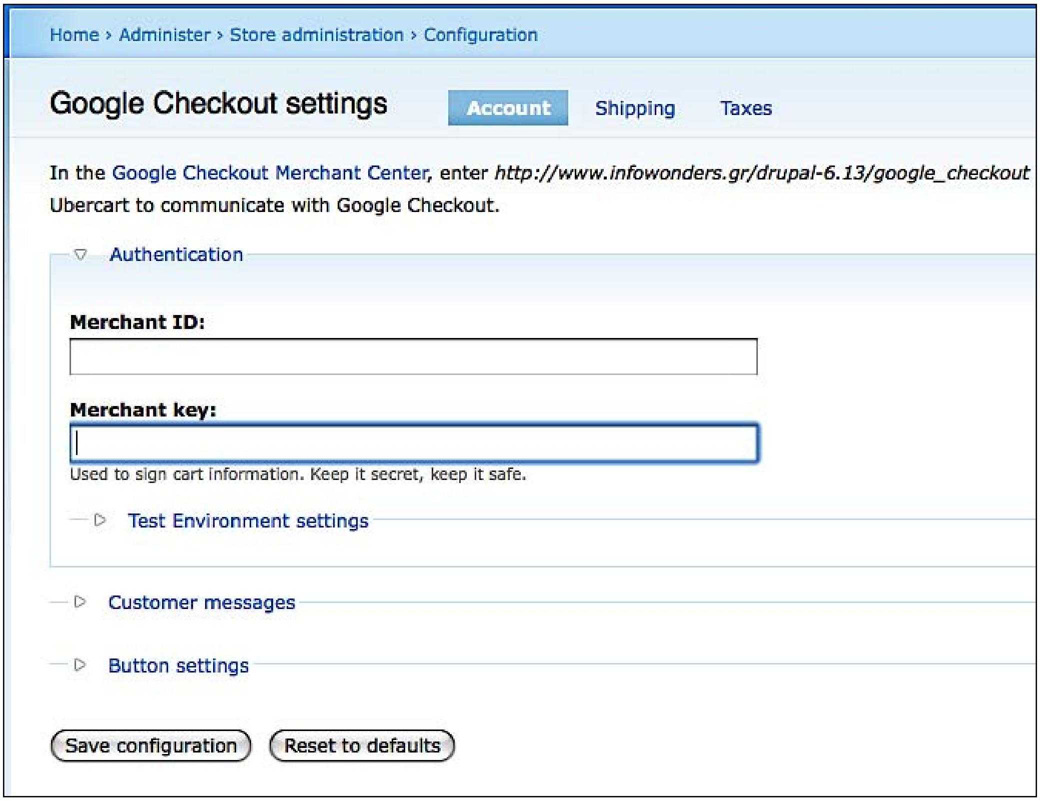
Google Checkout: Create account, customize, connect with Ubercart, manage

Google Checkout is the new kid on the block when it comes to payment gateways. It is being marketed with competitive commission policies in order to position the service highly in the market and the brand name of Google as the supportive and reliable partner that everyone would want to have. To configure Google Checkout you will need to install the appropriate payment gateway implementation module `uc_google_checkout`. You can find it at `http://drupal.org/project/uc_google_checkout`.

Configuring Google Checkout

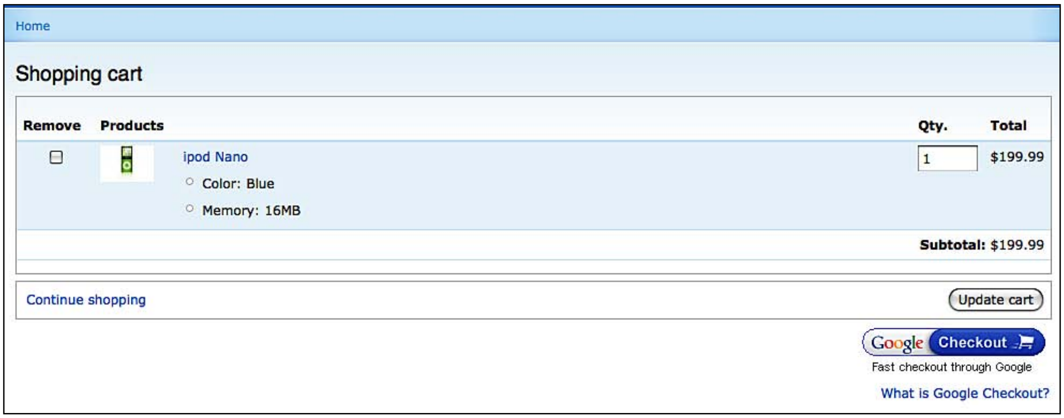
1. Go to Google Checkout (`http://checkout.google.com/`).
2. Click on the **Sign up now** button.
3. Read the instructions and provide your information. You need to be a UK or US citizen and must have a credit card with a valid address assigned to it.
4. Provide your credit card details.
5. Go to the Google Checkout settings page at **Home | Administer | Store administration | Configuration | Google Checkout settings**. A **Merchant ID** and **Merchant Key** are needed to send orders to Google Checkout, and they can be found through the link to the Merchant Center at the top of the page or by just browsing to `http://checkout.google.com/sell`. This link sends you directly to your account administration page in Google Checkout.

6. At the Merchant Center, go to the **Settings** tab and click on **Integration**. The Merchant ID and Key (which should be mailed to you during registration) appear on the right-hand side.
7. Insert Merchant ID and Key into the administration screen in **Home | Administer | Store administration | Configuration | Google Checkout settings**.



8. Click on **Save configuration**.

9. Go to **Home | Administer | Store administration | Configuration | Cart settings | Cart panes** and check the Google Checkout pane.
10. Done! Test your result by reviewing your cart page at `http://www.infowonders.gr/drupal-6.13/cart`:



Create your own payment gateway from scratch

We have gone through some of the most widely used payment gateways available out there. However, the truth is that you are most probably not living in the USA so inevitably you do not have a US VAT number. This fact minimizes the available selections to a very narrow range. In other words you cannot escape from PayPal redirect service. (At this moment, as we are writing this book, Paypal Pro is not available outside the US.) In this case and even more if you already have your business attached with your region's local bank, you may come to the awkward position of developing your own payment gateway module. We are referring to this situation as awkward because you will be facing a dilemma, similar to or greater than all the other development decisions you have made during the process of your online store design.

The question is whether to try to build it by yourself or seek professional help. We will further explain the thought process you should follow to come to an answer for both possible answers.

1. If you have no or only moderate experience with PHP and MySQL development, the answer to this question is obvious. You do not want, by any means, to mess with your online business transaction credibility and integrity, so the best action is to refer to Drupal module development experts. Furthermore, as Ubercart has a pretty specific hook mechanism and some concepts are module-specific, you should start your search with Ubercart module developers and then move on to Drupal module developers. A good place to search for developers near your area is the Drupal.org member section. There you can perform location- and country-based searches for Drupal-registered users and by inspecting their profile you can also check out their contributions. Drupal and Ubercart provide an open source platform and come with all the benefits of products under GPL license policy. In other words, you can check out other people's contributed code, inspect their quality, and if you have a little experience, get to know the developer without even meeting that person. If you decide to ask for professional expert services, you do not need to read this chapter further. Some places to start your search in order to contact experienced and competent Drupal and Ubercart developers are:
    - `http://association.drupal.org/`
    - `http://drupal.org/forum`
    - `http://www.ubercart.org/forum`
2. If you believe in your skills and have developed a Drupal module successfully in the past, you can try to build your payment gateway on your own. You understand that you are dealing with your bank transactions and should be extremely careful with good design practices and a solid testing plan. Before you start building your module, you need to perform the following actions:

i. Contact your local bank or the bank institution that you would like to perform business with. Ask them about their automated payment gateway interface and ask to get all the documentation material that accompanies the system. Most of the modern banking systems provide information for your development and testing phases that includes types of gateways in the form of documentation, code samples, error codes and their definitions, test credit cards, and CCV codes. If you get lucky, you may get PHP sample code. Request types generally vary between web service calls (SOAP envelopes), XML/RPC calls, and RESTFULL services. Each of these interfaces especially the POST methods can be either URL or request body posting procedures. We will not go through the entire TCP/IP communication protocols, as every bank organization implements its own and no interoperability standards apply to the field. However, we will provide you with code samples for two of the most common request methods XML/RPC and Web Service calls.

ii. Come up with a solid and "bullet proof" testing plan, either by consulting your gateway documentation material or improvising and implementing your own failure situations. Do not forget to simulate a connection loss, a socket timeout error, and a response that exceeds 20 seconds, which is the default HTTP response timeout.

iii. Download existing and thoroughly tested gateways from Drupal.org and check out what best practices and problem solving techniques other developers have adopted. Do not forget that you are coding on an open source platform so turn this fact on your benefit.

iv. Decide the data that you want to store per transaction. We will see how Ubercart provides this option or how you can expand it if you need more specific information persisted. Keep in mind though that you need extra hardening and standard matching in order to store sensitive financial information such as credit card numbers. Furthermore, to be PCI-complainant you cannot store CCV numbers whatsoever.

If you have completed the checklist above, then you are ready to start developing your payment gateway module. As final remark, we should point out that as we have been discussing in the payment section of this chapter, there are two kinds of payment gateways, the redirects and the APIs. Recall that redirects do not need extra security as the credit card information is handled in third-party websites, as opposed to gateway APIs where you manage the credit card submission and secure data transferred to your server by yourself. The later approach demands an SSL certificate to mark your web domain as trusted.

Summary

In this chapter, we have thoroughly discussed shipping taxes and online payments. These two issues are some of the most important aspects of your business and should be tackled before you start building your online shop. We have approached them with clarity, providing a superset of the available information out there. When it comes to payments and tax handling, it is better to be safe than sorry. In addition, it is our common believe that issues like online payments and taxes, which are world-wide concerns, should be approached in a case-study manner, as tax systems and payment gateways vary not only from country to country, but from state to state in most cases. Summarizing in this chapter we have gone through:

- Tax rate configuration
- Payment gateway integration

Finally keep in mind that good research pays off and always search beforehand for the payment gateway and the shipping policy you will adopt and bind your business to it. This will save you from a lot of trouble in the future.

In the next chapter, we are going to explore the best practices for order processing; trying to make your clients happier and your life easier. We will guide you on how to provide better customer service using Customer Relationship Management and Community building tools.

7
Managing Customers and Orders

This chapter talks about CRM configuration, reporting services, and after-sales support for your online shop. All these aforementioned areas form crucial aspects of your day-to-day business administration and should be designed keeping in mind that you probably are going to spend most of your time working in front of these screens. We will provide the initial steps and guide you through the overall process of designing a proper customer support plan and keeping your customers happy and satisfied during the whole purchase process. Good customer relation management is the essence of successful online trading, where the average customer is suspicious and provides fertile ground for your business reputation to grow.

The topics that we will discuss in this chapter are:

- Managing customers
- Managing orders
- Ubercart reports
- Customer Relationship Management and royalty programs
- Community Building Actions

Managing Customers

We are now stepping into the field of your core business and day-to-day administration tasks of your online e-shop. Therefore, remember that the configuration tips and methodologies we will go through in this chapter are actually going to make your life easier and help you to get the most out of the automation an online shop brings.

Viewing customers

A significant aspect of successful CMSs is the inspection and administration of potential new customers as well as tracking down their activities and purchases. To view all the customers of your online shop, go to **Home | Administer | Store administration | Customers**. On this screen, you can see all newly-added users as well as past users that have gone through a successful purchase on your online store. From this screen, you can also navigate to customer orders.

Viewing orders

Furthermore in our discussion about administration features that Ubercart provides the end user with, we will review another important part of your everyday business, namely checking your store's customer orders. Ubercart uses Drupal's advanced reporting services and can provide vast amount of data for viewing statistical information and orders per se. To do so, navigate to **Home | Administer | Store administration | Orders**, and take a quick view of entire order list. As you can clearly see, you can sort the order list by clicking on the column names and then choosing between ascending and descending order. In addition, you can have extra search criteria, such as order statuses and order number. This is your main entry point, and from here you can also edit, fulfill, and administer your orders. You can easily access order edit screens using the buttons next to each order, shown in the following screenshot:

The action buttons will appear according to your status and order fulfillment workflow. Therefore, if you have not configured integration of order shipping with external providers, no shipping options will appear, and the equivalent will happen for further processes like packaging and payment. This can be handy when you want to respond to customer tickets about their order status. Keep this in mind because we provide more details in the last section of this chapter. Also take into account that if you want more sophisticated screen and search options, you can always use the additional Ubercart Views module or use the core Drupal Views module to create reports that match your everyday administration needs.

> Ubercart Views module provides useful views for order listing and bulk order management. We'll also examine it in *Chapter 9, Using Interface Enhancement Techniques*, but you can download it from `http://drupal.org/project/uc_views`.

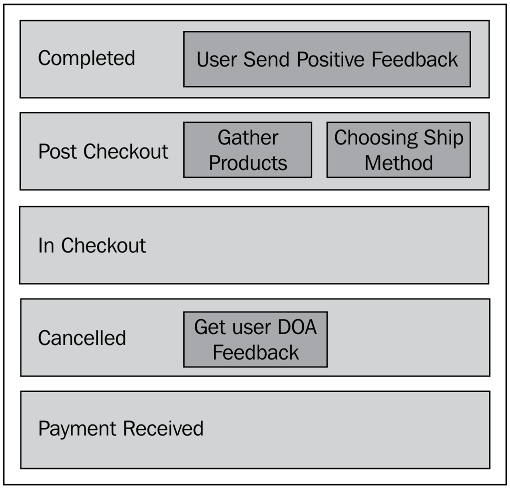
Managing and fulfilling orders

After going through the details of the administration screens for customer and order review, you may be asking yourself where the actual working screens that your personnel and you will be using are. So far, experience dictates that most of your daily workload related to store management will be laid out in the following order completion and fulfillment screens that Ubercart provides out of the box. No need to waste time then; we will go straight ahead describing their structure and functionality.

Order workflow (payment, e-mails, packaging, and delivery)

We are far from covering the area of order workflows, but we provided a high-level understanding of it in *Chapter 6, Managing Taxes and Payments*, where we discussed payment workflows. Order workflow is rather a super set of the described ordering process, as it involves many more considerations like product availability or payment problems. Before we continue our discussion about our online shop administration, we should take a moment to understand the basic concepts of Ubercart order completion workflow.

In every well-defined workflow engine, there are two basic concepts: the **workflow states** and the **workflow statuses**. Consider that states are abstract phases of the business, and statuses are these phases as you apply them in your everyday work according to your business model. For instance the Ubercart state "Completed" refers to a general order status when no further actions should be taken. This is a built-in feature of Ubercart and cannot be changed. Now, if your *loyalty and customer care* policy is to get user feedback after a successful purchase and shipment, you should configure two order statuses in the "Completed" state. Use self-explanatory names like "User feedback collected" and "User order delivered"; this way you can manually change your order status and perform better order monitoring. We show some generalized examples in the following diagram, where the outer boxes provide the structure of the predefined Ubercart states, which are associated with some real-life examples of personalized statuses.

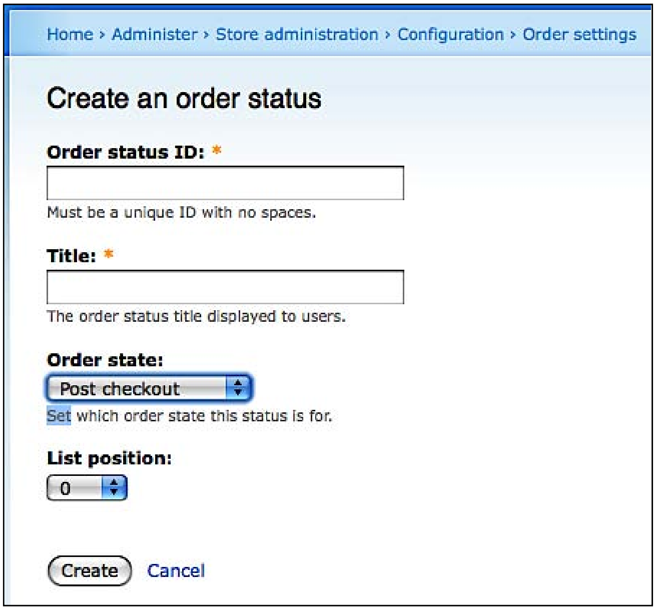


Now that we have grasped the concept of order statuses versus order states, we will proceed with configuring our online store with our custom statuses. Navigate to **Home | Administer | Store administration | Configuration | Order settings | Order Workflow**, to review order statuses created by default from Ubercart. It is clear that each order status is associated with a corresponding order state. Drupal basic states are as follows:

- Canceled
- In checkout
- Post checkout
- Payment received
- Completed

> You can also change the status order, and therefore change your entire business process fulfillment workflow. Unfortunately, currently Ubercart does not support parallel states and the entire order fulfillment execution workflow is serial.

To create a new order, click on the **create new status** button just after the status table. The following screen will appear, providing the ability to create new custom names for your business statuses and associate them with the default states. Finally, click on **Create** to make your new custom order status.



Next, we need to cover **order panes**. Order panes define the exact amount of information you want to show to your customers during the different steps of this completion of their order. You have three options to configure the following view screens of the order workflow:

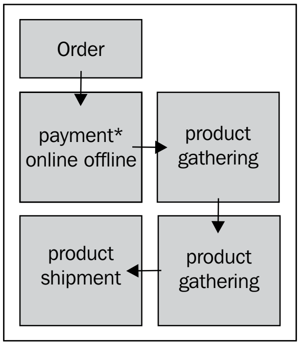
- Order panes on the view screen
- Order panes on the edit screen
- Order panes on the invoice screen

Go to **Home | Administer | Store administration | Configuration | Order settings | Order Panes**

Order panes are actually the information lines you want to add to your invoice pages. You can fully customize the behavior to role level, providing more detailed invoice screens to your end users. You can customize edit, view, and invoice screens. It is strongly advised to provide the maximum amount of information on your customers' invoices and order review pages, because this will result in a huge work load relief to your customer support center.

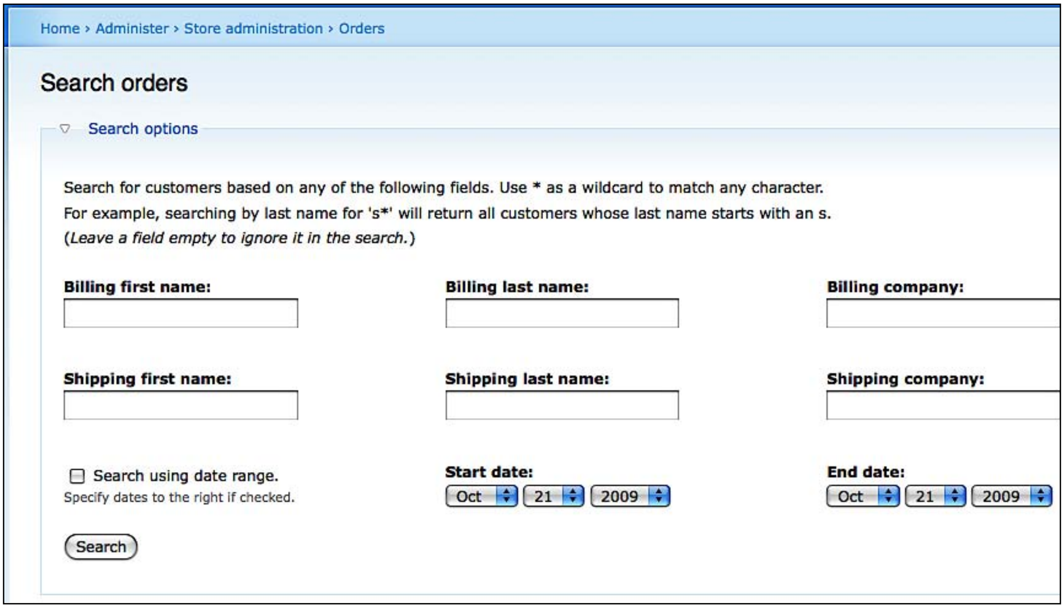
Moreover, to configure customer and order view screens, you need to go to **Home | Administer | Store administration | Configuration | Order settings | Order settings**. Here you can administer the order viewing screens. You can tweak configuration options like maximum number of orders in the overview screen; you can also use paging, so set this to something particularly large, or enable the end user to view invoice pages concerning their orders. This last feature will save your personnel and you from large amounts of client-service work, as often there are users with completed orders who come back and ask you to resend their invoices. This will save you some trouble and keep your customers pleased, as they will always be able to make it clear to their local tax office.

As a general guide and a starting point to create your own customized business workflow in accordance with your special market and sales needs, we provide a simple but quite complete diagram.



Searching orders

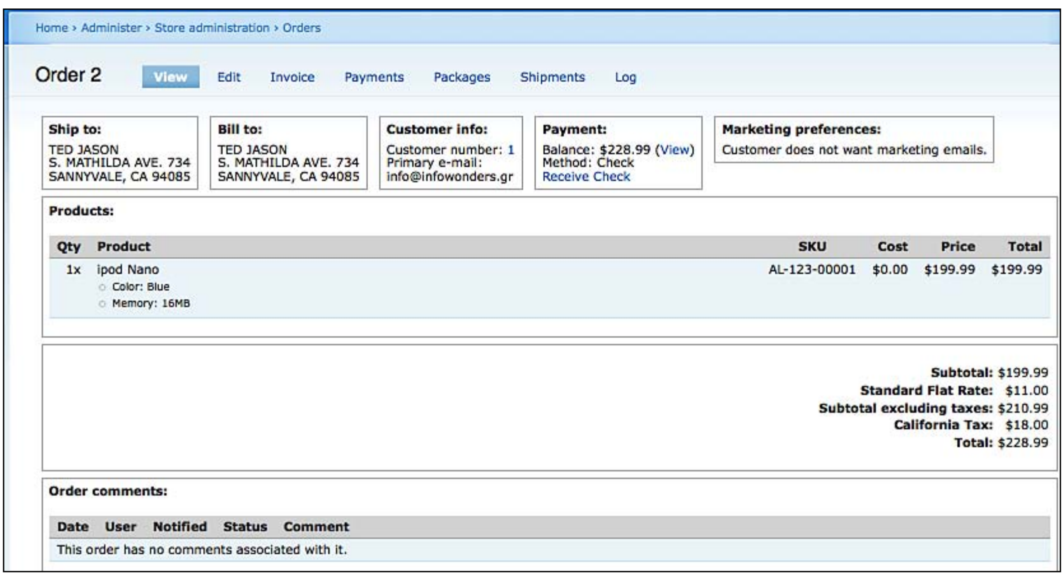
Now, after dealing with **First-In First-Out** (**FIFO**) order screens, we will give some insight into the order search screens and point out some important areas one has to pay attention to in order to make day-to-day administration easier. Ubercart view enables us to perform searches based on a number of different criteria, such as customer details along with shipping and packaging options. Although there are a variety of criteria, most times a simple date and customer name would be enough. Search order results are provided in the form of the customer and order views we have discussed. This is also a starting point to further administer each particular order we got as our search result, just like one can do in the customer and order view screens. A screenshot of all possible order search criteria is provided below:



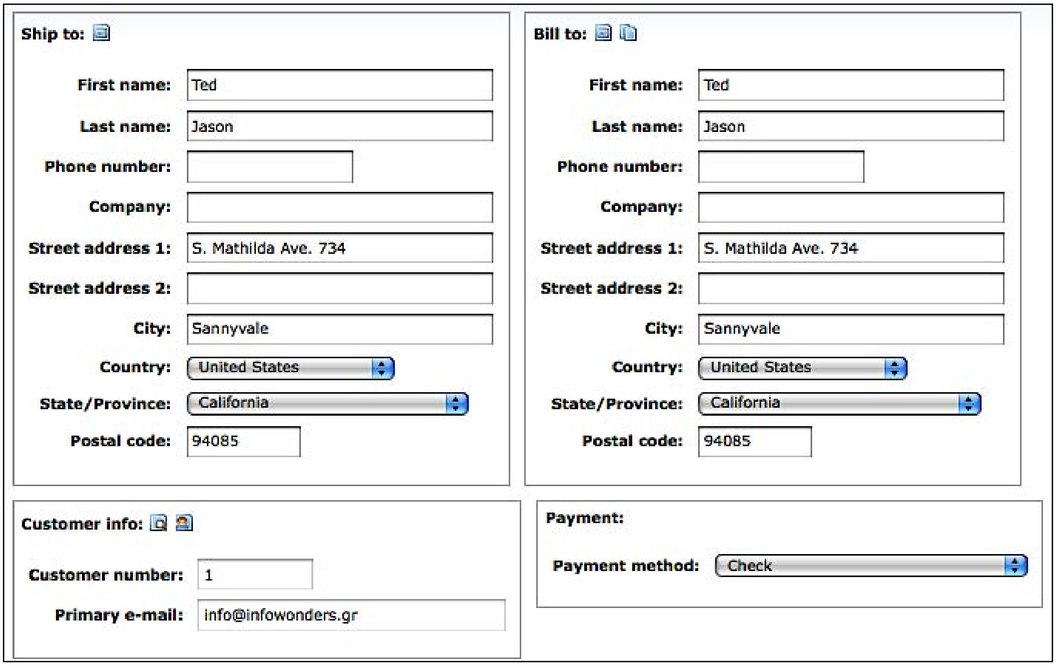
Viewing orders

When it comes to order monitoring and tracking, Ubercart provides the end user with advanced reporting services with a vast amount of statistical information and order data. To access this screen navigate to **Home | Administer | Store administration | Orders**. From this point on, you can perform actions like sorting the order list, by clicking on column names and choosing between ascending and descending ordering. In addition, you have extra search criteria like order statuses and order numbers. This can come in handy when you want to respond to customer tickets about their order status. Keep this in mind as a basic strategy because we will discuss it in detail in the last section of this chapter. Also take into account that if you want more sophisticated screen and search options, you can always use additional Ubercart view modules or use Drupal's Views module to create more detailed reports.

On the orders view screen you can see all the extra details that have been saved by the system during customer checkout and additional details concerning payment, packaging, and shipping corresponding to different steps of the order process workflow. The order review page looks similar to the following:



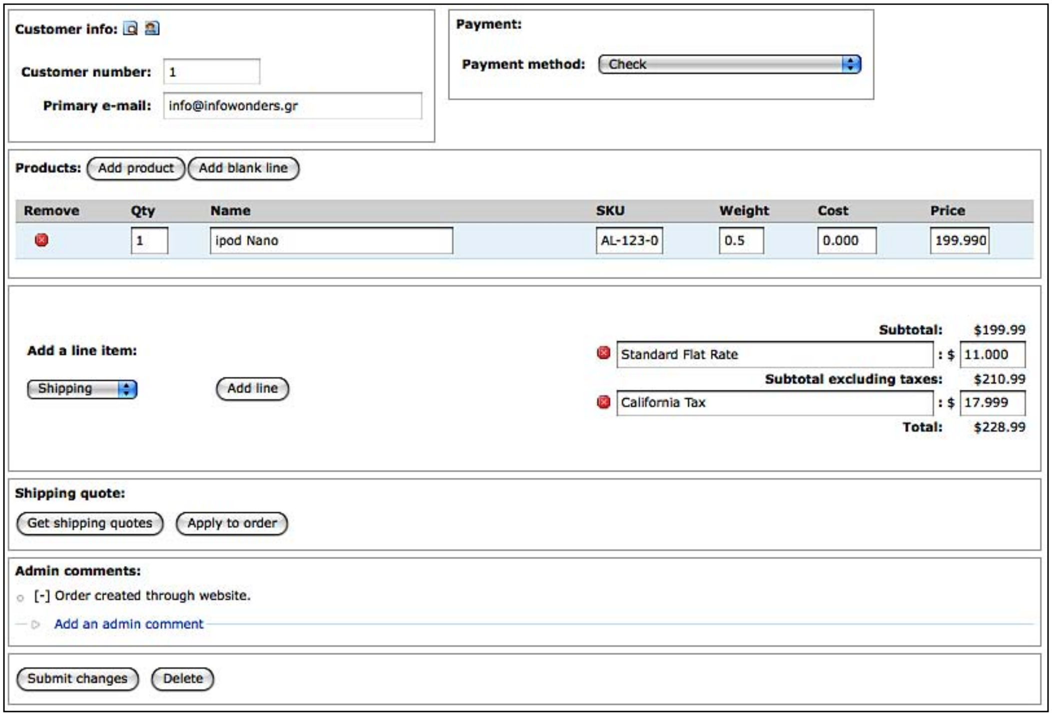
Of course, viewing orders is not the most common thing to do at this screen. You will spend most of your time editing orders, answering to customers' change requests and packaging preferences, making manual payments, and so on. In this context, this detailed edit page comes very handy. You can change personal and contact information, order items, order number, as well as shipping and billing addresses. The configuration screens for these options are as follows:



Another interesting customization feature is the order cart itself. All the items in the order can be deleted and added from the administration screen. More specifically, you can configure the following aspects of your customers order:

- **Number of items**: You can modify the total item number and also add different items from the same or another product category. You may also attach entire product kits to the order.
- **Manually change payment statuses**: This feature lets you configure the way you want the payment made, and in the case of external payment, you can set the order status as "Has been paid".
- **Configure line items**: In the case where you want to apply extra fees to the order or manually calculate service, shipping, and insurance fees, you can add a new line to the item ordered, adding up to the total amount, also providing a short description about the nature of the charge. This is a manual administrative way but offers more flexibility and control over your online store order management. If you plan on fulfilling such a large number of orders as to make manual integration with the order workflow forbidding, do not forget that, as we have discussed in the previous chapter, you can always automate this procedure by integrating with external systems and letting Ubercart do the calculations for you.

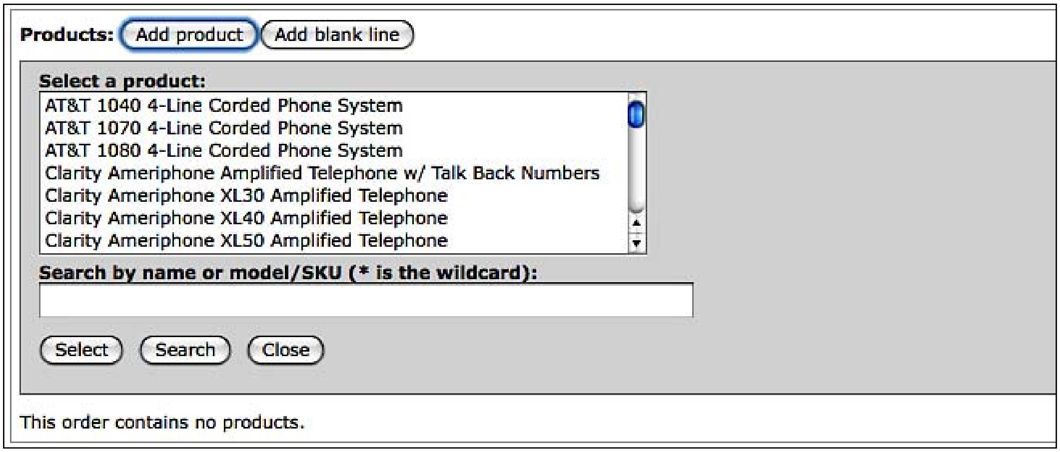
- **Commenting**: You should use this feature in each order you review and modify. It will save you a lot of your future time, when you look back and ask yourself why you gave this discount or why you offered free shipping and so on. Often, in the case of just one or two order administrators, you are strongly advised to use it, but in the case of multiple users claiming and closing orders, you are obliged to use it. Literally is the only way to keep track of what is happening. You may also associate auditing messages with each action that resolves to an order status or data change.



Creating an order from the admin panel

To create a new order in your online shop, go to **Home | Administer | Store administration | Orders | Create Order**. If your business model supports alternative sales channels besides the Internet, you will be visiting this screen often, so we will go though it in detail.

1. After you select the **New order** button, you are provided with two options: either create an order associated with an existing customer, or create a new customer associated with this order. If you choose to use an existing customer, you can perform a search, either using the customer ID or customer e-mail address. This is actually a straightforward procedure that you can follow even from your customer care call center. Your customer tells you his or her e-mail address and you search for it. If an e-mail address search retrieves no results, this means that he or she is not an existing customer, and you should create a new customer account. User creation is a single-step process, as it only involves creating a user and associating that user with the e-mail address you provide.
2. The next page is the order view page as we discussed in the previous section. However, in this case, there is no data filled in it except for the information concerning the user we have just chosen.

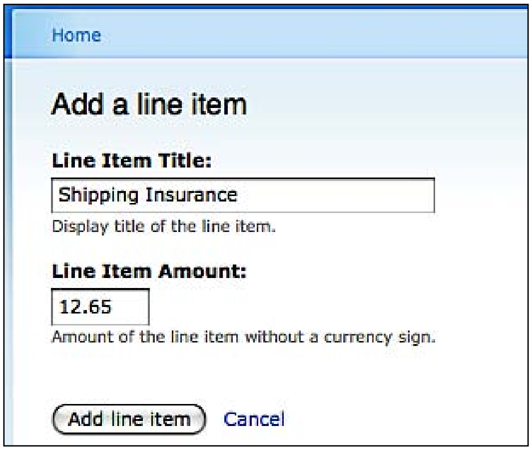


Processing orders

To provide complete coverage on the task of order management, which will take more than 70 per cent of your time running your online shop, we will categorize the most common administration tasks one should carry out for a sustainable and successful business. On top of simple order processing that we have been discussing, are the following more complex tasks.

Adding line item amount

Adding a new line item from a customer request you got through e-mail or a phone call should be your routine from now on. To complete this task, go to the order view screen (**Home | Administer | Store administration | Orders**) and, to view the order you want to change, click on the **Add a line item** button. You will see a screen similar to the one seen in the following screenshot and you'll be prompted to enter an explanatory description for the charge and a valid amount:



Line items represent additional charges and often encapsulate the business logic of extra charges, such as:

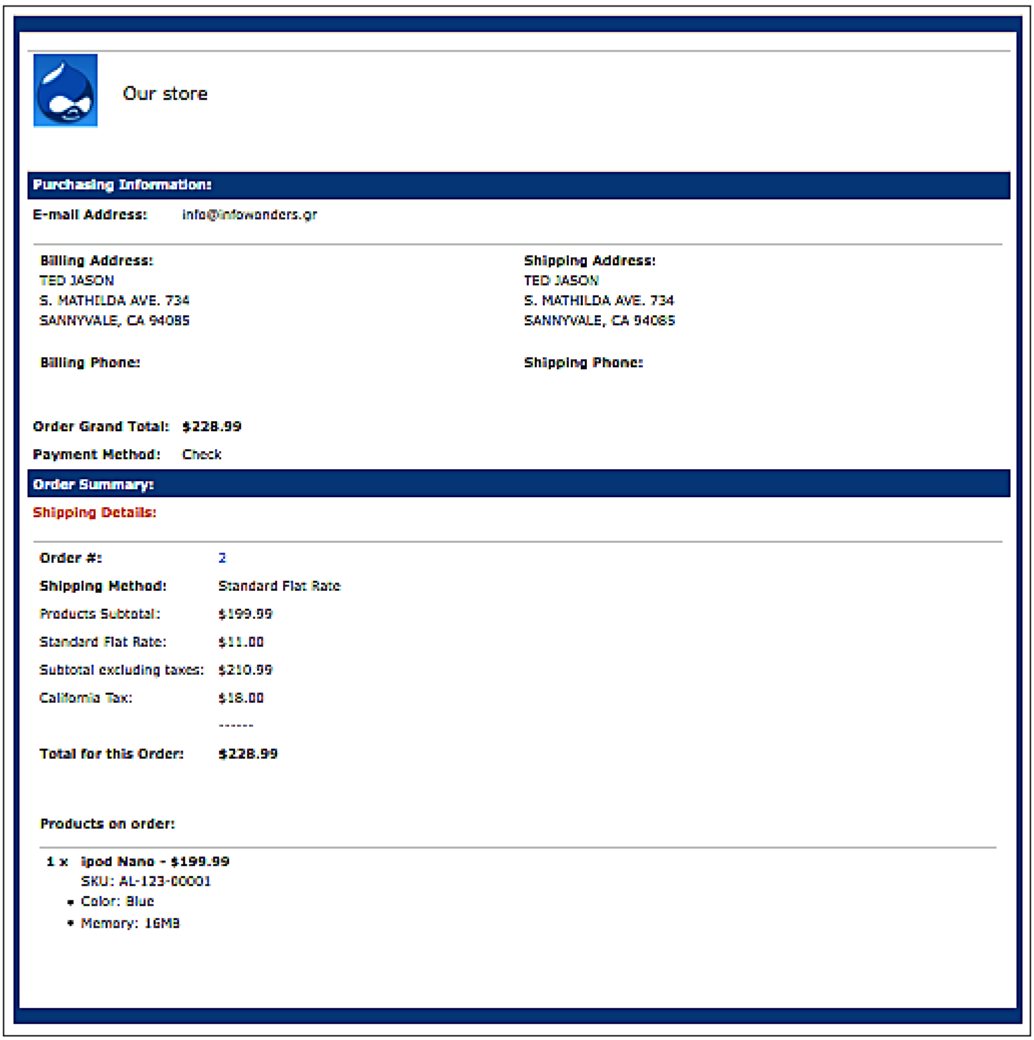
- Shipping insurance
- Shipping fee, if you are managing the shipping manually
- "Hard-to-find" fee

Merging orders

There is no default procedure for merging orders in Ubercart. We may come across such requirements because in the case of online stores, often it so happens that a user places another order after a gap of few days or they may be having second thoughts about a product they are searching for and eventually buy it. In this case, you have to erase one of the two orders and then add the extra products to the other. This is the only way to make a single transaction and perform only one packaging action. This is not true if the customer has already paid for the purchase through an automatic payment gateway. The only thing you can do in that case is to handle the shipping and packaging configuration outside the system.

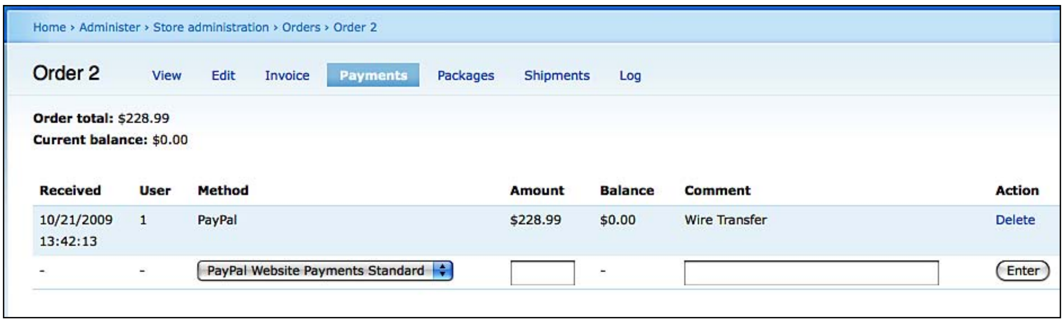
Invoicing

Invoicing is a very common action and also an inevitable part of the order procedure. You cannot sell anything without providing your customer with an invoice, no matter which place on the planet he or she is buying from. Navigate to **Home | Administer | Store administration | Orders | Order # | Invoice**. The Invoice screen is fully configurable and holds all the purchase data. You can either see the invoice document in rich HTML view or print view. In addition, you have the option to mail it directly to your customer.



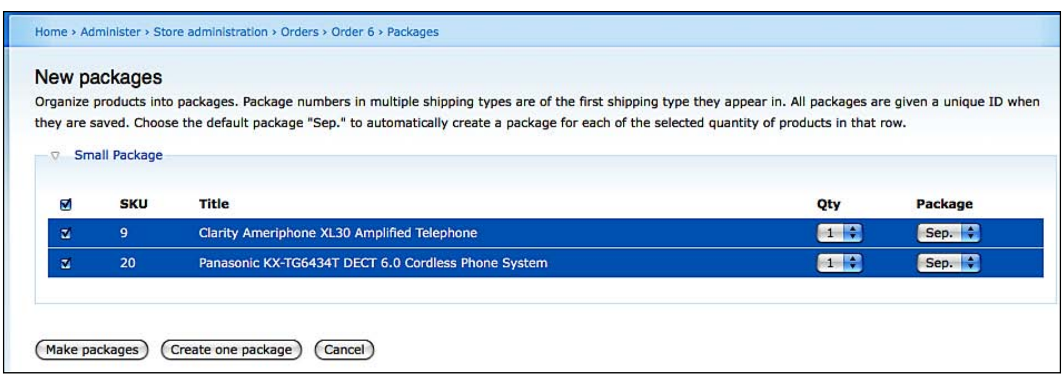
Logging payment action

This is the screen where you can edit the payment actions for an order. In other words, you can administrate your payment transactions, especially if you have not integrated your site payment to a payment gateway and you need manual confirmation from an e-banking back office. To enter payment actions and amounts for specific orders, you need to navigate to **Home | Administer | Store administration | Orders | Order # | Payments**. You will be provided with a list of available payment options, which you have configured as we discussed in *Chapter 6*. You should also provide some reason in a comment to define the transaction.



Creating package option

Packaging is another step for creating the order. Packaging depends on the availability of your products, the estimated delivery time, and the mutual relevance of the products to be shipped. To define a packaging policy for an order, go to **Home | Administer | Store administration | Orders | Order # | Packages**. This makes perfect sense as we can only define packages among items in the same order.

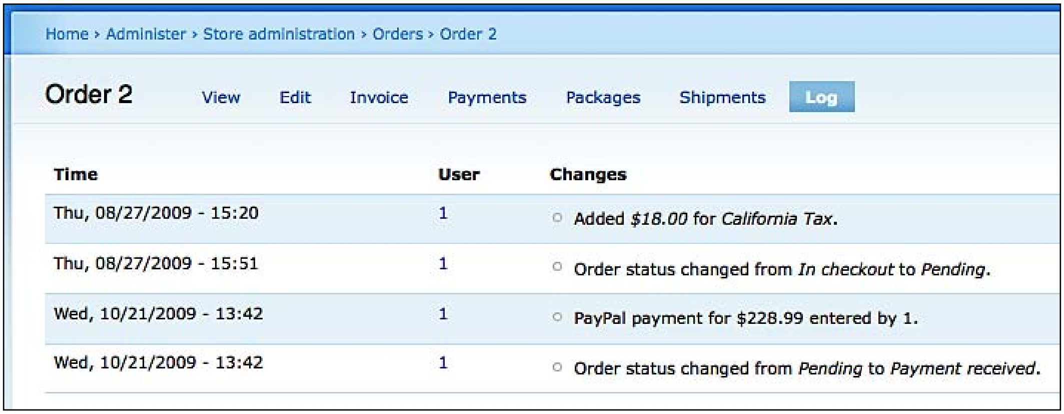


Creating shipping cart

A shipping cart is essential for the order fulfillment process, and can be skipped only if you integrate with an external shipping system. This is the proper way to keep track of things even if you have a small online shop that sells handmade jewelry and you manage the shipping actions by yourself at the local post office. Good tracking and overall control will save you a lot of trouble and misconceptions. Each individual item can be treated separately or as a part of a bundle-shipping cart. You can also break a single order with let's say four items to two shipping carts if one of the items will take some time to collect and the customer wants the others as soon as possible. You can administer your shipping carts at **Home | Administer | Store administration | Orders | Order # | Shipments**.

Viewing order auditing trail logs

As we have mentioned before, a good and well-structured auditing system will sustain proper operation and order fulfillment in the long term. To view all past log actions for a specific order navigate to **Home | Administer | Store administration | Orders | Order # | Logs**.



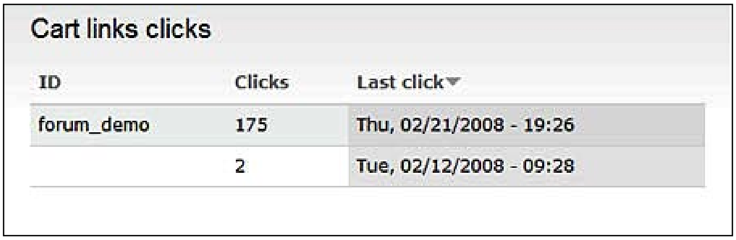
Ubercart reports

We do not have to stress the importance of good trend and sales monitoring for the fructification for your online business. Ubercart comes with a full-feature reporting engine based on Drupal views. To access these reports, navigate to the **Administer | Store administration | Reports** screen. Because Ubertcart's reporting engine depends directly on Drupal's embedded reporting system, you need to activate the Drupal Reports core module. To do so, you should browse to **Administer | Store administration | Configuration | Report settings**. Here, you are able to set the paged table length and the order statuses used in the reports. This is useful, especially if there are a few order statuses where you've already received payment, thus you need to include them in sales reports.

The main core module for reports creates reports for customers, products, and for sales. All the aforementioned reports, except sales summary, can be exported into CSV files.

Cart links

The Cart links module creates a report for clicks on generated cart links. This report displays the various cart link IDs, with the number of clicks each has received, and the time of their last click. To view this report, navigate to **Home | Administer | Store administration | Reports | Cart links clicks**. Cart links is a more targeted alternative for Google Analytics and other traffic monitoring tools and can show you trends and the most viewed products. This is also a good way to calculate conversion ratios for your products because you have both the number of views and the number of purchases.



Cart links clicks

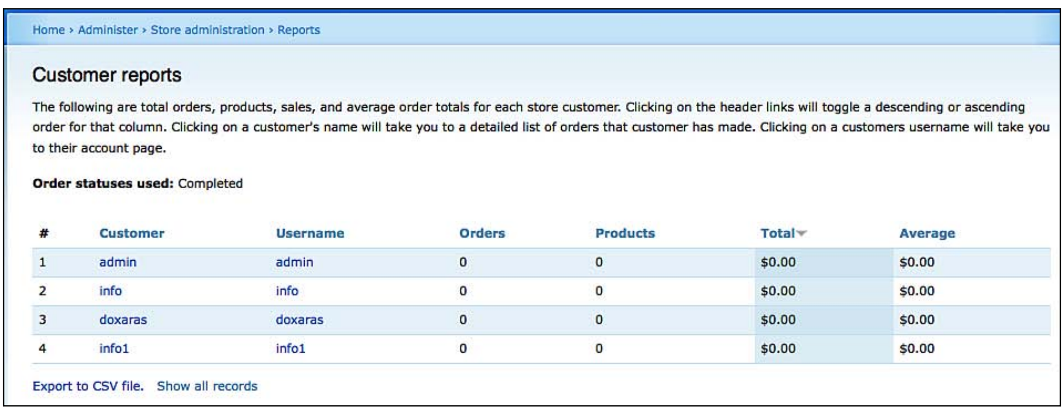
ID	Clicks	Last click
forum_demo	175	Thu, 02/21/2008 - 19:26
	2	Tue, 02/12/2008 - 09:28

Customers

Under Customer reports, you can view customers and end users of your site. For each customer account you'll be able to view the following elements:

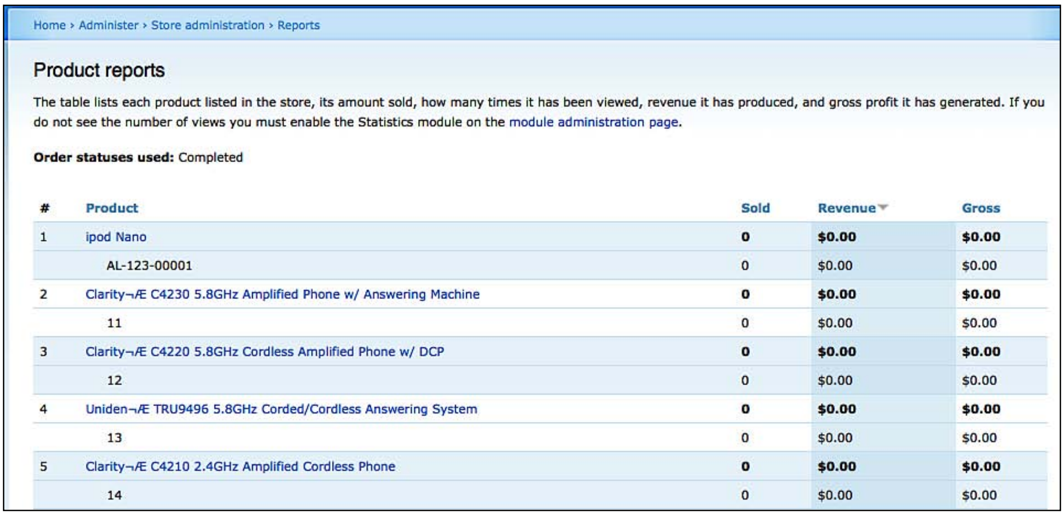
- Customer name
- The billing or shipping name depending on preference
- The username
- The total number of orders placed by the account
- The total number of products ordered
- The total revenue produced for the store
- The average amount of revenue the customer has produced for each order

You can easily navigate to each customer order by clicking on customer names. In the same logic, clicking on a username will take you to the corresponding user account page where the user's order history can be reviewed. Following is a screenshot of the order administration page:



Products

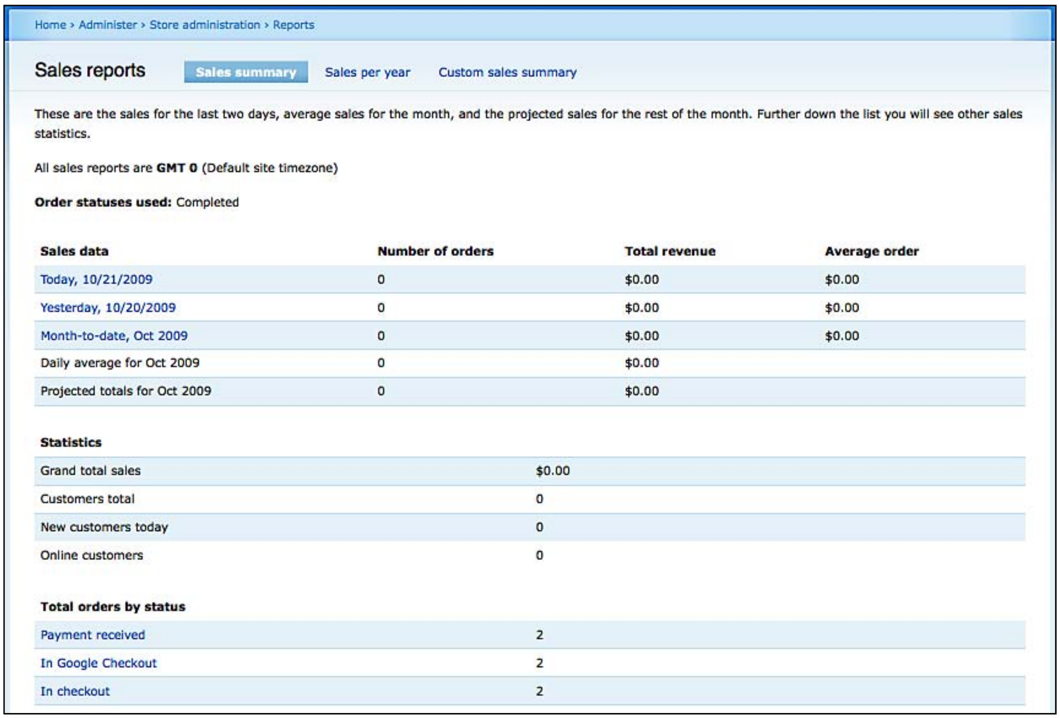
Switching over to the product reports, you will be able to view all products on your site. This view provides the most important information about each product: information such as product name, model, number of views, amount of product sold, revenue produced for the store, and the gross revenue each product has contributed to the shop's total. In the case where your products have attributes and model or SKU adjustments, each product details will be broken down by model or SKU name.



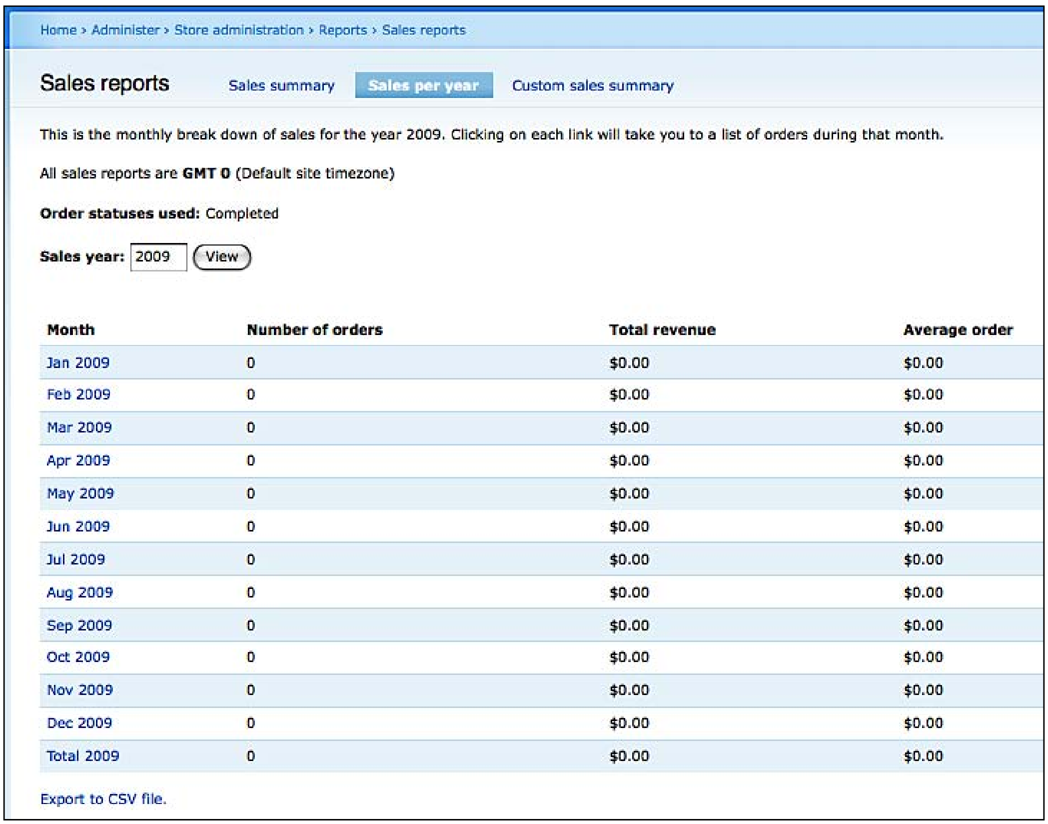
Sales

Sales reports consist of three types. Do not forget that you can make your own reports using Drupal Views module as we will see in *Chapter 10, Optimizing and Promoting Your Store*. Report options for sales are as follows:

- **Sales Summary**. This report creates a general overview of the store sales in a few tables. The first table provides the number of orders, income produced, and the average income per order for the current day, yesterday, and the daily average. Below that, a monthly projection is provided. The second table provides the grand total sales, total customers, new customers for the day, and current logged in customers. The last table provides a breakdown of order statuses for all orders in the store.

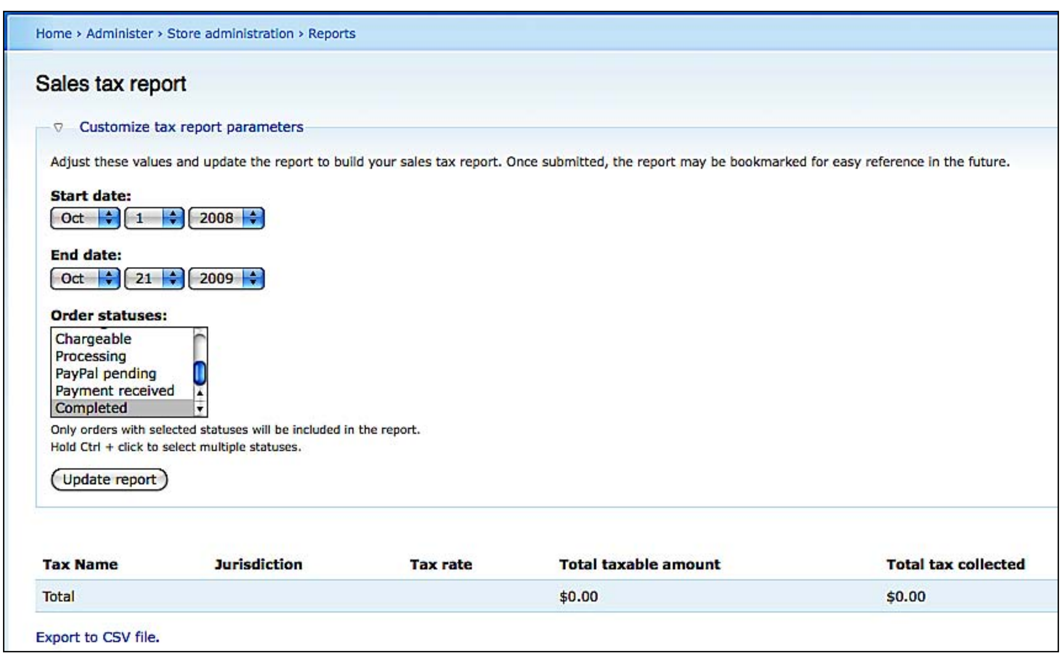


- **Sales per year**: This report creates a list of sales for a specified year. For each month, the number of orders, income produced, and an average order income is displayed. At the bottom the total income for the year is displayed. Clicking on each month will provide current orders placed in that month. This report can be exported to a CSV file if needed. By default, the current year is displayed. Store administrators can view any past years by entering a new year and pressing the **View** button. This view resembles the Excel pivot tables that sales managers are so fond of.



- **Custom sales summary**: An automated report that summarizes orders (by status ordering), products, and revenue produced in a specified time interval. This report also displays sales for the past 12 months of orders according to the specified order statuses in the configuration screen (**Store administration | Configuration | Report settings**).

- **Sales tax**: To get the visual representation of your sales tax, you first need to enter the appropriate search criteria. Once created and submitted, the report can be bookmarked for easy reference in the future. Only orders with selected statuses will be included in the report, so if you need more than one status displayed, use the *Ctrl* key when selecting multiple statuses.



- **Stock**: The Stock module creates a report for all SKUs that are tracked by the module. It shows all tracked SKUs, the product associated with them, stock value, and threshold value (the value at which an admin is notified of the stock level). Like the reports generated by the Reports module, you can export the report to a CSV file. This is the list of product SKUs that are currently active. Stock levels below their threshold have highlighted rows. Toggle the checkbox below to alter which stock levels are shown.
- **Extra Reports**: In addition to Ubercart default reports, having activated the statistics module from core Drupal, you can also have access to a site-wide reporting system. This information provides valuable data about most viewed pages, recently viewed pages, top referrers, top page, and top visitors. To access these reports, browse to **Home | Administer | Reports**.

Customer Relation Management and loyalty programs

Customer Relationship Management (**CRM**) software consolidates all customer information and processes into one, holistic view of each customer. This enables customer-facing employees in sales, marketing, and customer service to make quick, informed decisions on everything from cross-selling and up-selling opportunities to target marketing strategies and effective problem solution. Analytical studies show that on average, companies using CRM software grow their businesses at a rate two or three times faster than those that do not.

So, the time has come to discuss customer relationship management and how Drupal and Ubercart make the integration of all these concepts feasible. Before we start, you need to ask yourself, "Why do I need a customer relationship management program in the first place?" In other words, why do you need to track your clients' purchase trends and offer them deals? The answer is obvious: because you need to increase sales and keep up with your customers' buying trends. But in order to choose the best CRM implementation, we need to have a perfectly clear visualization of why we need to track the profiles of our customer purchases and, of course, the level of analysis we want to make on this collected data. This feature list will eventually help you choose the correct platform customization that suits your needs. It is common opinion that a low-functionality system that provides poor customer information can be as frustrating as an overdesigned platform where you find yourself lost in non-intuitive workflows. So, the correct approach is to trade off between functionality features and operational complexity. Here are some considerations you should pose yourself to:

- How important is it to improve contact management and develop information flows between your customers and your supply chain partners?
- Do you need to integrate systems to make collection and analysis of customer and sales data more effective?
- At what scale you need to improve your customer service? Do you need to implement faster and more efficient internal processes?
- Is it vital to collaborate more closely in your supply chain?
- Does your business depend on loyal customers, or do the majority of your customers just make one or two purchases?

Using a CRM system we can:

- Win more deals by collaborating throughout the sales cycle
- Centralize customer information for all employees to better service customers
- Build profitable relationships with customers through more effective target marketing
- Increase customer loyalty and drive repeat business
- Automate customer-facing business processes and streamline repetitive tasks
- Gain visibility into staff and business performance to make timely decisions

Integration with CiviCRM

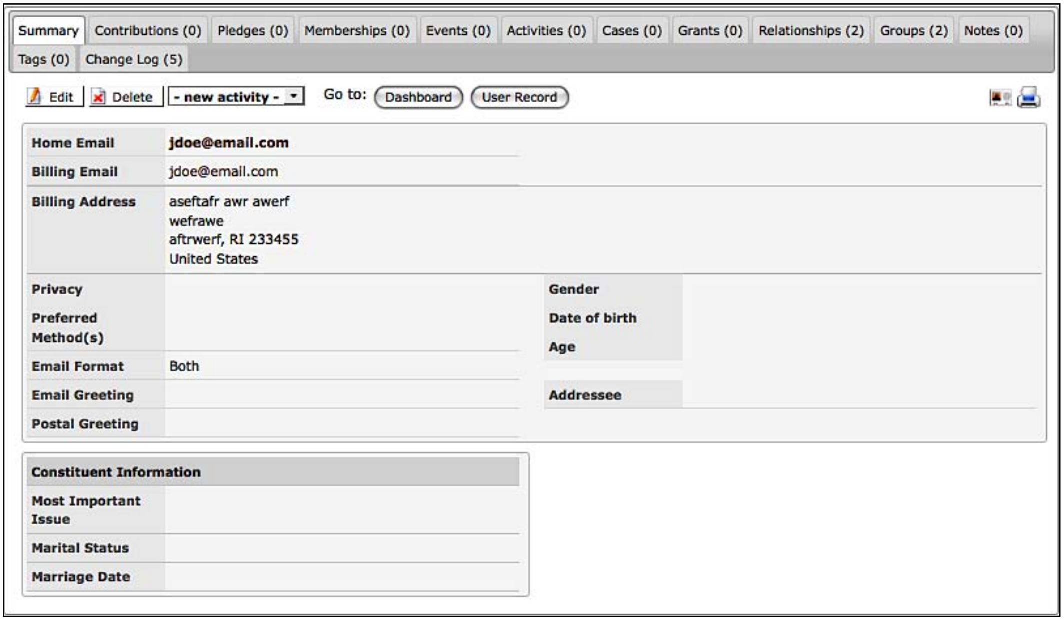
CiviCRM is non-profit community group management software that integrates seamlessly with Drupal. Furthermore, it enables Drupal sites to maintain comprehensive constituent (or customer) relationship management information in a single database, creating efficiencies and new opportunities for your business to better communicate and benefit from relationships with its customers. CiviCRM also integrates with Views module, providing enormous flexibility in exposing CiviCRM user data in custom reports.

CiviCRM installation is not as simple as other Drupal modules, but we will provide a step-by-step guide:

1. Download CiviCRM from `http://sourceforge.net/projects/civicrm/files/`. The current version is `civicrm-3.0.1-drupal.tar.gz`, which only works with Drupal 6.x.
2. Upload and unzip the distribution to your `/sites/all/modules` directory.
3. It will then create and populate a database for CiviCRM as well as create your CiviCRM settings file (`civicrm.settings.php`).

4. Log in to your Drupal site as administrator and navigate to `/sites/all/modules/civicrm/install/index.php`, which is the CiviCRM Installer screen.
5. Ignore the **These database details don't appear to be correct** message, as it is generated due to lack of database credentials at this moment.
6. Click on **Next** and fill in the CiviCRM database settings. It is advisable to use the same database as your Drupal installation because it will make things easier during backup.
7. This is it! Your new CiviCRM installation will be set up and ready to operate. You will then be redirected to the Drupal welcome page.
8. To configure permissions, go to **Administer | User management | Permissions**, and verify that the Roles that you want to have access to CiviCRM have the appropriate permissions checked.
9. Create CiviCRM contacts for existing Drupal users. Once installed, CiviCRM keeps your Drupal users synchronized with the corresponding CiviCRM contact records. The practice is that there will be a matched contact record for each Drupal user record. Conversely, only contacts that are authenticated users of your site will have corresponding Drupal user records.
10. When CiviCRM is installed on top of an existing Drupal site, a special CiviCRM administrative feature allows you to automatically create CiviCRM contacts for all existing Drupal users. To do so, go to **Administer CiviCRM** and click on **Synchronize Users-to-Contacts** in the top group of icons.
11. Review the configuration checklist. The configuration checklist provides a convenient way to work through the settings that need to be reviewed and configured for a new site. You can link to this checklist from the installation success page and you can visit it at any time from **Administer CiviCRM | Configure | Configuration Checklist**.
12. To test your CiviCRM installation, click on the link at **CiviCRM Menu | Shortcuts | Search**, and new individual blocks should appear.

That is it; you now have a synchronized user database between your online store and the CRM system. In addition, you have a fully functioning Ubercart e-shop, so go ahead and explore all CiviCRM's end-user features and begin configuring CiviCRM for your site or organization needs. The objective of this book is the configuration and administration of an electronic shop. So, discussions that refer to CiviCRM's best practices and success stories go beyond the scope of this book. Refer to the CiviCRM Administrator Guide for information on configuration tasks and options.



Ubercart Wish List

You are all familiar with wish lists and product catalogs for future buys. This is a good business practice that enhances community-building actions and widens the user database. It practically allows your customers to create and manage simple wish lists of products in your Ubercart store. Other customers can search for a particular wish list and purchase items from the list on behalf of the wish list creator. The Wish List module adds an **Add to wish list** button beside the normal **Add to cart** button on product pages. To install the module, follow these steps:

1. Download the latest snapshot from `http://drupal.org/project/uc_wishlist`.
2. Uncompress the archive in your Ubercart `contrib` directory: `/sites/all/modules/ubercart/contrib`.
3. Enable the Wish List module under **Ubercart - extra** in the Drupal module's administration page.

That is it! You are ready to use wish list now.

Community-building actions

Community and social media campaigns are the contemporary holy grail of marketing. A large number of conventional media is seeking web publicity and awareness through Internet marketing campaigns. The funny thing is that this has taken the form of a mystic science of SEO and adwords, viral campaigns, and other black or white hat techniques generating a large number of self-appointed consultants, but on the bottom line is a series of good versus bad practices and quality content provision as we will also see in *Chapter 10*. Therefore, in this section, we will focus on community building and sustaining actions, which can easily be turned into the power house of your client base expansion.

Newsletters

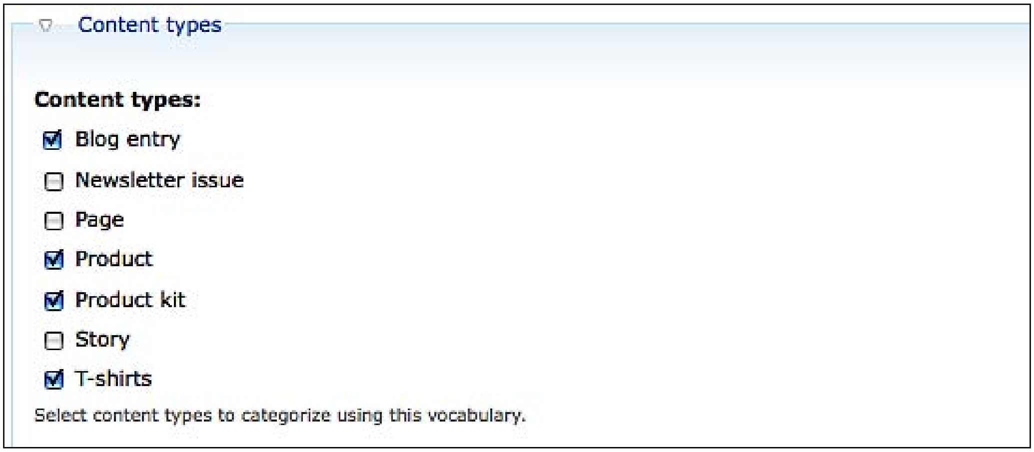
Newsletters are a very powerful sales tool, which helps you interact with inactive customers and establish a sense of continuous activity to the end user. Composing and sending newsletters aligned with the current time frame and context can raise your credibility to end customers. For instance, to achieve frequent communication with your customers, you may compose and send newsletters for the autumn sales period: one for Christmas holiday deals, and another one later on just before the summer vacations. Another interesting aspect is to send tips and, in the case of the electronic shop, send product comparison letters and tips on how to efficiently clean your electronic screens. Your end users will pretty much appreciate free information and tips sent to your subscription database. To implement newsletters in Drupal, we need to download the Simplenews module from `http://drupal.org/project/simplenews`.

1. Create a new directory named `simplenews` in the `sites/all/modules` directory and place the entire contents of this `simplenews` folder in it.
2. Enable the module on the Modules admin page: **Administer | Site building | Modules**.
3. Grant access at the Access control page: **Administer | User management | Access control**.
4. Configure Simplenews from the Simplenews admin page: **Administer | Site configuration | Simplenews**. Select a content type for the newsletter, and select a vocabulary to define which tags will be used in newsletter series.
5. To enable users to manage their subscriptions to the newsletter, use the **subscribe to newsletters** permission. This will enable the subscription form in the Simplenews block and give access to the subscription page at `/newsletter/subscriptions`. To restrict the access for different newsletters only to their audience you may use some of the access control modules such as Taxonomy Access Control Lite.

6. Make use of Simplenews blocks. With the Simplenews block, users can subscribe to a newsletter. For each newsletter, one block is available. Enable the Simplenews block on the Administer blocks page (**Administer | Site building | Blocks**), and visit the block configuration page to configure the block settings. The **subscribe to newsletters** permission is required to access the subscription form in the Simplenews block or to view the link to the subscription form.

Corporate blogging

Corporate blogging is another best practice to enrich your corporate profile. With corporate blogging, you gain more publicity as you share more information with your readers (in your case your customers). Your justified opinions and articles can help you adopt a more professional profile providing solid evidence about your undisputed expertise on the domain you are doing business in. Drupal also provides an RSS 2.0 feed for each blog, so you can give your users the option to subscribe with an RSS reader client program to your blog. You are strongly advised, as in most of the cases in community-building actions, to associate your Ubercart product catalog vocabulary with your blog posts content type. To make this association, navigate to **Administer | Content management | Taxonomy | Catalog**, scroll down to Content type section and check the **Blog entry** checkbox.



If you cannot find the blog entry, you have to activate the core blog module. To activate the blog module follow the next steps:

1. Go to **administer | modules** and check the box next to **blog**.
2. Under **administer | users | access control**, check the **edit own blog** box for each role you wish to maintain blogs.

3. Afterwards, once logged in, each user with the permission to maintain a blog will be able to click on **create content | blog entry** and will see **my blog** (which displays blog entries as other people will see them) in the user navigation block. At the top of each individual blog post, the original blog author will find an **edit** tab.

To add instructions for users on creating their blogs and set workflow options such as **published**, **promoted to the front page**, and so on, follow these steps:

1. Select **Administer | Content management | content types**, then click the **edit** link next to blog entry.
2. Enter your instructions in the **Explanation or submission guidelines:** text field.
3. Set workflow options.
4. Use the **Minimum number of words in a blog entry** setting to specify a minimum length for all blog posts.

Afterwords, you can create new blog post regarding specific products.

Forum for product discussions

The idea of a discussion forum is not as old fashioned at it sounds. Drupal helps you override the drawbacks of serial and static forum threads by allowing you to tag each discussion topic with words from every vocabulary of your portal. Therefore, you have a site-wide discussion forum where each post is associated with a product from your product catalog—do not forget that the catalog is actually a vocabulary—that can be referenced and retrieved with Drupal views.

If you want to provide more insight to the tagging options of the post or allow free tagging features, it would be wise to create a custom vocabulary for this purpose.

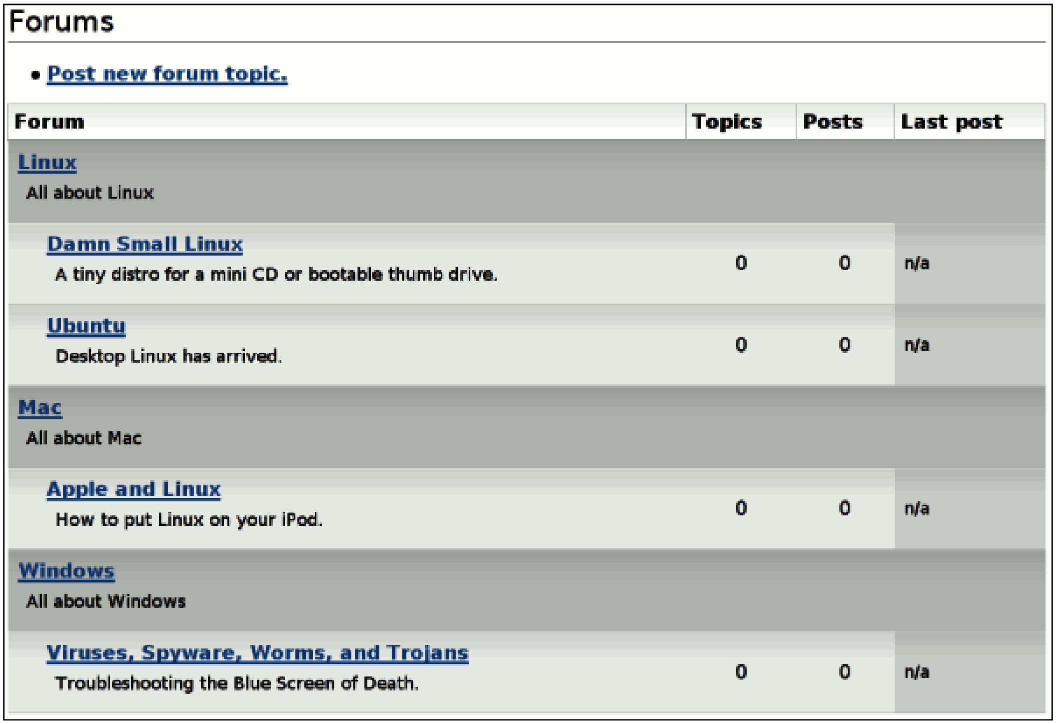
Customers and end users can post topics to forums. Forums can appear below other forums in a hierarchical mode, or can appear in containers. To create a forum follow the procedure described below:

1. Browse to **Administer | Forums** and click on the **add forum** tab.
2. Type in a forum name along with an appropriate description. This description will show up below the forum name when users visit the forum section.
3. Select a parent. You can choose either a forum or a container to be a "parent" for a forum. If there are no containers or existing forums to choose from, you will only see **<root>** as the option, meaning it will be at the top of the forum hierarchy.
4. Select a weight. A higher number means the forum will sink down in the list; a lower number means the forum will rise up.
5. Click on the **Submit** button.

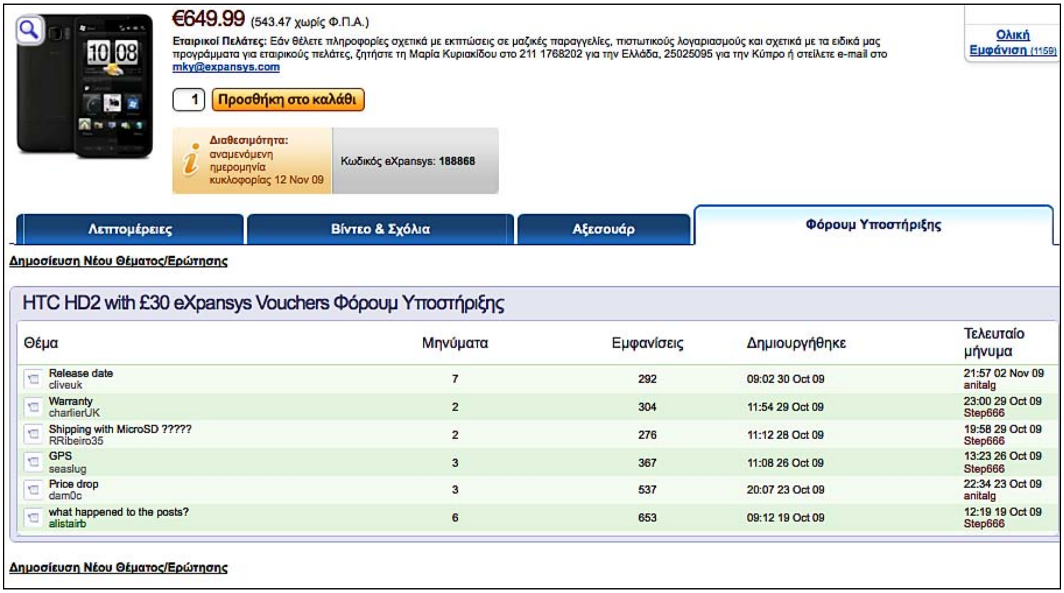
Moreover, to produce a fancy look and feel for your product site, you should also adopt the use of **containers**. Containers are "parent" entities for forums (discussion boards); they are used to group forums together. Customers cannot post topics directly to containers, but on forums associated with them. Your online electronics shop site with only a few forum discussion boards may not need any containers.

1. Click on **Administer | Forums**, and then click on the **add container** tab.
2. Type in a container name and a container description. This description will show up below the container name when users visit the forum section.
3. Select a parent. If there are no containers or existing forums to choose from, you will only see **<root>** as the option, meaning it will be at the top of the forum hierarchy.
4. Select a weight. A higher number means the container will sink down in the list; lower numbers mean the container will rise up.
5. Click on the **Submit** button.

At the end, with the proper content categorization, you will come up with something like the following screenshot:



It is extremely important not to forget to associate a free tagging vocabulary along with a shop product categories vocabulary to each forum entry. This way you will be able to create transparent associations from each product page to all forum discussion associated with the specific product (as we will see in *Chapter 10*), enabling your users to have all the information needed to make a decision for purchase based on the maximum amount of information they can get.

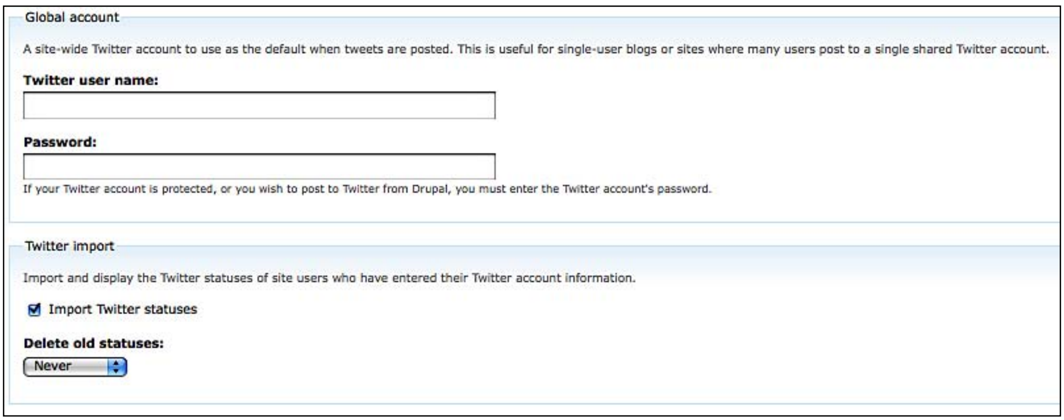


Twitter account

Twitter is gaining more and more publicity these days. Recent Twitter list's features will add up to the hype and make user diffusion more rapid. The idea of having a business account in Twitter and exercising corporate blogging and Twitter posts is appealing to a wider audience, especially in the age group of 15-45. The Twitter module provides out-of-the-box API integration with the Twitter microblogging service.

> Another common practice in social media-aware sites is to use site-wide blocks that display Twitter activity that links to the displayed page. A very nice and small module to achieve this functionality is `http://drupal.org/project/tweetbacks`. Another useful mechanism to achieve automatic Twitter posts is `http://twitterfeed.com`, a web service that transforms your site RSS feed to Twitter posts.

The Twitter module also supports full views integration, allowing administrators to create customized listings of Twitter messages based on time, date, author, and so on. By now readers should be familiar with module installation procedures, so navigate to `http://drupal.org/project/tweeter`, and right after you download the module, upload and unzip to your site's `/sites/all/modules` folder. Activate the module from **Administration | Site building | Modules** and then navigate to **Home | Administer | Site configuration | Twitter Setup** to set up the twitter account information.



In addition, you can choose the content types you want to be posted in Twitter as well as the type of the message for the tweet.

Summary

In this chapter, we have reviewed some of the most important pieces for the success of your online shop. Fast order processing is the key to satisfied customers; transparent business practices are the basis of good business reputation; and customer relationship systems can guarantee customer loyalty and satisfaction. Build up detailed customer profiles in Ubercart with CiviCRM in order to manage to capture and present useful data, including account details, sales histories, service records, campaign responses, web activity, and demographics. By applying information and tips provided in this chapter, you will at least start your online business equipped with the best tools for boosting and enhancing your sales profile. It all comes down to your business incentives and agility to use these systems for the betterment of your own business and to build on top of this solid infrastructure.

8
Customizing the Frontend

After a lot of work, you're just a few steps before launch. Your store is full of products, has great functionality, and offers the client a great browsing experience. The only problem is that you're still using the default Drupal theme, so your website looks too ordinary and rather ugly. People won't be able to perform proper searches and you cannot interfere with the UI by simply promoting items from your stock or communicating special deals and offers.

Customizing the frontend is very important, because you want the look and feel of your site to match your corporate visual identity. You have to use your company logo and colors to create a familiar and friendly environment for your visitors.

You have two main options here:

- You can install a ready-made theme, free or commercial, and use it as is or customize it
- You can create a theme from scratch

In this chapter, we'll show you how to proceed with both alternatives. We'll also show you how to install an administrator theme and some advanced theming techniques for better page structure and layout.

Install a ready-made Drupal theme

We have to admit that Drupal was not famous for its plethora of available themes. Until recently, the Drupal community was focused on developing the backend, debugging the code, and creating new modules. The release of Drupal 6 made theming much easier and helped the theming community to grow. Now, there are not only thousands of Drupal themes, but also dozens of themes designed and customized especially for Ubercart.

Basic principles when choosing a theme

Choosing a theme for your online shop is not an easy task. Moreover, it can be even harder considering that you want to promote specific items from your catalog, you need to change first page items often, and you need to rapidly communicate offers and loyalty policies and other business-related stuff. Ubercart-specific themes mostly target the following special areas:

- Product catalog
- Shopping cart
- Product-specific views

You should keep these layout regions in mind, while going through the following section on theme selection. Before you search for any kind of theme layout, provide your neurons with enough input to inspire you and help you decide. Perform a quick Google search for online shops in your target market to get some inspiration and track down sites that make you, as a customer, feel comfortable during product searching and navigation. If you decide to search for professional help, a list of existing sites will help you to communicate your preferences much more directly.

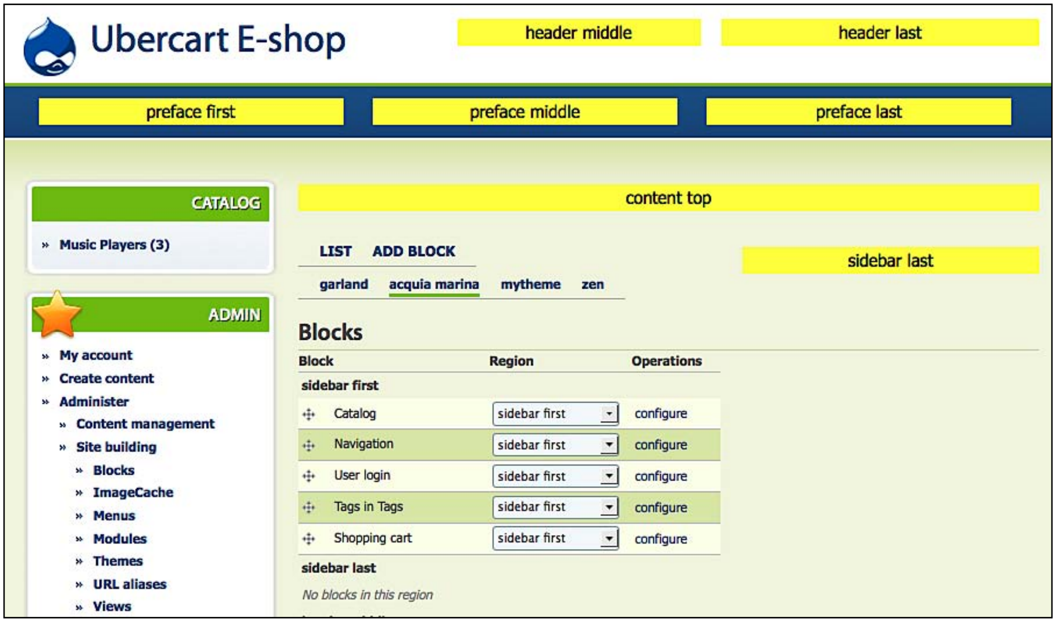
> What better place to search for inspiration and successful practices than Ubercart's live site repository! You will find good practices and see how mostly people like you (without any development background) have solved all the problems that might occur during your search for themes.
>
> `http://www.ubercart.org/site`

Next we describe the main user interface components that you should keep in mind when deciding for your online shop:

- **Number of columns**: The number of columns depends on the block information you want to provide to your end customers. If you need widgets that display on every page, information about who bought what, and product or kit suggestions, go with three columns. You will find a plethora of two-column Drupal themes and many three-column Drupal themes, while some of them can alternate between two and three columns.
- **Color scheme**: From a design perspective, you should choose a color scheme that matches your company logo and business profile. For instance, if your store sells wooden toys, go with something more comic such as rounded corners, but if you are a consulting firm, you should go with something more professional. Many themes let you choose color schemes dynamically; however, always keep in mind that color is a rather easy modification from the CSS. You can get great color combination ideas from COLOURlovers online service (`http://www.colourlovers.com/`) that match your logo and business colors.

Be careful though. If you choose a complex theme with rounded corners, lots of images, and multiple backgrounds, it may be difficult to modify it.

- **Drupal version**: Make sure the Drupal theme you choose is compatible with the version of Drupal you are running. Before using a Drupal theme, look up notes on the theme to see if there are any known bugs or problems with it. If you are not a programmer, you do not want a Drupal theme that has open issues.
- **Extra features**: Many Drupal themes expose a large set of configuration options to the end users. Various functionality such as post author's visibility or color scheme selection are welcome for managing the initial setup. Moreover, you can change appearance in non-invasive ways for your online marque.
- **Regions available**: We have discussed column layouts, but for the Drupal template engine to show its full capabilities and customization, you definitely need multiple regions. The more regions, the more choices you have for where to put blocks of content. Therefore, you can have space for customizing new affiliate ads for instance, or provide information about some special deals, or even configure your main online shop page, as we will see in the next section.



- **Further customization and updates**: When you choose your theme, don't just keep the functionality of version 1.0 in mind, but consider all of the future business plans for approaching your target market and raising sales figures. Make a three-year plan and try to visualize any future actions that should be taken into account from day one. Although you can change themes easily, you are better off choosing a more flexible theme ahead of time than having to change the theme as your website grows.

Always bear in mind that the famous razor of Occam also applies to online shop theme design. Keep it simple and professional by choosing simple layouts, which allow ease of use for the end user and ease further customize designs and themes (changing colors, adding a custom image header, and so on).

Before you start, clearly define your timeline, risks, total theme budget, and skills. Theming is usually 25-40% of the budget of an entire online shop project. Drupal's theming engine closely integrates with actual functionality and many features are encapsulated inside the theme itself.

There are a number of different ways in which you can get yourself the best theme for your online store. We will go through all these approaches with useful comments on what options best suits your needs.

Do it yourself

If you have moderate skills in XHTML/CSS and you also know how to operate a graphics editor, you may very well be able to create your own Drupal theme. Knowing how Drupal's bits and pieces work is more important than any kind of advanced PHP knowledge. If you have decided to experiment and do it yourself for the first time, probably it is best to install and modify a ready-made theme.

Total time spend on theme selection

Time will vary from one week to one month, especially if you create your theme from a Photoshop template (PSD) and cut the layers to XHTML/CSS. It also depends on the number of different page themes you create.

Budget

If you do not count your personal time in the project budget, then total budget spend is zero.

Places to look

Drupal theme documentation (`http://drupal.org/theme-guide/6`) and especially the page **Anatomy of a Drupal theme** (`http://drupal.org/node/171194`) will get things started.

Get it for free

The Drupal theming community is expanding and many themes are ported from other popular CMSs like Wordpress and Joomla!. This is the fastest way and you are advised to go with it if you are short on budget or time to spend.

Total time spend on theme selection

Time will vary from one hour to two days max; most of the time will be spent on installing a theme and then changing it, in order to find the most suitable one for your site.

Budget

The budget would be the same as if you were to do it yourself.

Places to look

You can visit the following pages:

- **Drupal Themes** (`http://drupal.org/project/Themes`): Here you can find the most updated and fullest repository of ready-to-use Drupal themes. Use of search filters is strongly advised; you can search for popularity, recent activities, and target Drupal version.
- **Theme Garden** (`http://themegarden.org/drupal6/`): Theme Garden is a project that provides useful information and insight about Drupal themes along with theme previews, a very clever feature that saves time.

Buy it

You can go with the nearly free theme route, downloading a theme from an inexpensive theme seller, or you can buy a premium, high-quality theme. The former will get a site up and running for little to no cost, but you may find yourself limited by the number of regions, quality, and features of the inexpensive themes.

Total time spend on theme selection

Time will vary from one hour to two days max; most of the time will be spent on downloading and installing a theme, and then again changing to new one to suit your needs.

Budget

From $30 for a basic theme, to $400 for a premium theme, along with a few hundreds of dollars if you choose to get a Drupal themer to alter an already-purchased theme in order to suit your needs and stand out from similar templates.

Places to look

You can visit the following pages:

- `http://www.websitetemplates.bz/drupal-themes.html`
- `http://topdrupalthemes.net/`
- `http://www.cmslounge.com/templates/drupal-6-themes.html`

> You can find high-quality themes especially designed for Ubercart at:
> `http://www.topnotchthemes.com/drupal-ubercart-themes`

Hire a designer

Finally, you can hire a designer or themer to make a completely custom Drupal theme for your site. This will give you the maximum control, but is the most time-consuming and expensive option. There are many Drupal design and theming shops out there. A custom theme involves having a design created for your site from scratch, usually by a designer who then provides mockups (or mockups and CSS) to a Drupal themer.

Total time spend on theme selection

Time taken may vary from one week to two months depending on the communication between the themer, the designer, and you.

Budget

Trying to provide a rough estimation about the cost of a custom theme is not straightforward and depends on a number of factors. The bare minimum for an extremely basic, single-page design would be a few hundred dollars. The average cost for a small business site design plus Drupal theme would be in the lower range of thousands. This can vary widely due to things like number of unique pages, whether or not Drupal module support required, if you need a logo, and so on, but hopefully that gives you a general idea.

Places to look

There are a small but rapidly emerging number of Drupal theming shops out there. We suggest you make a quick search and ask for references and portfolios. Be careful to ask for XHTML/CSS-strict themes and make a detailed list of what you want developed before signing the contract to save both parties from misconceptions.

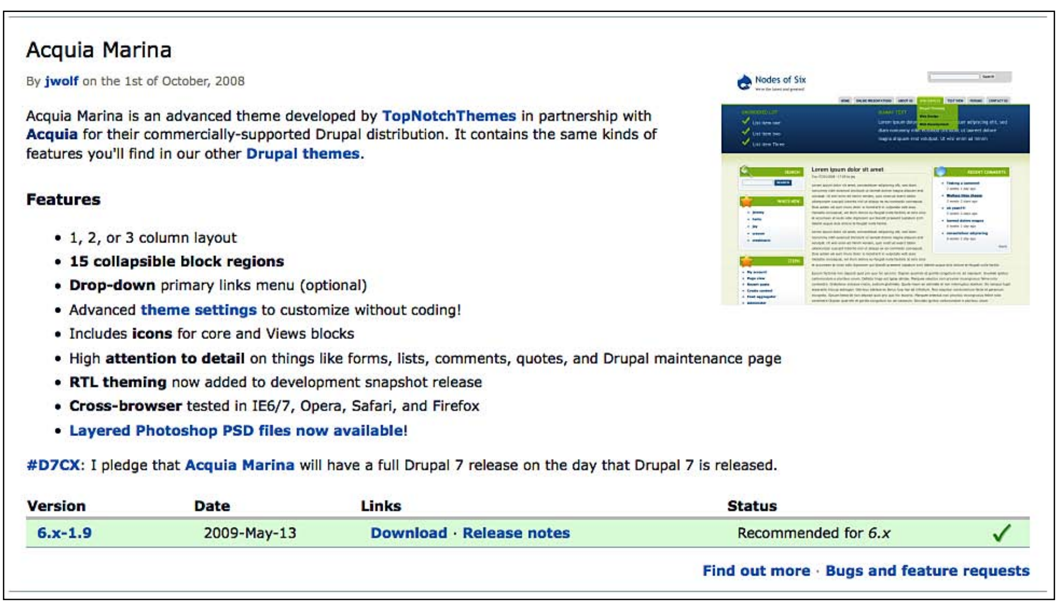
Don't forget to take a look at online marketplaces, such as `http://www.elance.com` or `http://www.odesk.com`.

> Another recent and extremely helpful option, especially for agile and rapid theme development, is Artisteer (`http://artisteer.com`). Artisteer is a web design automation product that instantly creates Drupal themes.

Installing a Drupal theme

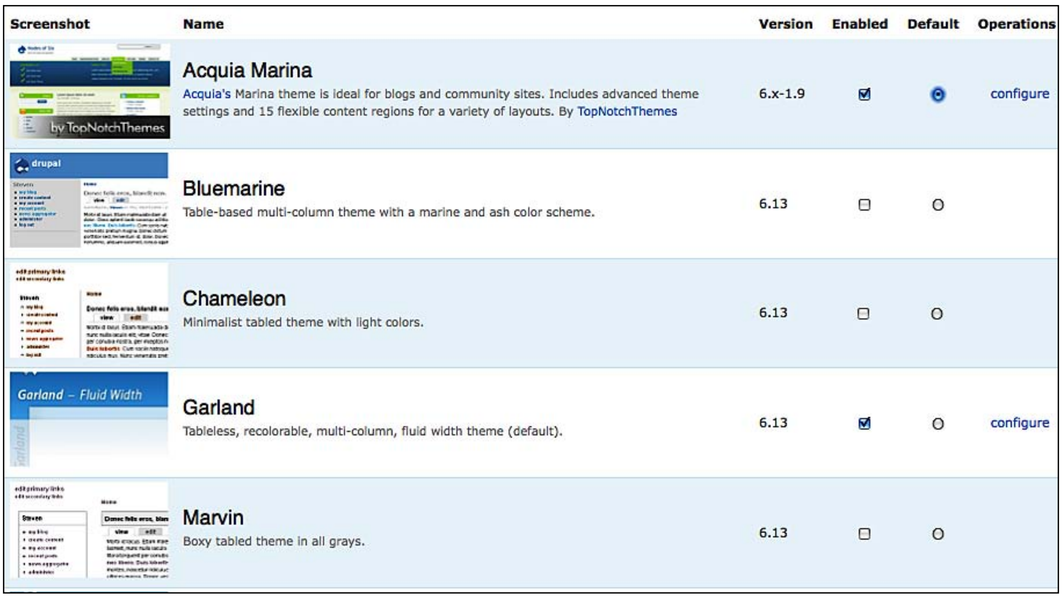
After you have decided the best theme to go along with your online store, you should follow these steps to install it. This is a rather straightforward procedure, very similar to the one performed in *Chapter 2, Installation of Drupal and Ubercart*, during module installations:

1. We will go through installing the second most popular Drupal theme according to Drupal.org statistics. Go to `http://drupal.org/project/acquia_marina` and click on the download link next to the **6.x-1.9** version.



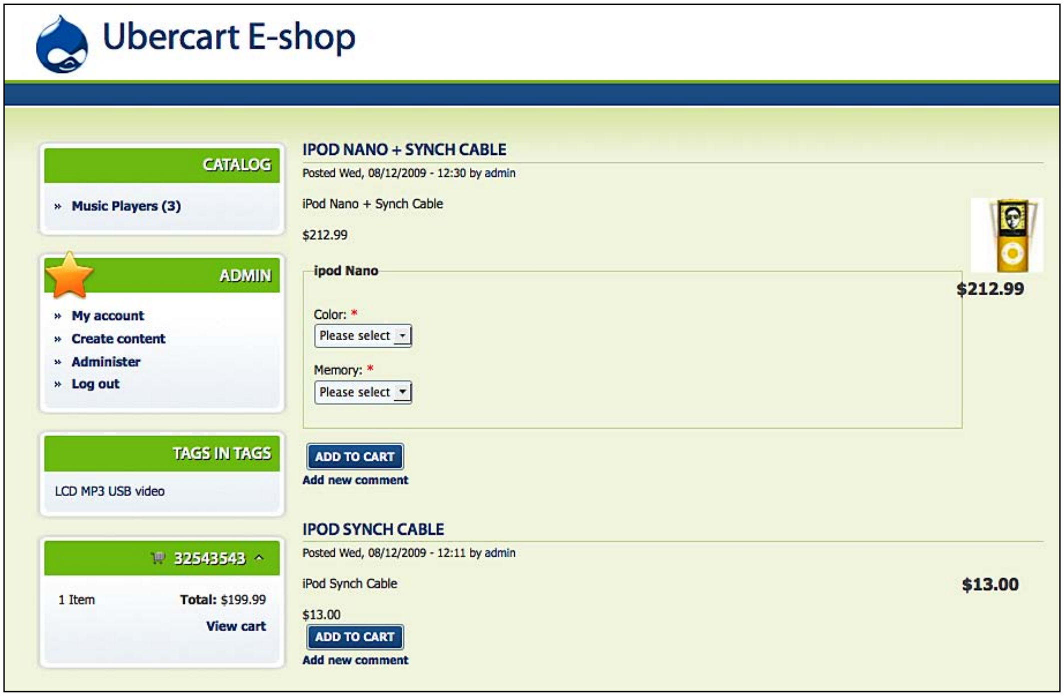
2. Secondly, you have to upload the theme to your local site theme directory using FTP. Theme should be unzipped and placed in the `[DRUPAL ROOT]/sites/all/themes` path of your Drupal site installation. See *Chapter 2*, for more details.

3. After you extract the zipped file in the `themes` directory, navigate to the themes administration screen (**Home | Administer | Site Building | Themes**), or hit a page refresh if you are already there. Select the theme that you want to install and press **Save configuration**.



Screenshot	Name	Version	Enabled	Default	Operations
by TopNotchThemes	**Acquia Marina** Acquia's Marina theme is ideal for blogs and community sites. Includes advanced theme settings and 15 flexible content regions for a variety of layouts. By TopNotchThemes	6.x-1.9	☑	◉	configure
drupal	**Bluemarine** Table-based multi-column theme with a marine and ash color scheme.	6.13	☐	○	
	Chameleon Minimalist tabled theme with light colors.	6.13	☐	○	
Garland – Fluid Width	**Garland** Tableless, recolorable, multi-column, fluid width theme (default).	6.13	☑	○	configure
	Marvin Boxy tabled theme in all grays.	6.13	☐	○	

After you apply the theme, your online shop will look like the following:



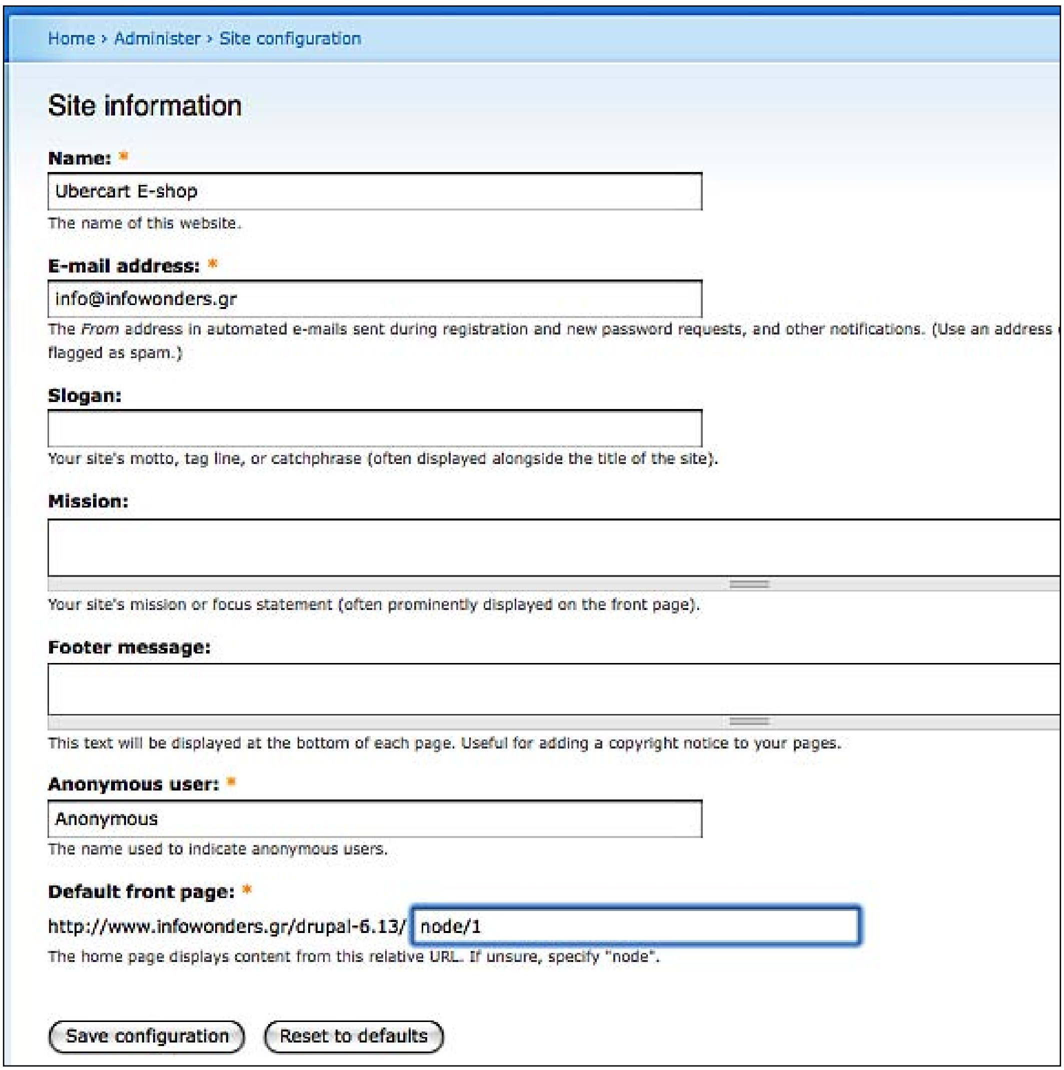
For those of you coming from another CMS such as Joomla!, where you are more familiar with a separate administration interface, you may also simulate this behavior in Drupal. Another good reason for doing so is that, this way you will be able to perform administrative tasks without any distraction that eye-catching e-shop features will pose. To do that, go to **Home | Administration | Site configuration | Administration Theme** and choose a light-weighted theme, such as **Garland**. This theme will be activated only when you authenticate using a user with administration privileges.

Customizing a theme

In this section, after we have elected our primary theme, we will go step-by-step customizing it and making it suit our business need. These configurations are necessary even if you choose to hire a designer or buy a ready-made theme.

Changing basic elements

Every Drupal theme using the template engine produces HTML code from Drupal core objects. Therefore, some content of the final HTML code generated is actually site-wide property such as site slogan, mission, and site name. We will have to change Drupal default settings and provide our business details. To do this, go to **Home | Administer | Site configuration** and edit the fields as we describe next. If you do not want to provide specific information, for instance if you do not have a corporate slogan, you need not fill this option. Nothing will appear if the attribute is not set to the main page of your online shop.



You can edit the following elements:

- **Name**: This is your site's name and will be displayed in the site name theme section and can also be a part of the HTML `<title>` element.
- **E-mail address**: A valid e-mail address for your website, used by the mailer functionality during registration, new password requests, notifications, purchases, and all mail communication to your users. E-mail server details that your site uses are placed in your `php.ini` file. The majority of web hosting solutions have a preconfigured mail server environment and you will not have to deal with it.
- **Slogan**: The slogan of your website. Some themes display a slogan when available. It will also display in the title bar of your user web browser, so if you decide to choose one, do it wisely.
- **Mission**: Your site's mission statement or focus. Your mission statement is enabled in your theme settings and requires that the theme supports its display.
- **Footer**: This text will be displayed at the bottom of each page. Useful for adding a copyright notice to your pages. You can also use HTML tags to include an image for instance.
- **Anonymous user**: The user name for unregistered users is "Anonymous" by default. Drupal gives you the option to change this to something different according to your target user group (for example "New Customer").
- **Default front page**: This setting gives site administrators control over what Drupal-generated content a user sees when they visit a Drupal installation's root directory.

We quote from the Drupal documentation section for site configuration:

> *This setting tells Drupal which URL users should be redirected to. It's important to note that the URL is relative to the directory your Drupal installation is in. So, instead of*
>
> *"http://www.example.com/node/83"*
>
> *or*
>
> *"http://www.example.com/drupal_installation_directory/node/83,"*
>
> *it is only necessary to type "node/83". For those not using clean URLs, note that there is no need to type in "?q=" before typing the URL.*

By default, the "Default front page" is set to "node," which simply displays articles that have been "Promoted to front page." Note that when changing the "Default front page" to something other than "node", nodes that are "Promoted to front page" will no longer appear on the front page. They can however, still be viewed by visiting the relative URL path "node".

If the path specified is not a valid Drupal path the user will be confronted with a "Page not found" error. It is not possible to redirect users to any web documents (e.g. a static HTML page) not created by the Drupal site.

In addition, as we will see in *Chapter 9, User Interface Enhancement Techniques*, default front page definition ID is also a very good way of customizing Drupal front pages.

Changing page properties and layout

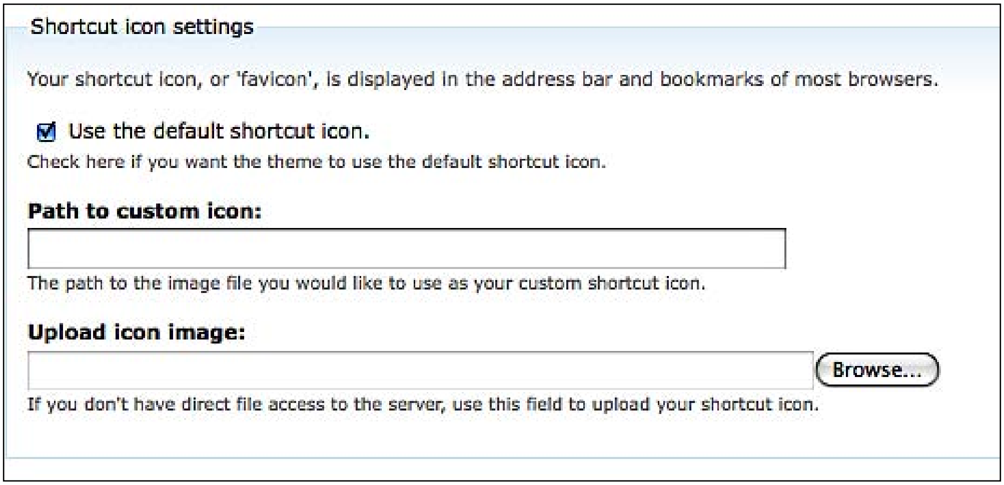
As we have seen, each theme comes with its own special options along with Drupal's default configuration. These options let us interfere with most of theme layout settings and perform in-depth customization. In this section, we will refer to these options one by one. To do so, we should navigate to **Home | Administer | Site Building | Themes | Configure**.

Theme global settings

In every fresh Drupal installation, Garland is enabled as the default theme. Click on the **Configure** tab next to **List**, and you will see the global settings for themes. Apparently these settings apply to all themes. On this screen, you can configure the following options:

- **Toggle display**: It handles visibility for specific page elements. You have to configure them in the **Site configuration** page first. (see previous section).
- **Logo image settings**: If toggled on, the following logo will be displayed. You have the option to upload a new logo image replacing the old one.
- **Shortcut icon settings**: Your shortcut icon, or **favicon**, is displayed in the address bar and in the bookmarks of most browsers. Drupal also provides the option to upload your own custom favicon to replace the smiling Druplicon. We like Druplicon, but your site should display your identity and a custom favicon is indication of good design.

  If you want to easily create a favicon for your site, you can use an online tool like FavIcon generator, available at `http://tools.dynamicdrive.com/favicon/`.



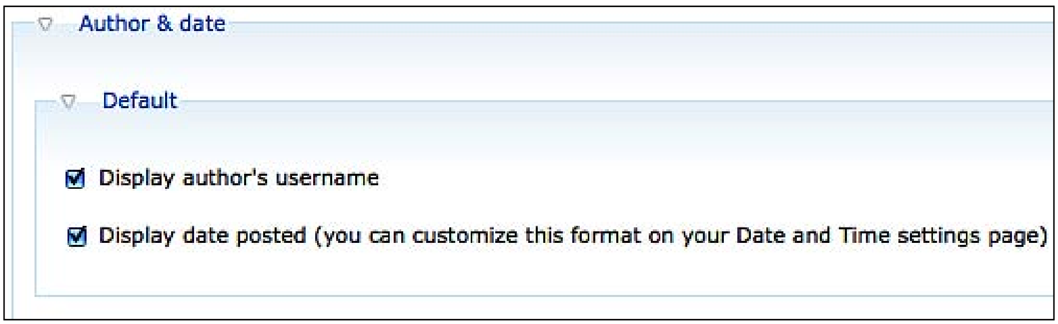
- **Display post information on**: Enable or disable the *submitted by Username on date* text when displaying posts. This is a very useful option and should be used in non-content or community-oriented Drupal installation such as online shops. It is strongly advisable to uncheck this option.

Theme-specific configuration

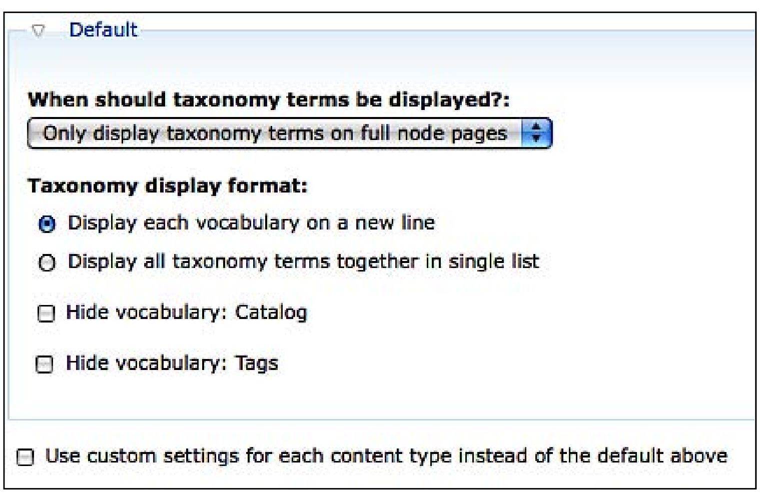
Next to the **Global settings** subtab is a list of enabled themes. On a new installation, only Garland will be listed. Select Garland for now. At this moment, you should feel free to experiment.

- **General Settings**: These are the settings concerning visibility of certain UI elements.
- **Mission Statement**: This option lets you choose where to display your mission statement.
- **Breadcrumb**: This option lets you choose whether or not to display a breadcrumb trail. If you have a product catalog, this is a very useful feature for your site. For instance, if a customer navigates to the iPod page, the breadcrumb trail will provide the full navigation, **Electronics | MP3 | Apple**. Therefore, your customer can go directly to the MP3 section and see other products, such as Creative or Sony walkmans.
- **Username**: Display "not verified" for unregistered usernames.

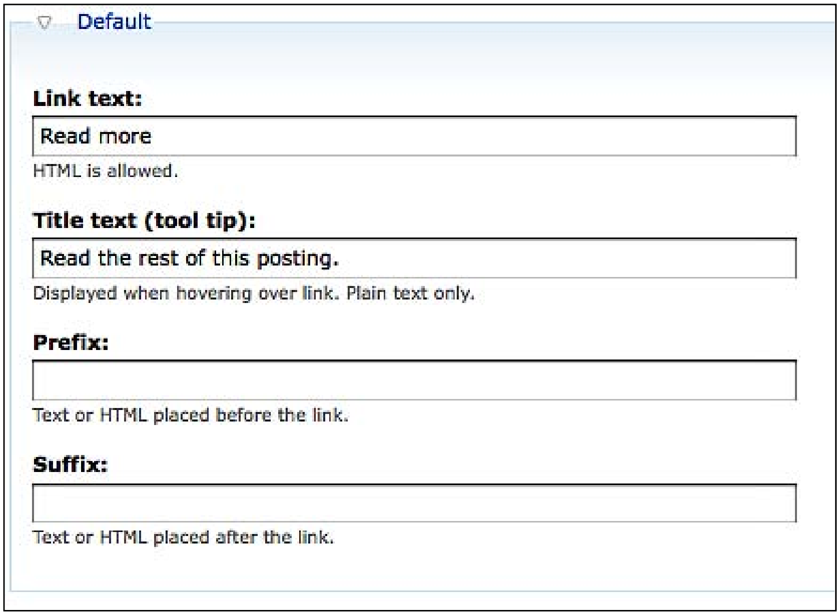
- **Node Settings**: Adjust how and which information is being shown with your content. You can modify these settings, so they apply to all content types, or check the **Use content-type specific settings** box to customize them for each content type. For example, you may want to show the date on stories, but not pages.
- **Author & Date**: Check accordingly to display or not the author's username and the date the node is posted. Remember that you can globally configure this display from the default theme setting page. The date format is also customizable through the Date and Time settings page **Home | Administer | Site configuration | Date and Time**.



- **Taxonomy Terms**: This is a very important screen, because as of Drupal 6, there are so many theming and display options in theme configuration. You can customize the way your taxonomy displays and provides cross-selling and product suggestion functionality, a feature that we will discuss in *Chapter 10, Optimizing and Promoting Your Store*.



- **Links Read More**: This setting provides options to change the display texts for the **Read more** functionality below node teasers or views.



- **Comments Full Content**: The link when the full content is being displayed. HTML is allowed.
- **Comments Teaser**: Customize new comment links. You can choose the label end users will see when they are called to comment on content.

In all the fields we discussed, you can use HTML fragments along with performing label translations through Drupal translation interface.

> We will not be discussing search engine optimization (SEO) settings, as this section will be addressed in *Chapter 10*. Theme development settings, on the other hand, are strictly the developer's concern and they handle theme registry parsing, a procedure that searches for all template files from the filesystem. We should leave this checkbox unchecked.

Creating a theme from scratch using the Zen theme

In the previous section, we showed you the easy way to install and customize a ready-made theme. This solution is good enough for many shop owners, but if you want to use a unique design for your store, the only solution is to build a theme from scratch. We are going to use the Zen Theme, maybe the most popular theme for Drupal. Zen is actually not just a simple theme, but rather a theming framework, because it allows the creation of subthemes. Using a subtheme, we can use all of the great code of Zen and apply only our design customizations, using some simple tools and writing only a few lines of code. So, don't be afraid but enjoy the process. Just think how proud you'll feel when you will have finished your amazing frontend for your store. You don't have to be a programming Ninja, all you have to know is some HTML and CSS. If you have no programming experience at all, you can follow some very interesting tutorials at `http://www.w3schools.com/`.

The tools

We are going to use some simple and free tools, which are easy to download, install, and use. Some of them are extensions for Firefox, so if you are not using this particular browser, you have to download it first from `http://www.getfirefox.com`.

Firebug

This is the first extension for Firefox that we are going to use. It's an open source and free tool for editing, debugging, and monitoring HTML, CSS, and JavaScript in our web pages. Using Firebug, you can understand the structure of an Ubercart page and inspect and edit HTML and CSS on the fly. To install it, go to `http://getfirebug.com/`, skip the terrifying bug, and click on **Install Firebug for Firefox**. You will be transferred to the Firefox add-ons page. Click on **Add to Firefox**. A new window opens with a warning about possible malicious software. It's a common warning when you try to install a Firefox extension, so click on **Install now**.

When the download is completed, click on **Restart Firefox**. When Firefox restarts, Firebug is enabled. You can activate it by clicking on the little bug icon at the bottom-right corner of the window.

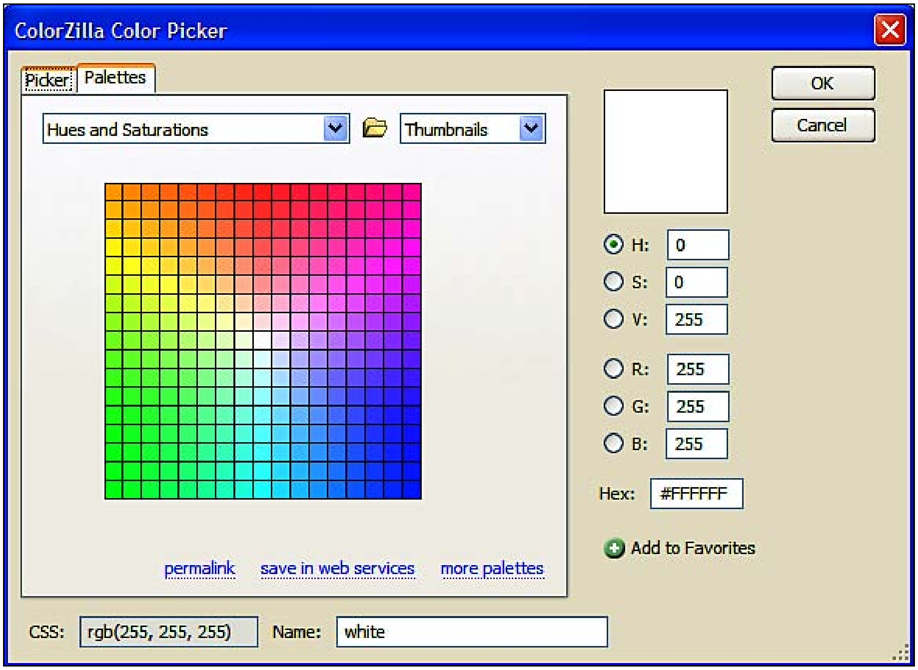
When Firebug is activated, it splits the browser window into two parts. The top part is the normal page and the bottom part shows the HTML or CSS code of the whole page, or for a selected element. There, you can inspect or edit the code, make tests, and try different alternatives.

It is now possible to use Firebug in Internet Explorer, Opera, or Safari, using Firebug Lite. It's a small JavaScript file and you can download it from `http://getfirebug.com/lite.html`.

ColorZilla

ColorZilla is also a Firefox extension. It provides useful tools related to color management, such as eyedropper, color picker, or palette viewer.

You can download it from `http://www.colorzilla.com/firefox/`. Click on **Install ColorZilla**. A new window opens with a warning about possible malicious software, like we saw in the Firebug installation, so click on **Install now** and then **Restart Firefox**. When Firefox restarts, ColorZilla is enabled. You can activate it by clicking on the little eyedropper icon at the bottom left corner of the window.



A code editor

We need it to write and edit our template and CSS files. There are many freeware applications, such as HTML Kit (`http://www.chami.com/html-kit/`) and Webocton (`http://scriptly.webocton.de/9/34/start/englishpage.html`), or commercial applications, such as Ultraedit (`http://www.ultraedit.com`) or Coda (`http://www.panic.com/coda/`).

Characteristics and advantages of Zen theme

According to its creators, "Zen is the ultimate starting theme for Drupal", and this is the truth. It is appropriate not only for the site administrator who wants to create a new simple subtheme, but also for the advanced developer who needs to create a complex theme for a large portal. It uses standards-compliant and semantically correct XHTML code and allows the creation of a new subtheme, just by copying some files, changing their basic information, and editing a CSS file.

The main advantages of Zen theme are:

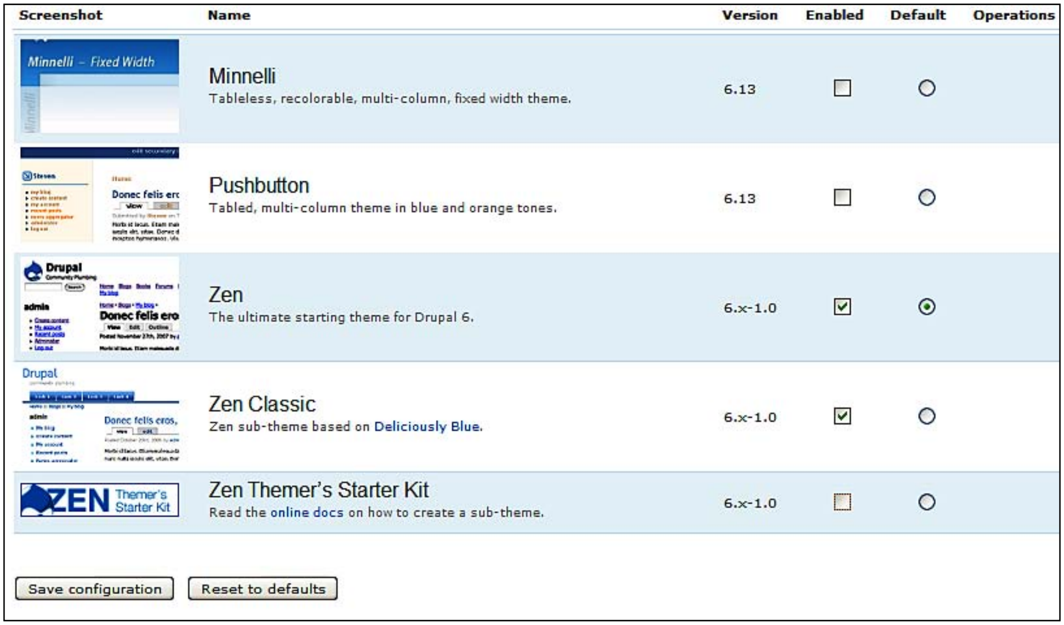
- It has great online documentation.
- It has a large community, providing amazing support for every problem.
- It has a small learning curve.
- Supports fixed or liquid layout.
- Supports layouts with one, two, or three columns.
- When you hover the cursor over a block, it provides an **edit** link, allowing you to edit the content of the block on the fly and return back to the page where you were working. There is no need to go to the admin pages just to edit a block.
- Extra body classes, such as `.front` or `.logged-in`, give the developer the opportunity to create custom themes for separate pages, just by targeting those classes. There is no need to mess up with custom Drupal template files.
- The `print.css` file optimizes the content of a page for printing.
- The `ie.css` file makes debugging for Internet Explorer a much easier process.

Installing Zen theme

The first step is to install Zen theme. The procedure is exactly the same as with every other Drupal theme.

1. Go to `http://www.drupal.org/project/zen` and download the latest release for Drupal 6.
2. Unzip the file and upload it to the `../sites/all/themes` folder of your site.
3. Go to **Administer | Site Building | Themes**.
4. There, you will see three new available themes: **Zen**, which is the default theme; **Zen Classic**, which is a ready-made Zen subtheme in blue color; and **Zen Themer's Starter Kit**, which is actually the framework that allows us to create a new custom subtheme.

5. Enable the Zen theme and set it as the default theme.



Screenshot	Name	Version	Enabled	Default	Operations
Minnelli – Fixed Width	Minnelli Tableless, recolorable, multi-column, fixed width theme.	6.13	☐	○	
	Pushbutton Tabled, multi-column theme in blue and orange tones.	6.13	☐	○	
Drupal	Zen The ultimate starting theme for Drupal 6.	6.x-1.0	☑	◉	
Drupal	Zen Classic Zen sub-theme based on Deliciously Blue.	6.x-1.0	☑	○	
ZEN Themer's Starter Kit	Zen Themer's Starter Kit Read the online docs on how to create a sub-theme.	6.x-1.0	☐	○	

Save configuration | Reset to defaults

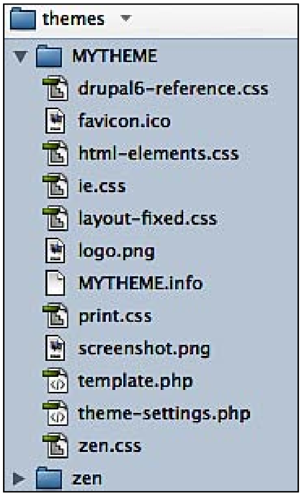
6. Click on **Save configuration**.

Creating and customizing a Zen subtheme

Creating a Zen subtheme is a standard process with specific steps. If you just want to add some simple colors by using CSS, or if you plan to slice and convert a PSD file to a complex Drupal theme, the process is almost the same. You have to create a Zen subtheme and make all the modifications there. There is a simple question that we hear all the time: Why do we have to create a Zen subtheme, isn't it simpler to just edit the files of Zen theme? Well, maybe it's the easiest path, but you may face the following problem: Zen is an active project and every few weeks there is a new version including bug fixes and new features. If you've changed the source code of the original Zen theme and you install a new version, your old files will be overwritten and all your changes will be lost. On the other hand, the files of the subthemes are stored in a subfolder, isolated from the core files of Zen theme, so you can install updates without any problems.

This is the first time and maybe the only one in this book, where you have to edit the code of some files, or create new code. We're going to show you the step-by-step process, explaining every step in detail and analyzing every line of code. So, let's start and create your first Ubercart theme, named `MYTHEME`.

1. Copy the `STARTERKIT` folder, located inside the `Zen` folder, to `..sites/all/themes`.
2. Rename the `STARTERKIT` folder to `MYTHEME`.
3. Copy the following files from the `Zen` folder to the `MYTHEME` folder: `html-elements.css`, `layout-fixed.css`, `print.css`, `ie.css`, `zen.css`. These are the basic CSS files needed to customize your theme. `html-elements.css` contains the basic format for the main HTML elements, such as fonts, headings, lists, links, and tables. `layout-fixed.css` is used to create a table-free, three-column, two-column, or single-column layout depending on whether blocks are enabled in the left or right columns. `print.css` is called when we want to print a page. It removes all the unimportant elements and sends a simpler version of the page to the printer. `ie.css` is a CSS file especially for Internet Explorer. Finally, `zen.css` is our main CSS file. It already contains the basic classes and IDs, but as you proceed, you can add your own and edit the existing ones.
4. Rename `STARTERKIT.info` to `MYTHEME.info`.
5. Open `MYTHEME.info` and replace all `STARTERKIT` instances to `MYTHEME`. In Drupal, every theme has an `.info` file, which is actually a configuration file. There are some required options, such as name, core, and engine and some optional ones, such as description, screenshot, base theme, regions, features, stylesheets, scripts, and PHP.



This file is extremely important, so we are going to analyze together every line of code.

```
; $Id: MYTHEME.info,v 1.13.2.1 2009/02/13 06:13:18 johnalbin Exp $
```

This first line shows the name and the version of this file

```
name        = Enter your desired name
description = Enter your desired description
```

The name and description of the theme that you enter here are used on the **admin | build | themes** page.

```
screenshot = screenshot.png
```

The screenshot of the theme that you enter here is used on the **admin | build | themes** page.

```
core       = 6.x
base theme = zen
```

`core` shows the version of Drupal that this theme is compatible with, and `base theme` shows which theme our subtheme uses as a main theme. You don't have to change anything here.

```
stylesheets[all][]   = layout-fixed.css
stylesheets[all][]   = html-elements.css
stylesheets[all][]   = MYTHEME.css
stylesheets[print][] = print.css
```

Here, you can declare the CSS files that are used by our theme. The word inside the brackets shows the media type that uses this specific CSS file.

```
conditional-stylesheets[if IE][all][] = ie.css
```

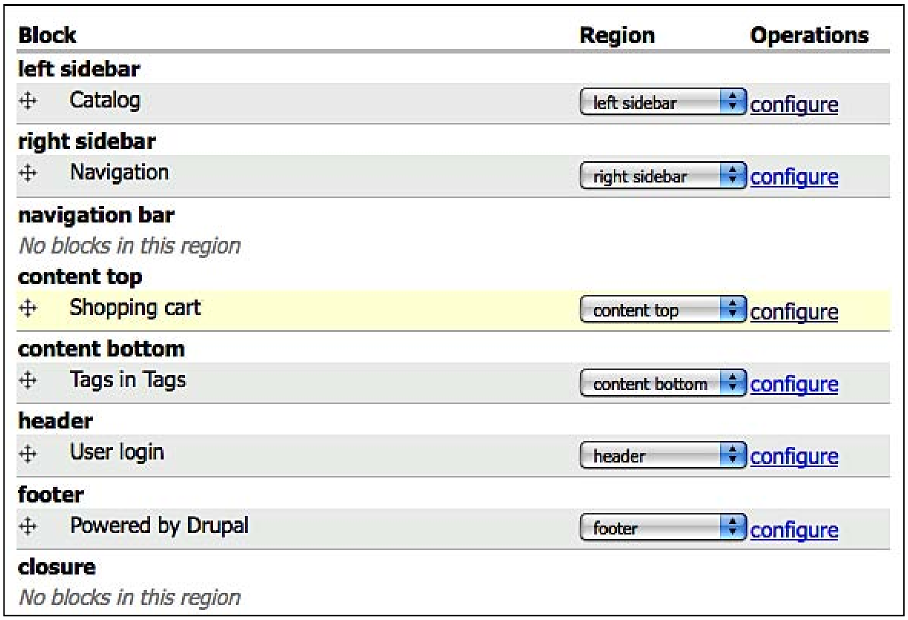
Here, we set the conditional stylesheets that are processed by Internet Explorer.

```
scripts[] = script.js
```

We can also use jQuery scripts in our theme. jQuery is a javaScript library used to enhance user experience. Drupal includes jQuery version 1.2.6 in its core.

```
regions[left]           = left sidebar
regions[right]          = right sidebar
regions[navbar]         = navigation bar
regions[content_top]    = content top
regions[content_bottom] = content bottom
regions[header]         = header
regions[footer]         = footer
regions[closure_region] = closure
```

Zen theme uses eight regions. A **region** is nothing more than a placeholder for one or more blocks. The word inside the brackets is the name of the region in Drupal template files and the word at the right is the description of the region used in **admin | build | blocks** page.




```
features[] = logo
features[] = name
features[] = slogan
features[] = mission
features[] = node_user_picture
features[] = comment_user_picture
features[] = search
features[] = favicon
features[] = primary_links
features[] = secondary_links
```

A Drupal theme contains many extra features, which can be customized from **| admin | build | themes | settings | MYTHEME** page. Here we declare all the features that we are going to use. You can find a list of all the available features in `http://drupal.org/node/171205#features`.

```
settings[zen_block_editing]         = 1
settings[zen_breadcrumb]            = yes
settings[zen_breadcrumb_separator] = ' › '
settings[zen_breadcrumb_home]       = 1
settings[zen_breadcrumb_trailing]  = 1
settings[zen_breadcrumb_title]      = 0
settings[zen_rebuild_registry]      = 1
settings[zen_wireframes]            = 0
```

In this section, we set all the default settings that are going to be used in `theme-settings.php` file, such as block editing or breadcrumb usage. Usually, these settings are just fine and we don't have to change anything.

```
; Information added by drupal.org packaging script on 2009-02-13
version = "6.x-1.0"
core = "6.x"
project = "zen"
datestamp = "1234555897"
```

The information in this section is added automatically when the creator of Zen theme publishes it on `http://www.drupal.org`.

6. Open the `template.php` file and rename all `STARTERKIT` instances to `MYTHEME`. Customizing this file is out of the scope of this book, as it requires the use of PHP programming language. In complex sites, it's impossible to create custom themes without altering this file; however, for our small site, you don't have to make any customizations here.
7. Open `theme-settings.php` file and rename all `STARTERKIT` instances to `MYTHEME`. This is the only change that you have to do in this file, and keeping all the default options is just fine.
8. Rename `zen.css` to `MYTHEME.css`.
9. Write your CSS code. The recommended workflow is to build the basic layout using the `layout-fixed.css` file, customize the basic HTML elements in `html-elements.css`, work in `MYTHEME.css`, and finally debug for Internet Explorer in `ie.css`.

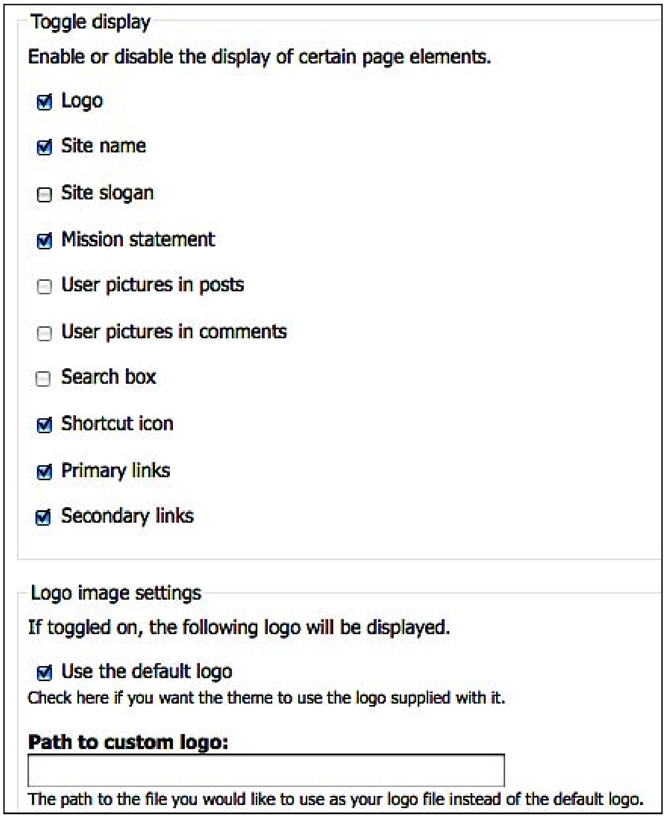
> If you need more details about the aforementioned files, remember that Zen theme offers great documentation, available at `http://drupal.org/node/193318`.

```
  #user-login-form li.user-link, /* The "Cancel OpenID login" links. */
  #user-login li.user-link
  {
    margin-top: 1em;
    list-style-type: disc;
    list-style-position: outside;
  }

  #user-login li.openid-link, /* The OpenID links on the /user form. */
  #user-login li.user-link
  {
    margin-left: -2em; /* Un-do all of the padding on the ul list. */
  }

/** Drupal admin tables **/
  /* We overrode these styles in html-elements.css, but restore them for the
   * forms on the site.
   */
  form tbody
  {
    border-top: 1px solid #ccc;
  }

  form th
  {
    text-align: left;
    padding-right: 1em;
    border-bottom: 3px solid #ccc;
  }

  form tbody th
  {
    border-bottom: 1px solid #ccc;
  }

  form thead th
  {
    text-align: left;
    padding-right: 1em;
    border-bottom: 3px solid #ccc;
  }
```

10. Finally, go to **admin | build | themes | settings | MYTHEME**, and configure the theme from the administration section of Drupal. There are display settings, logo image settings, and shortcut icon settings.

    In the display settings section, there are several elements (declared in `MYTHEME.info`), which you can enable or disable, depending on your needs. There are elements such as logo, site name, site slogan, mission statement, user pictures in posts, user pictures in comments, search box, shortcut icon, primary links, and finally, secondary links.

    In the logo image settings and shortcut icon settings sections, we can use the default logo or icon, give a path to a custom one, or upload a new one directly to the server.



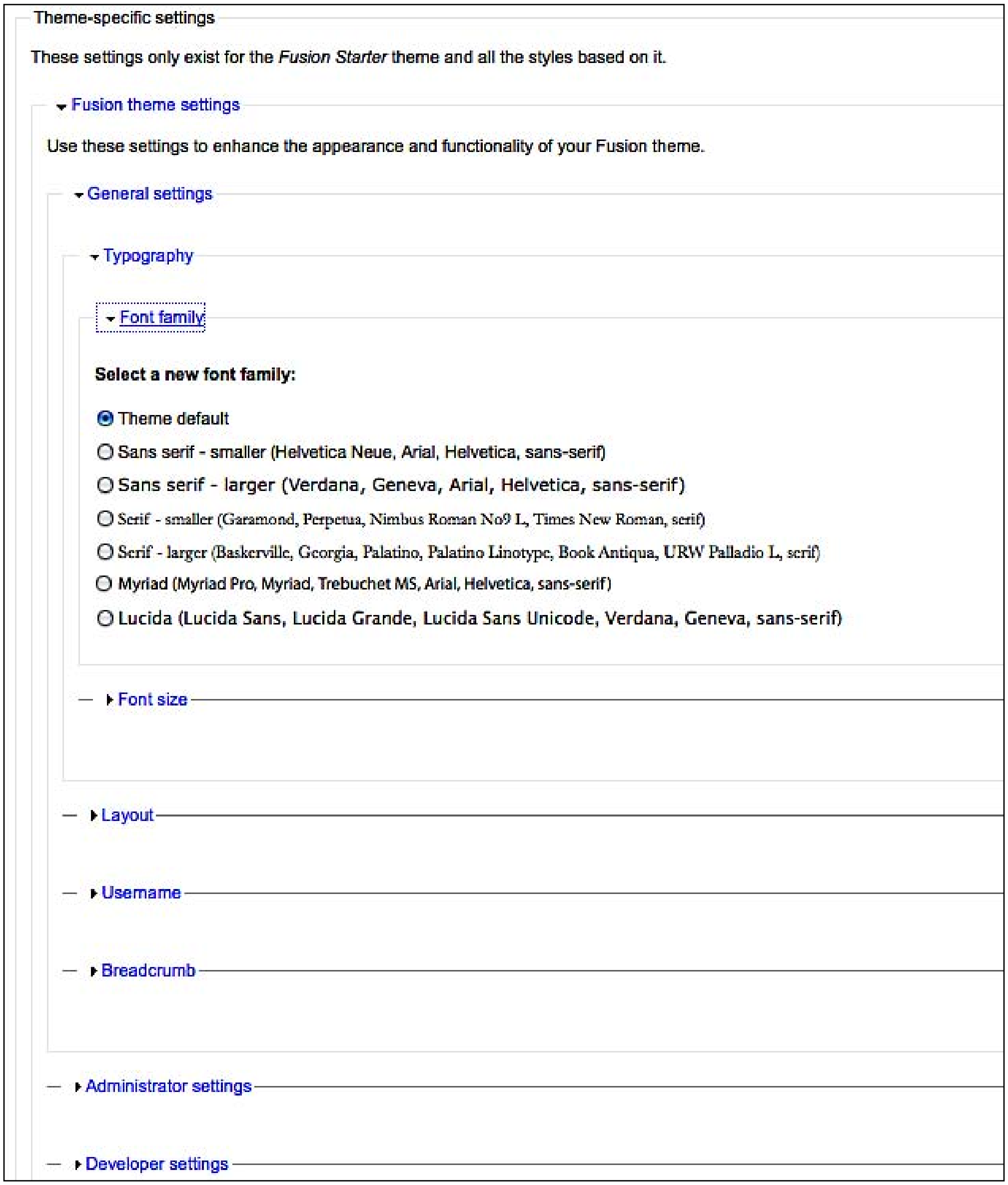
> There is a very interesting video tutorial about Drupal theming. If you have an hour available, watch it at `http://gotdrupal.com/videos/drupal-theme-makeover`.

Fusion Theming System with Skinr module: A new approach to Drupal theming

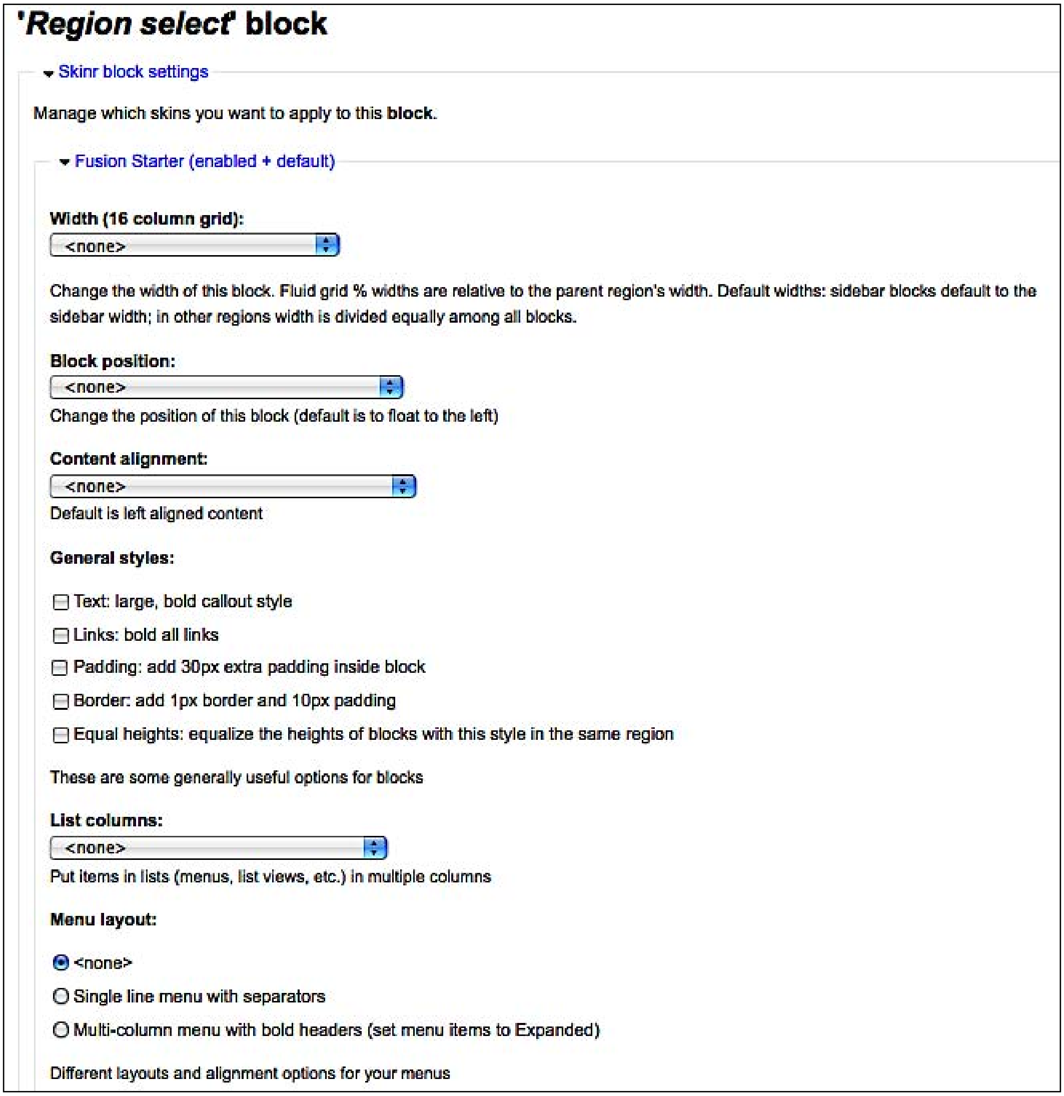
At the end of this chapter, we're going to give you a brief reference to the Fusion Theming System. It was introduced only a few months ago and it's still under heavy development. It's a base theme, meaning that you can create your own subthemes easily, using the Fusion Starter, a commented starter theme created especially for this reason. It uses a 960px or fluid 16-column grid, and its main advantage is that, with the help of Skinr module, it creates layout and style configuration options that the site administrator can control using the website's User Interface, without messing with CSS. So, let's see how to install it, and how to use it for simple customizations.

1. First navigate to `http://drupal.org/project/skinr`, and right after you download the module, upload and unzip to your site folder (`/sites/all/modules`). Then, activate the module from **Administration | Site building | Modules**.
2. Navigate to `http://drupal.org/project/fusion`, and right after you download the theme, upload it and unzip it to your site folder (`/sites/all/themes`). Then, go to **Administration | Site building | Themes**, enable both Fusion Core and Fusion Starter themes and set the Fusion Starter theme as the default one.

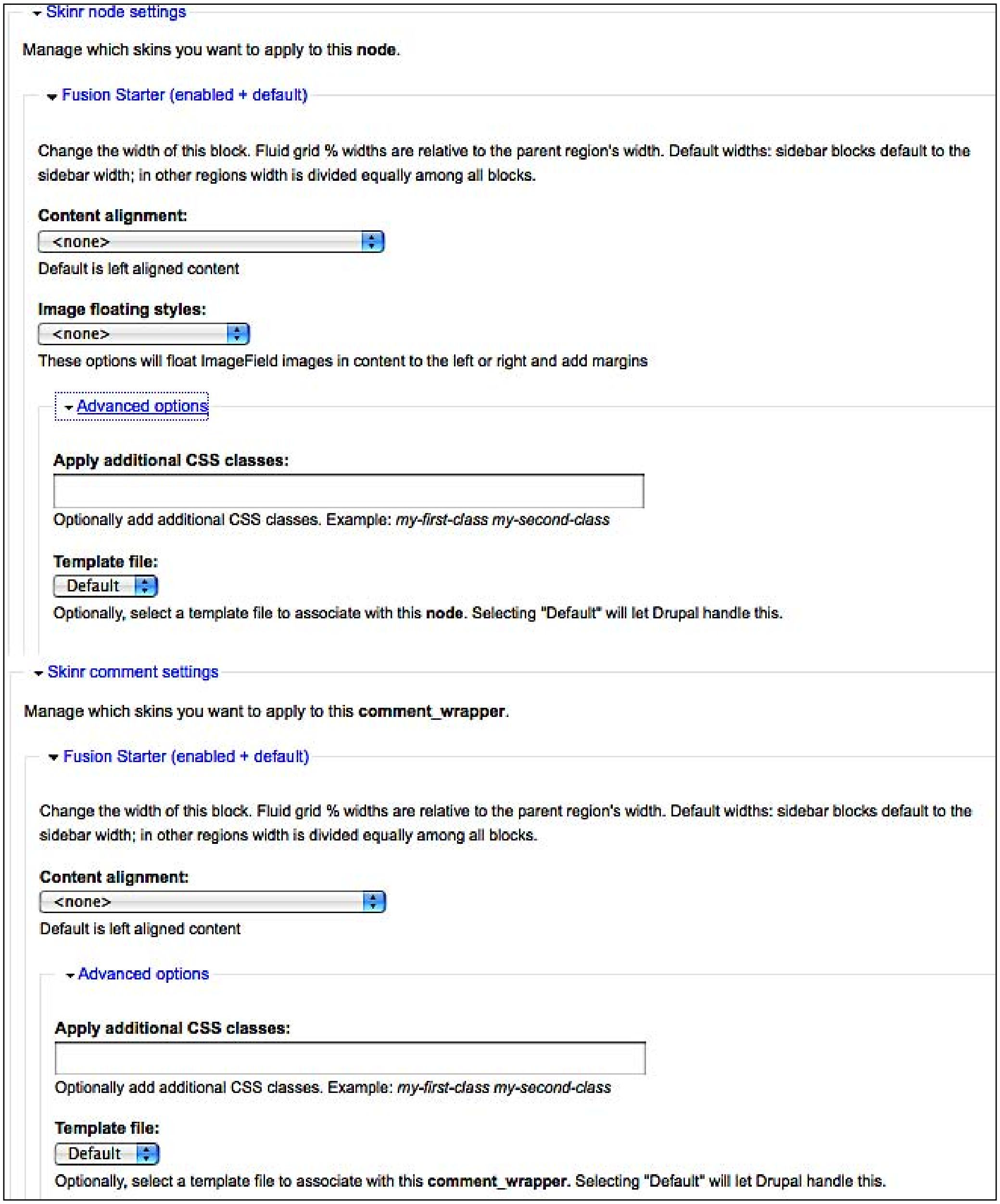
3. Browse to **admin | build | themes | settings | fusion_starter** to configure the settings of Fusion Starter theme. There you will find the default settings of every Drupal theme, such as **logo image settings** or **shortcut icon settings**. However, there is also a new section, named **Fusion theme settings**. There, you can easily change the basic styles and the layout of your theme, such as font family, font size, fixed or fluid layout without using any CSS at all. Click on **Save configuration** to store your settings.



4. Now, if you hover the cursor over any block of your site, you will see a new icon. Clicking on it allows you to configure the properties of this block. You can change the width of the block, the block position, the content alignment, and apply custom styles to the elements of the block, such as padding, border, equal heights, or multi-column menus.



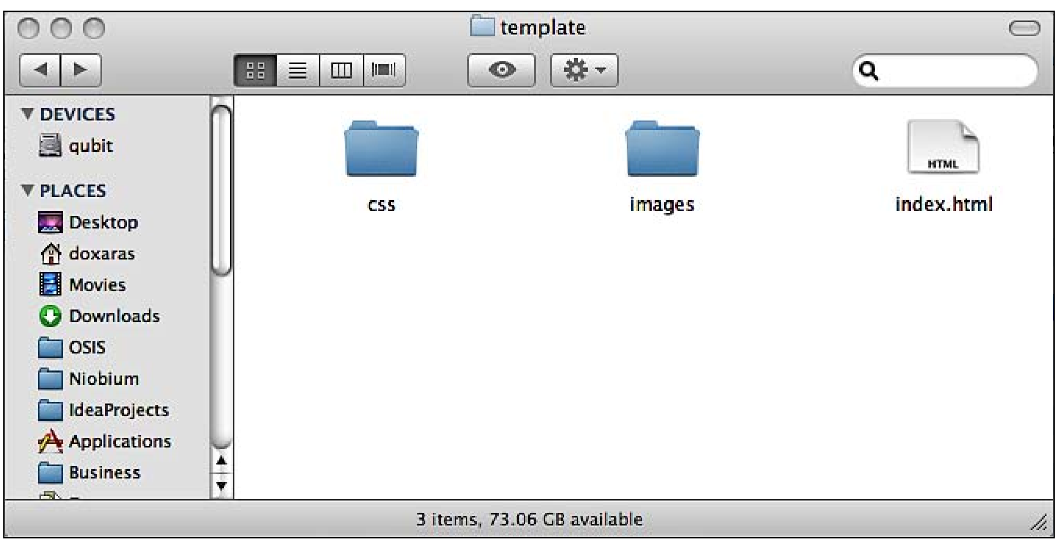
5. There are also special settings for every content type. For example, if you go to **Administer | Content Management | Content Types | Product**, you will see two new sections, named **Skinr node settings** and **Skinr comment settings**, where you can apply custom styles to the product page and the product comments.



> If you want to create your own custom styles for your theme, and present them in the User Interface, you have to study the documentation of the Skinr module, available at `http://www.drupal.org/node/578574`.

Migrating an HTML theme to a Drupal template

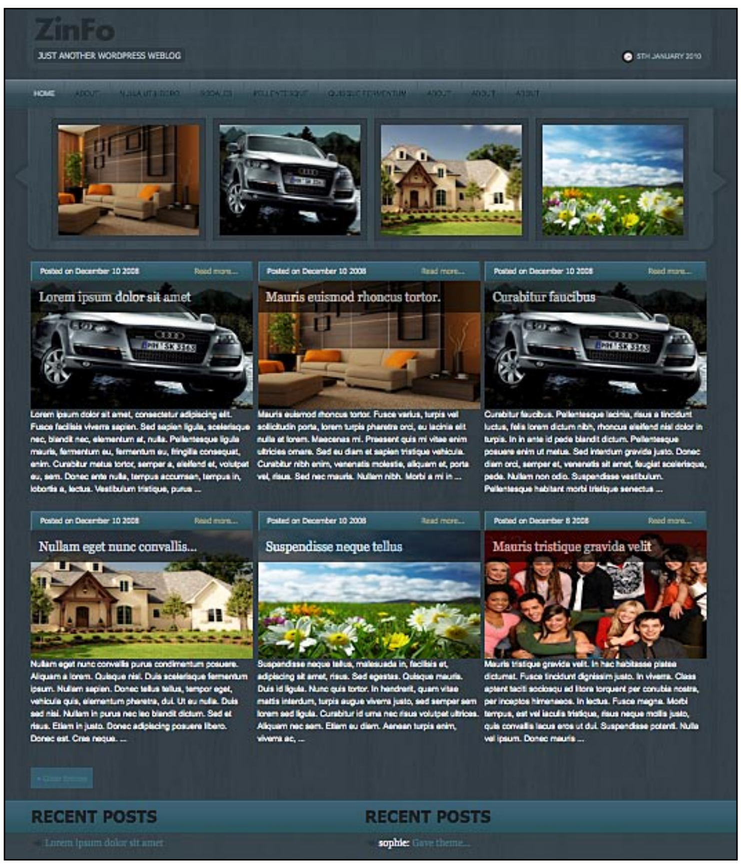
After all these previous discussions, we are now ready to address some more sophisticated XHTML and CSS issues. In this section, we will tackle the problem of converting an existing XHTML CSS template that you may have purchased or created in the past in a Drupal theme. The good news is that what we will describe here is a process of just a few minutes, and it will make your site up and ready for a mockup demo and put it at an 80%-completed level. The bad news is that the remaining 20% may take days to accomplish, as you would probably need to theme your views and specific content areas. However, it is always better to design visualizing the final outcome in front of you, and having some clear indications on what the desirable outcome should look like. We are making the assumption that the HTML template you are starting your conversion with is in the file structure depicted in the following screenshot:



This is the file structure most commonly used by the designers out there. Each HTML template that you will be provided has three basic elements:

- `css`: The folder or just a single CSS file that contains the visual styles and layout.
- `images`: This folder contains all the images that are used either from the CSS or the template by itself.
- `index.html`: This file includes all the elements and the hierarchy of our HTML document model.

Many of the free commercial themes out there, often, also provide Photoshop files that correspond to logos or sophisticated typesetting. We will not need any extra files, so we might as well put them aside. For our case study, we have decided to get a free theme from `http://wpjunction.com/regresso/`, and download it locally, using a brilliant plugin called SaveComplete (`https://addons.mozilla.org/en-US/firefox/addon/4723`), which manages to capture full CSS (even the `@imports`) and JavaScript files.



We have chosen this particular theme, because it provides a nice layout for product promotions on the first page.

The Drupal template engine follows strict coding and naming standards, which makes our life easier, as there's only one good practice to follow in order to create our Drupal theme from our plain HTML/CSS files. To avoid confusion, we provide the following conversion steps. Some of them overlap with actions that we have discussed in previous sections, but it would be useful to put them all in one place here:

1. Move your entire template folder to `[youreshoppath]/sites/all/themes`. For instance if your theme is called `packtpub-eshop`, create the folder `[youreshoppath]/sites/all/themes/packtpub-eshop` and place all your files in there.
2. Rename your main `index.html` file to `page.tpl.php`.
3. Create a file called `[yourthemename].info` (in our case, `packtpub-eshop.info`), with the following elements. For additional information on their significance, you can refer to the *Installing Zen theme* section of this chapter.

   ```
   name = packtpub-eshop
   description = niobium basic theme.
   regions[header] = Header
   regions[left] = Left sidebar
   regions[right] = Right sidebar
   regions[center] = Center Sidebar
   regions[content] = Content
   regions[footer] = Footer
   regions[toplinks] = Toplinks
   regions[topright] = Top Right
   version = VERSION
   core = 6.x
   engine = phptemplate
   ```

   If you create the above file, navigating to your Drupal theme section, you will be able to see your new theme displaying as an option. Do not hurry to select it, as we have to complete the entire process before the changes can make visual effects.
4. Then, because the Drupal's default template engine can automatically recognize and inject a file called `style.css` in the template (we will see how in a moment), just rename your CSS file to `style.css`, and place it in the same level with your `page.tpl.php` file.

5. Open `style.css` and make a quick find-replace changing the image paths to match your new image folder location. If you have preserved the same structure as the initial configuration, you do not need to do anything because URL paths inside CSS are relative to the CSS file location.
6. Finally, a group of some small changes is the injection of the PHP variables of the template engine inside your `page.tpl.php` file. There are a number of content and context definition variables, each of which have specific meaning and can be used in certain locations. Furthermore, we will elaborate on the properties made available by the template engine that we will be using in our theme conversion. For the complete list of Drupal's template engine, you can refer to the online documentation at `http://api.drupal.org/api/drupal/modules--system--page.tpl.php`.

   General utility variables

   - `$base_path`: This is the URL that corresponds to the path of the Drupal installation. If Drupal is in the root folder, your domain will be `/`; otherwise, something like `/mydrupal/installation/`. Note that if you have changed the path of your Drupal installation, you have to run `/update.php` to update the engine with the new path.
   - `$directory`: This variable provides the current theme directory, for instance `themes/garland`. It's useful for defining paths for theme-specific JavaScripts and images.
   - `$is_front`: This is a Boolean variable declaring whether or not the current page is the front page. It's useful for defining front page-specific elements in an `if` statement while theming.
   - `$logged_in`: This again is a Boolean variable, defining if the current user is registered and signed in.
   - `$is_admin`: Another Boolean variable, defining if the user has permission to access administration pages.

   Page metadata

   - `$head_title`: This represents the page title that includes the site name with the specific content page title. In *Chapter 10*, we will address the procedure of turning this variable into SEO-friendly notion.
   - `$styles`: Style tags are necessary to import all CSS files for the page. Style tags including necessary CSS files for the current theme can be configured through the theme info file we discussed previously.

- `$scripts`: Script tags are necessary to load the JavaScript files and settings for the page. If you have theme-specific JavaScript files, you need to add them manually using the following theme path conversion:

```
<script type="text/javascript"
        src="<?php print $base_path . $directory; ?>
    /scripts/script.js">
</script>
```

Site identity

- `$front_page`: This variable defines the URL of the front page. This variable should be preferred over `$base_path`, as it includes language-specific prefixes.
- `$logo`: This variable defines the path to the logo image, as defined in theme configuration, namely the `theme.info` file.

Navigation

- `$search_box`: This variable declares HTML formatting displaying a site-wide search box. Note that this variable is empty when search is disabled.
- `$primary_links`: This variable defines a themed array containing primary navigation links for the site.
- `$secondary_links`: This variable defines a themed array containing secondary navigation links for the site. This will be empty if no secondary links have been configured.

Page content (in order of occurrence in the default `page.tpl.php`)

- `$left`: This variable is used to hold the formatted HTML code for the left sidebar, usually the left blocks section. The left section, like every Drupal theme section, should be configured in the `theme.info` file.
- `$breadcrumb`: This variable is used for the breadcrumb trail for the current page.
- `$content`: This variable defines the main content of the current Drupal page. Either a view or a content page or a CCK page.
- `$right`: This variable is the same as `$left`, except it is used to hold HTML for the right sidebar block area.

All of these variables are included in the following code snippets, which you can, by all means, reuse, as they will do the trick in most cases:

- **The header substitution**: Replace your custom HTML header with the code provided below. This code contains all Drupal-specific theming elements as well as all module-dependent stylesheets. If you need to add special JavaScript files for theme-specific functionality, add them below the `$scripts` variable:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xml:lang="<?php print $language->language ?>"
          lang="<?php print $language->language ?>"
          dir="<?php print $language->dir ?>">
  <head>
    <?php print $head ?>
    <title><?php print $head_title ?></title>
    <?php print $styles ?>
    <?php print $scripts ?>
    <!--[if lt IE 7]>
      <?php print phptemplate_get_ie_styles(); ?>
    <![endif]-->
    <meta http-equiv="Content-Language" content="en-us" />
    <meta name="robots" content="all" />
    <meta name="author" content="<?php print $site_name ?>" />
    <meta name="keywords"
          content="packtpub, online, eshop, electronics" />
    <meta name="description"
          content="<?php print $site_name ?> |
                   <?php print $site_slogan ?>" />
```

Having done that, you can now select the `packtpub-eshop` theme as your e-shop's default theme. You now have a styled page that does not have any content. In other words, your page now resembles the original theme you extracted.

Now comes the time to start theming properly and code the layout policy you will follow. Take into account that you can freely get code snippets and ideas from existing themes—both the core ones, such as Garland and Zen, and every other theme that you may find suits your needs from `drupal.org`. The idea from now on is to start replacing static content with the Drupal variables we have mentioned before, which correspond to HTML-formatted content generated from Drupal's theme engine. We will keep it very simple, so, in addition, we will provide some code snippets for the block layout and the main content of your theme.

- **Links theming**: This is one of the most importing and demanding task, especially for online stores with complex product catalogs. Easy navigation is the key to successful sales and conversion rates. In addition to that, SEO enabling demands a good tree structure, so you should make good designing decisions on the link arrangement you need and which of them should be HTML or JavaScript links.

```
<?php if (isset($primary_links)) : ?>
    <?php print theme('links', $primary_links,
        array('class' => 'links primary-links')) ?>
    <?php endif; ?>
    <?php if (isset($secondary_links)) : ?>
    <?php print theme('links', $secondary_links,
          array('class' => 'links secondary-links')) ?>
    <?php endif; ?>
```

- **Content theming**: In this section, you are generating the main content area of your site. Content can be a view or a CCK content instance, and can be themed further in other Drupal template engine files such as `page.tpl.php`, that are out of this book's scope. For further information, you can refer to the online documentation at `http://drupal.org/theme-guide`.

```
<?php if ($title):
    print '<h2'. ($tabs ? ' class="with-tabs"' : '') .'>'
    . $title .'</h2>'; endif; ?>
<?php if ($tabs):
    print '<ul class="tabs primary">'
    . $tabs .'</ul></div>'; endif; ?>
<div class="clear-block">
            <?php print $content ?>
</div>
```

- **Sidebar left theming**: The left sidebar is the left section of your block placement area. All blocks configured in the left area will be displayed here. Note that you should have configured the left section in your `theme.info` file.

```
<?php if ($left): ?>
  <div id="sidebar-left" class="sidebar">
    <?php print $left ?>
  </div>
<?php endif; ?>
```

- **Sidebar right theming**: This is the same as above, but for the right section.

```
<?php if ($right): ?>
  <div id="sidebar-right" class="sidebar">
    <?php print $right?>
  </div>
<?php endif; ?>
```

- **Footer theming**: The footer theming section just prints the footer HTML content, which is administrated from your Site Information page.

  ```
  <div id="footer">
    <?php print $footer_message ?>
    <?php print $footer ?>
  </div>
  ```

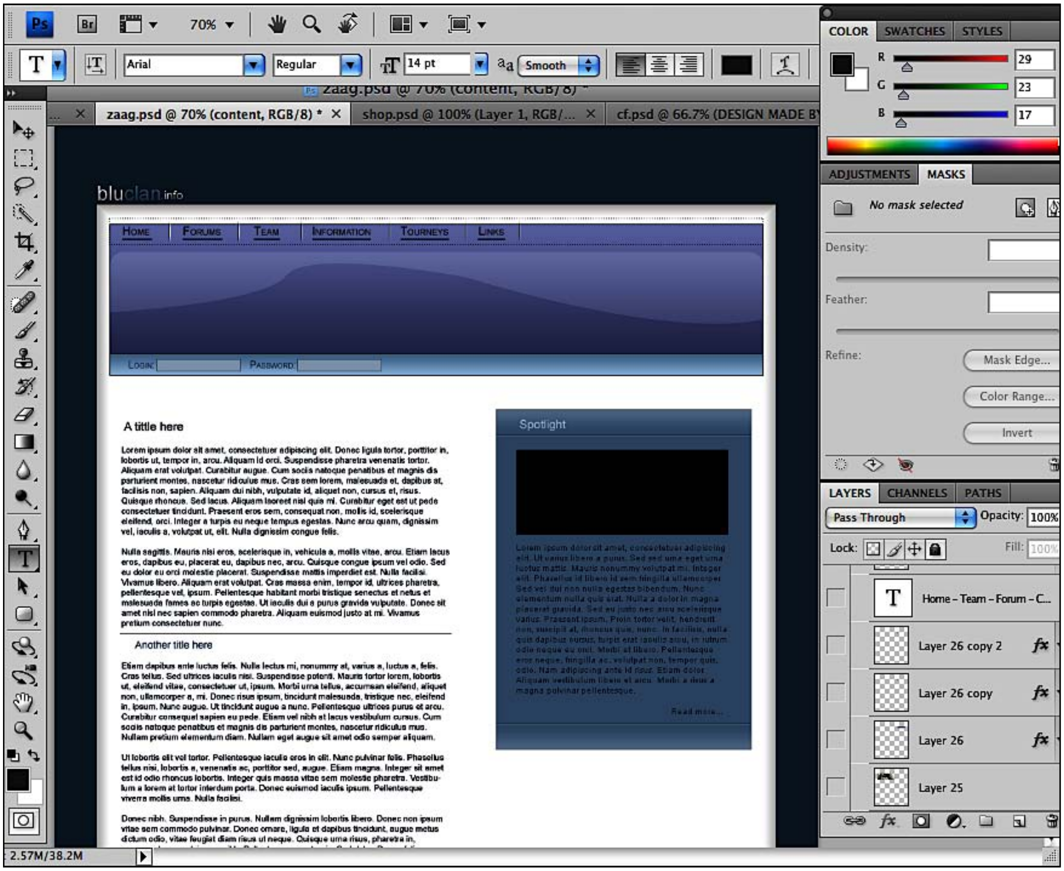
Creating a Drupal theme from a Photoshop template

Photoshop website design is a very common starting ground in modern web application design. People often tend to design their layout in Photoshop by adding image mirroring, shadows, transparencies, and other filters, and also visual enhancements in addition to defining the basic layout patterns.

In this section, we will cover the procedure of replicating (I prefer the term *replication* over *cutting*, you will see why in a while) your PSD Photoshop theme to XHTML and CSS. If you are the one who creates the designs, just as a general advice, try not to completely mock up all the elements in too much detail. This is because, in HTML, there is always the danger of not being able to replicate the proposed look and feel.

Our philosophy is to approach the PSD not just as an image that should be cut and layered on our page using HTML, but rather as a pattern that indicates how our page should look in the end. Therefore, we propose to start by trying to replicate all of the design elements using XHTML and as much CSS as possible, and refer to image cutting and exporting only when you see an effect that you either cannot replicate with CSS or would take an exceptionally long time to do it. Many web designers tend to cut large images into little pieces or use automated software components to achieve the same look and feel. However, in the end, this approach will result in cluttered HTML, and remove all the control that a good CSS layout adds to your design. Moreover, you do not want to rely on the images too much for your theming design, because there are hundreds of reasons that images might not be fetched, and you definitely do not want network problems to interfere with your basic template design.

So to start with, let's introduce our sample PSD that is available for downloading at the code section, and tries to simulate a common theme with areas of focus.



We will not go through the entire process, as such an activity presumes in-depth knowledge of CSS and XHML. However, for all those competent in these technologies, we will provide a straightforward approach that will help produce the most professional results with less effort.

> If you want to find out more about theming in Drupal and especially dive into the magnificent world of CSS, a good place to start is *Drupal 6 Themes* by *Ric Shreves*.

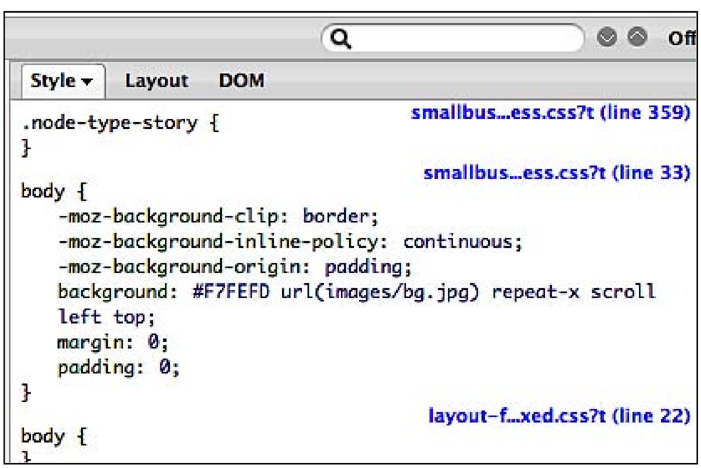
The key concept is only to cut out from Photoshop parts and images that you cannot get along without and leave everything else to handle in CSS magic. So, things such as the following should always be used as is from your PSD:

- Repeating backgrounds and gradients
- Rounded corners and shadows
- Typesetting that cannot be reproduced with standard CSS

All other components such as background colors, text, and decoration elements should be coded into your CSS. Another issue you should keep in mind is to try to write strict XHTML and separate all of your styling rules into your CSS.

After we have provided some general guidelines, we will cut and create two basic elements, the background and the logo, just to give you the idea of PSD to HTML conversion. To migrate your background images from your Photoshop template to your Drupal theme, follow the next steps:

- If our background is a structured image, there are many techniques for automatic window scaling. However, these techniques trim it down to a lower-pixel width or height to achieve smaller image file size. Take into account that if your pattern is vertical and demands vertical replication, you should cut a one-pixel height and full-width background image. In the same away, if you have for instance a horizontal gradient fill for your background, you should cut a one-pixel width image.
- Click on **Export for Web**, and theme your CSS as we display below:

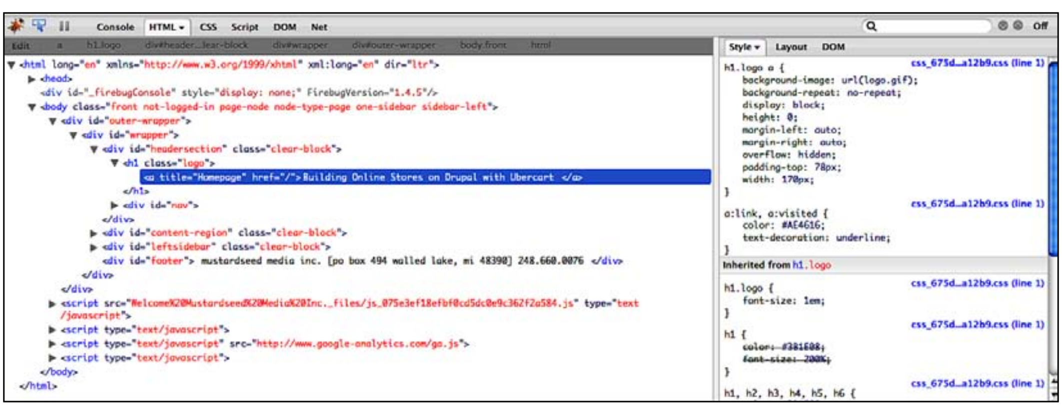


The CSS code is as follows:

```
body {
background-color:#FFFFFF;
background-image:url(body-bg.gif);
background-repeat:repeat-x;
color:#3B1E08;
font-family:arial,helvetica,sans-serif;
font-size:0.9em;
line-height:1.5em;
margin:0;
padding:0;
}
```

Next, we will see how we can export our logo from our PSD template. Keep in mind, that if our logo can be recreated with standard CSS typesetting, we should try to import it in the CSS. However, this is a rather rare case as corporate logos tend to change, and most of the times, you will need to export it in a GIF image. To do so, perform the following actions:

- Group all your logo components to a new layer in Photoshop.
- Trim the layer of all the transparent pixels to get the smallest file size.
- In Export for Web, choose the transparent GIF option. (Note that PNG background transparencies do not show up in Internet Explorer 6.) So if you indeed need to use PNG image formats, you need to include jQuery pngFix JavaScript library, located at `http://jquery.andreaseberhard.de/pngFix/`, or install PNG Fix Drupal module, located at `http://drupal.org/project/pngfix`.
- Theme the image as background to achieve an XHTML layout, and you are done.



You can also see the CSS code for the logo background as follows:

```
h1.logo a {
background-image:url(logo.gif);
background-repeat:no-repeat;
display:block;
height:0;
margin-left:auto;
margin-right:auto;
overflow:hidden;
padding-top:78px;
width:170px;
}
```

So, you are now ready to take it up from here, referring to the PHP variables of the Drupal default template engine we introduced in the previous section. Finally, keep in mind that putting most of your visual layouts and markup in your CSS and keeping your HTML clean and tidy will enable you to perform all kinds of changes in the future and enhance your site template maintainability.

Summary

We went through some of the vast configuration options that the Drupal theme engine provides us with. We have discussed important theme selection tips you should follow when choosing your custom template and configuring it to meet your needs. This process involves a lot of experimenting and page refreshes. Also note that at the time of uploading content, new needs will emerge, needs concerning visual enhancements and product advertising. Decision calls will be made for choosing the most appropriate user interface layouts. Finally, due to the fact that this is a continuous process and will never stop evolving, you need to have in-depth understanding of your theme functionality to better support future calls.

9

User Interface Enhancement Techniques

Right now, you may wonder "My e-shop is working just fine, and I want to start selling my products right now, so why do I have to invest more time on it?". The truth is that you have done an amazing job until now, and your e-shop is already great. But, you have to keep in mind that there are thousands of competitors out there, selling similar or the same products that you're selling and their owners want to earn money as badly as you do. So, being good is not enough, you have to be one of the best.

In this chapter, we're going to show you:

- How to implement more complex marketing techniques, such as cross-selling using taxonomy and recommendation systems
- How to create an alternative layout for products using panels
- How to enhance the user interface using views
- How to improve the shopping cart
- How to manage product prices and discounts in a better way

Product cross-selling

Product cross-selling is a very powerful policy that you might be familiar with. For instance Amazon.com was one of the early adopters of recommendation systems in a very sophisticated manner, and it boosted its online selling rates by hundreds of millions of dollars. If we move on to more algorithmic complexity, a more sophisticated example is Netflix, an online movie rental service, and the core of its business is its recommendation system and the hype that surrounds it. Finally, the most recent and simplest to implement is Last.fm with a very elegant and efficient recommendation algorithm. By adopting Drupal and Ubercart, things become pretty straightforward, as you have good modules that encapsulate the complexity of recommendation algorithms and require little configuration, and you may as well provide to your end customers a great consumer experience. In addition to that, do not forget the powerful and robust taxonomy mechanism that Drupal implements in its core and the site-wide content tagging it provides, so the relevant items and items that could be used in conjunction could be categorized. So now without further delay, we will go through all these interesting possibilities that our Drupal online shop could offer.

Using taxonomies

As we have already mentioned, taxonomies are the core of the Drupal system and grasping the high-level implementation can save us a lot of trouble most of the time. Taxonomies often help us create references for our Drupal system nodes, differentiate between them, and create easy-to-use, intuitive, and searchable views on our content. Therefore, in our example, the basic idea is to create a taxonomy not only for products that can be sold as groups (as we already have Ubercart product kit for that), but rather for the products administrator to be able to tag all these relevant products in a way that high-revenue electronic shops like ExpanSys and PixMania have adopted.

To achieve this we do not need any new module installation but rather the plain old Drupal taxonomy system. We will make two taxonomies: one for product mangers, which they can edit while they add new products, and another in which users can free tag your content. These free taxonomy vocabularies are also referred as folksonomies. Furthermore, everyday practice has shown that relevant taxonomy blocks can really boost your site traffic, page views, and eventually conversions that translate to purchases. The vocabularies that we will alter are the following:

- **Community Tagging**. We need this particular free tagging vocabulary to allow our end users to tag the products of our site in order to provide non-intuitive connections.
- **Product Types**. This stands for an internal tagging vocabulary with predefined terms namely **offer**, **best price**, and **new product** and will help create the corresponding views in order to perform product promotion in your online electronics shop.

To create the new vocabularies and change the existing one, we take the following steps:

1. Navigate to **Administer** | **Content management** | **Taxonomy**. Click on **Add vocabulary**.
2. Fill in the name as "Community Tagging" and provide a short description.
3. Choose **Product** and **Product Kit** as the **Content types** that will be associated for tagging.
4. In the **Settings** pane choose **Tags** to allow free tagging and **Multiple select**.
5. Finally choose **Not in Sitemap** for **XML Sitemap**.
6. To add another taxonomy for product visibility options and positioning, go back to the taxonomy page and again click on **Add vocabulary**.
7. Add the name "Product Type" along with a short description and click on **Product** and **Product Kit** in the **Content types** section. Finally add a priority **1.0** to the **XML sitemap element** and click on the **Save** button.
8. Navigate to your newly created vocabulary terms and add the terms "offer", "best price", and "new product".

This is an example of a user-defined term-tagging procedure on one of our products.

Use Taxonomies for Navigation and Menus

You can also use Drupal's system pages using the taxonomy view module for category listings. The end of the URL should look like this: `taxonomy/term/1` or `taxonomy/term/2`.

Note that taxonomy URLs always contain one or more Term IDs at the end of the URL. These numbers, 1 and 2 above, tell the Drupal engine which categories to display. Now combine the Term IDs above in one URL using a comma as a delimiter: `taxonomy/term/1, 2`. The resulting listing represents the boolean AND operation. It includes all nodes tagged with both terms. To get a listing of nodes using either taxonomy term 1 OR 2, use a plus sign as the operator: `taxonomy/term/1+2`

Using recommendation systems

Recommendation systems have existed a long time and make a crucial contribution in some of the most successful online shops. In this section we will focus on examples of implicit data collection of the customer's activities that include the following:

- Observing the items that a user views in an online store.
- Analyzing item/user viewing time.
- Keeping a record of the items that a user purchases online.
- Obtaining a list of items that a user has listened to or watched on his or her computer.
- Analyzing the user's social network and discovering similar likes and dislikes.

Having these data and customer behavior in our account, it is then easy to find the optimal item suggestions that fit people's profiles. We can then provide sections like "customers who bought this book also bought" on Amazon.com suggestions.

Further to our discussion we will install recommendation API and two other modules that depend on it. The Ubercart-oriented module is the Ubercart Recommender module. This module collects data through the Drupal Core Statistics module about user purchases and provides suggestions about other products that could be relevant to the returning customer. All recommendation systems assign special weights in their recommendation algorithm to purchased products since this generates returned value and we have a fully converted customer. In order to handle suggestions to users that have not made any purchases yet from our online shop we will also use the Browsing History Recommender and Relevant Content modules. You can find more information about the algorithms and the recommendation procedure implemented in the Drupal Recommender API at `http://mrzhou.cms.si.umich.edu/recommender`.

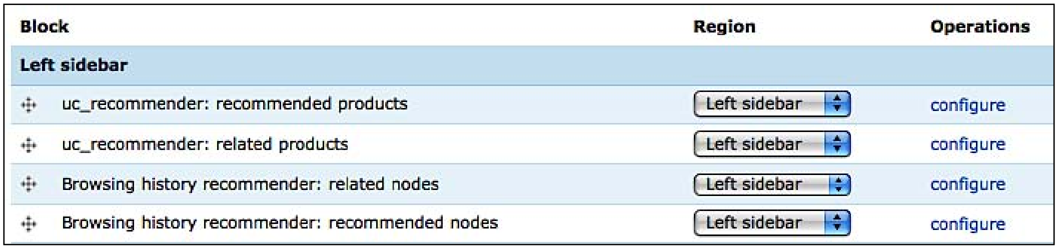
Next we provide a synopsis of the added value and the functionality of each module:

- **Browsing History Recommender**: This module adds two blocks in your site "Users who browsed this node also browsed" and "Recommended for you". To calculate the recommendations, this module uses Drupal statistics and in particular the history data and keeps track of 30 days of users' node browsing activity. The "Recommended for you" block provides personalized node recommendations based on a user's node browsing history.
- **Relevant Content**: This module provides two ways of referencing content relevant to the node in sight. Both of these methods provide configuration to filter for specific content types and vocabularies, limit the maximum size of the result, and provide some header text. The result in both cases is a list of nodes that the module considers most relevant, based on the categorization of the current page. You can configure multiple blocks with individual settings for node type, vocabulary, maximum result size, and optional header text.
- **Ubercart Products Recommender**: This module actually adds two extra block in our blocks section, one named "Customers who ordered this product also ordered", which performs a cross check between orders of customers that bought the particular product in sight and another named "Recommended for you", which provides personalized products recommendations based on a user's purchasing history.

To configure your online shop to provide content-related recommendations we need to perform the following administration steps:

1. Download Recommender from Drupal.org by navigating here:
   `http://mrzhou.cms.si.umich.edu/recommender`.
2. Unzip the file in your site's `modules` directory.
3. Navigate to the modules administration screen and activate this module.
4. Follow the preceding procedure for the following modules also: `http://drupal.org/project/relevant_content`, `http://drupal.org/project/history_rec`, and `http://drupal.org/project/uc_rec`.

5. After you have uploaded and installed all modules you will be able to see the following blocks in your blocks page (**Administer** | **Site building** | **Blocks)**. Although these modules have configuration screens, they do need extra configuration or property change actions and can be assigned to your theme regions as they are from the blocks section.



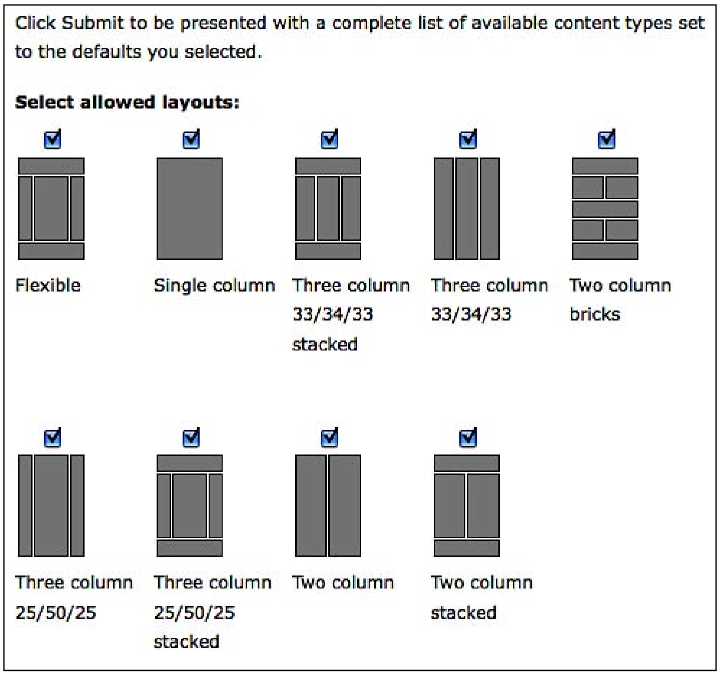
> You can find a very thorough discussion about all recommendation modules on Drupal at `http://groups.drupal.org/node/12347`.

Product layout using Panels

The **Panels** module is one of the most controversial Drupal modules. The main reason for this is the long period this module, for Drupal version 6, has spent in beta version. Only recently has a stable version 3.2 for Drupal 6 been launched. In addition there was a dead end in the Panels version 2 development branch that forced the module developers to skip the version and jump to 3. Panels module has gained a new momentum with new and rich page customization features that address both experienced and everyday users that want to configure their home page or their inner pages as well. Experienced themers that were reluctant to use the Panels module as they were used to performing most of the layout actions using XHTML and CSS (we discussed themes in *Chapter 8, Customizing the Frontend*) are becoming more open to adopt Panels, as it provides reusable page layouts along with the option for end users to change their site look and feel in just a few seconds. The real power of Panels emerges, of course, in conjunction with the Views module generating a fully customizable block view that can be placed in every single area of your template.

Panels in Drupal

The Panels module allows a site administrator to create customized layouts for multiple uses. In our case you may be thinking of changing the home page layout or your electronics search page. You can use the drag-and-drop content manager functionality that lets you visually design a layout and place content within that layout. Integration with other systems allows you to create nodes that use this, landing pages that use this, and even override system pages such as taxonomy and the node page so that you can customize the layout of your site with very fine-grained permissions. Panels comes with a number of preconfigured page layouts that you can choose, in order to fully customize each page according to your needs. You can see all possible page layout configurations available in your Drupal site in the following image. This image shows the Panel administration screen at **Administer | Site building | Panels | Settings | Panel Nodes**.



Panels makes a wide use of contexts in Drupal. Context in plain words is a mechanism Drupal uses for the runtime engine to decide on some global variables about the page being rendered at the present time. Therefore, Panels can change its layouts according to aspects of the page rendered such as the time you see the page, the role of the users that makes the request, and the preferred language of the user. These contexts can be checked for information and used not only to make content available to be displayed, but to choose which layout to display. For example, if your site is international, you can use context to see if the node being viewed is set for a particular language and choose to display it one way if it is in French or another way if it is in English. You can also select on attributes such as node type, whether or not the user has access to edit the node, and more.

Panels also includes simpler applications of the drag-and-drop system. There is a node type (the 'panel' node) that can simply be added as content to your system. By being a node it loses a lot of the features that the more powerful page system has, but it does have the advantage of simplicity, and gaining all of the functionality that nodes normally get.

Panels can also be used for items smaller than pages. What if you have a normal sidebar, and you have two pieces of content that waste a bunch of space because they're really too narrow, and your design looks significantly better if they are side by side? That's complex to do in Drupal because what you end up having to do is create a custom block with custom code to display these two pieces of content. With Panels, you just create a "mini-panel" with a two-column layout. Add one block to the left, one block to the right, and finish. This mini-panel will then be available to your system as an ordinary block, or as panel content to go in other panels.

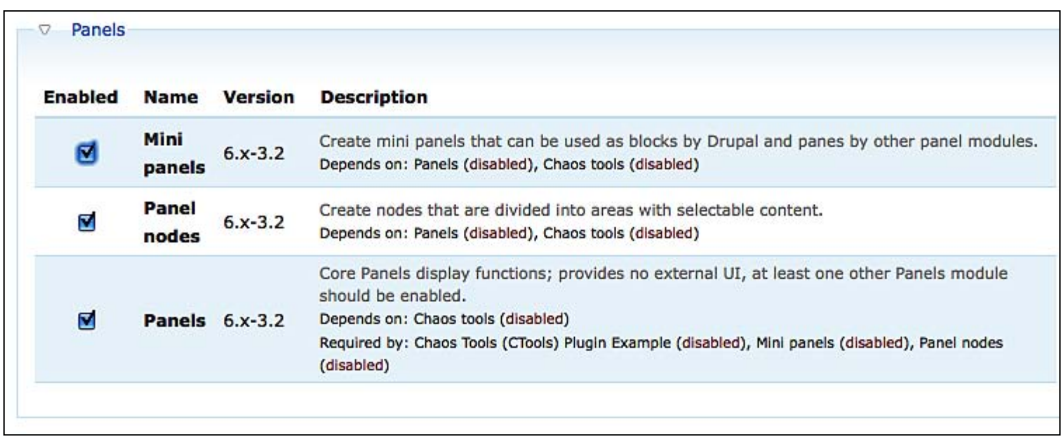
The Panels module integrates with Views to allow administrators to add any view as content. Or, for uses where the layout editor needs more tightly controlled content, Views can be given custom displays to provide only what the site administrator wants the panels builder to use.

- **Panel pages** are the primary Panels module; you can use this for creating single full page layouts. This replaces the standard panel that existed in the earlier versions of Panels. If you are upgrading your site from Panels 1, and you cannot find where your panels went, be sure to enable the panel pages module.
- **Panel nodes** are useful for creating layouts that only occupy the content area of your pages. Frequently, it is desirable to add an area to a node layout, such as a pull quote for a newspaper or a photo block, which you don't necessarily want on every node. Panels nodes let you control the layout of a single node at a time and place content such as blog posts, images, and blogs in and around the post.

- **Mini panels** are a layout mechanism for blocks. It won't take long using panels before you get to a point when you want a panel inside of a panel. Or a panel that can be used as a block. That is exactly what mini-panels do. You can create a small panel here with various pieces of content and then put it inside of a panel page or panel node.

In order to install the Panels module you need to download it from `http://drupal.org/project/panels` and unzip it in your site's `modules` directory. Take into account that the Panels module has a prerequisite of the `ctools` module so if you have not installed this you need to do it right now. You can find `ctools` at `http://drupal.org/project/ctools`. You also need to unzip `ctools` in your `modules` directory and then navigate to the modules page from your Drupal administration screen, **Home** | **Administer** | **Site building** | **Modules**. At this screen make sure you activate the following modules because to have a fully functional panels installation you need the page manager chaos tools counterpart. This is actually the old delegation module that was renamed and added to chaos tools in Drupal 6. If everything works fine you will be able to see the following modules:

- Chaos tools
- Page manager
- Views content panes
- Panels
- Panel nodes
- Mini panels



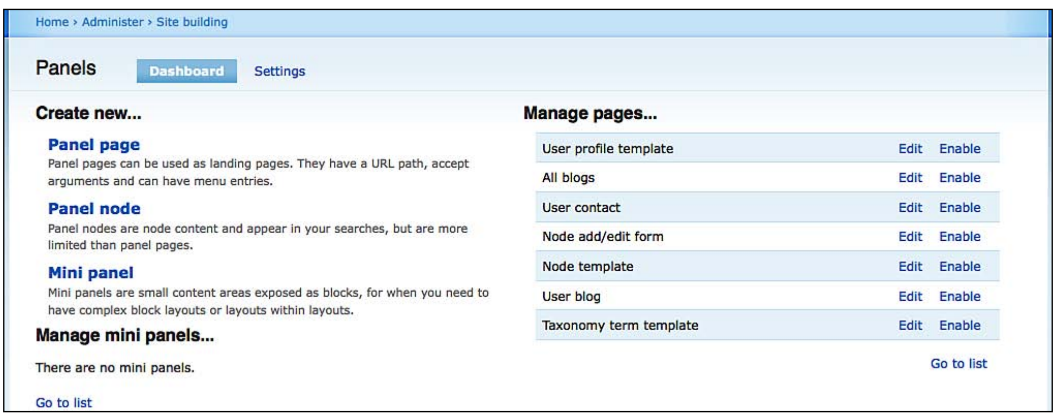
You probably need all of them so check them and click on **Save configuration**. There you are, with the Panel module installed and ready to start the action.

Panels used in theming

Creating a custom layout in your theme can also be accomplished using panels and to find out more navigate to `http://drupal.org/node/495654`. A panel layout is a layout that can be used by the Panels module. You can add content to defined content areas within a panel. The Panels module comes with several layouts, but you can also create your own for your own module or theme. Here we will create a layout for a theme.

Panels basic configuration

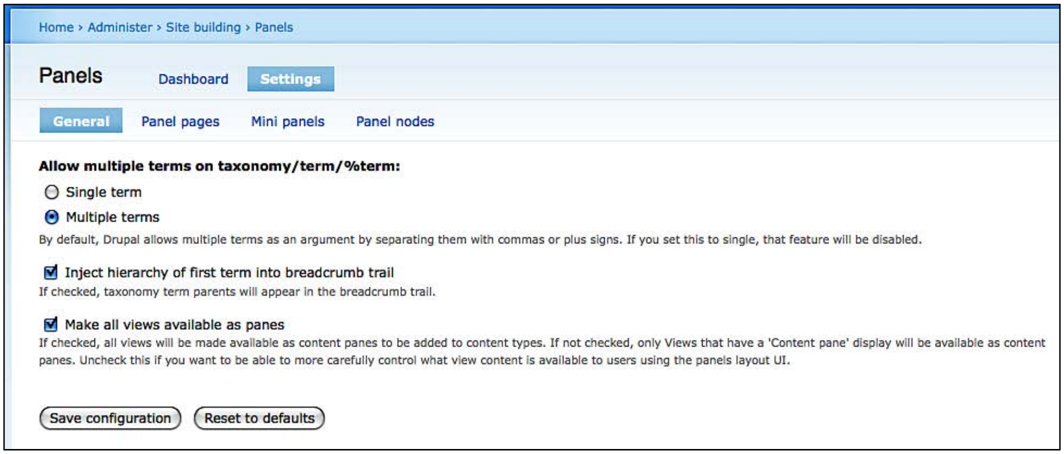
Before we start developing our first page we will take some time and review all the configuration options of panel pages. Luckily, as the developer of Panels is also developing Views we will see some resemblances and same clean and intuitive design as Views. Navigate to **Home** | **Administer** | **Site building** | **Panels** where you will see the following screen:



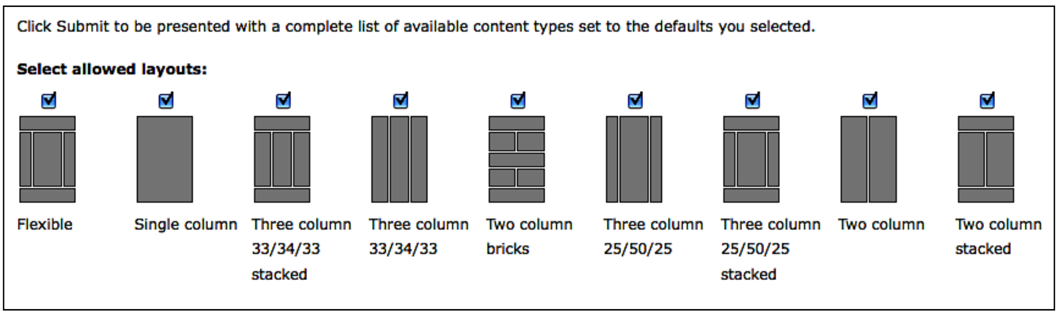
Actually, Panels relies upon third-party modules for page administration, hence it does not provide many options in its main administration screen. You will see all three possible panel types that you can make, namely, **Panel page**, **Panel node**, and **Mini panel**. **Panel node** behaves like a plain old Drupal node but with specific layout and mini-panels, which can be assigned in blocks. By clicking on these options you will be able to navigate to other sections such as **Pages** from the page manager module. We take the opportunity to point out that **Panels** is literally a huge module that provides a great number of configuration options to the end user. Therefore, in this book our main intention is not to provide great insight into Panels operations but rather to show how Panels can be used efficiently to interact with your customers and arrange your content properly. We will not go into detail on mini-panels and other special features like context role panel layout.

Furthermore, in the basic administration configuration you can change the following features and functionality:

- In **Home** | **Administer** | **Site building** | **Panels** | **Settings** | **General settings** you can change the following:
    - The first option is Term hierarchy URL parsing options. Panel pages can take terms as arguments in their URL and then pass them to each view that is configured in each panel area. By changing this option, you can alter this behavior by allowing only one term in each request.
    - If hierarchy injection is checked, taxonomy term parents will appear in the breadcrumb trail. For instance, if you have a term that is the child of the term cameras, which is the child of the term electronics then by checking this option you will get electronics, cameras as your taxonomy URL entries.
    - If the last checkbox is checked, all views will be made available as content panes to be added to content types. If it is not checked, only views that have a 'Content pane' display will be available as content panes. This is actually a way of telling Drupal if all your views including the existing ones should be available for placement inside panels.



- In **Home** | **Administer** | **Site building** | **Panels** | **Settings** | **Panel pages** you can alter basically the same properties as in the other two pages, **Administer** | **Site building** | **Panels** | **Settings** | **Mini panels** and **Administer** | **Site building** | **Panels** | **Settings** | **Panel nodes**:
    - Select the default behavior of new content added to the system. If checked, new content will automatically be immediately available to be added to Panels pages. If not checked, new content will not be available until specifically allowed here.



-
    - Check the boxes for all layouts you want to allow the users to choose from when picking a layout. You must allow at least one layout. You can see the available layout in the previous section.

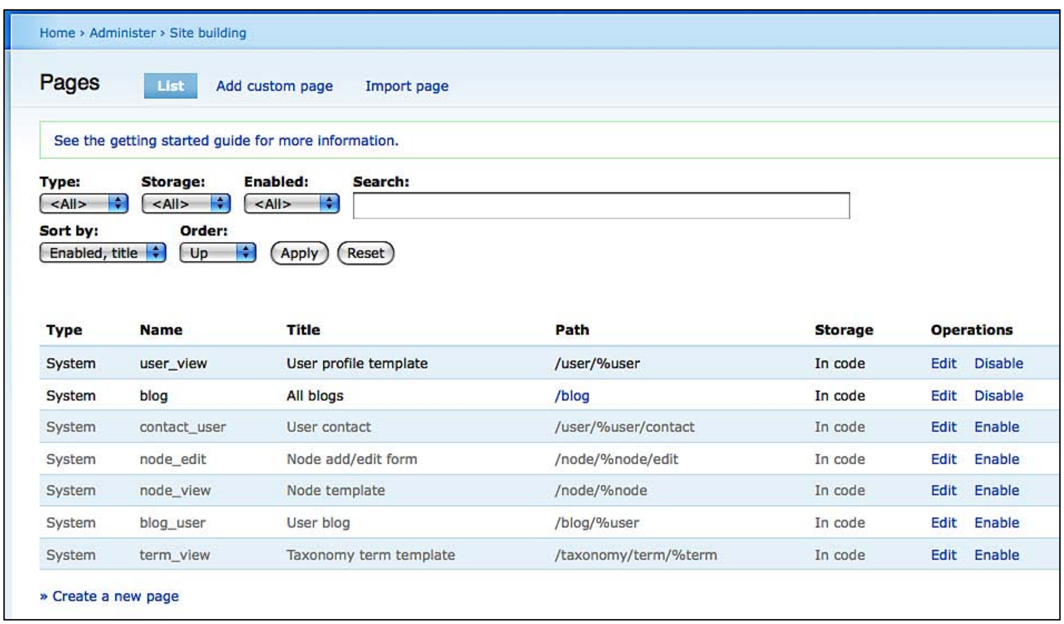
Panels for product layout

Keep in mind that the Panels module provides a vast number of features and functionalities that can customize your online shop to great extent. Following our analysis on Panels module 3 and Drupal 6 we will go through some of these features focusing on first page configuration, some special product categories, and some promotional areas for new products. We need to define two things before we continue:

- What exactly we need to show to our customers in our first page
- The outline of this information

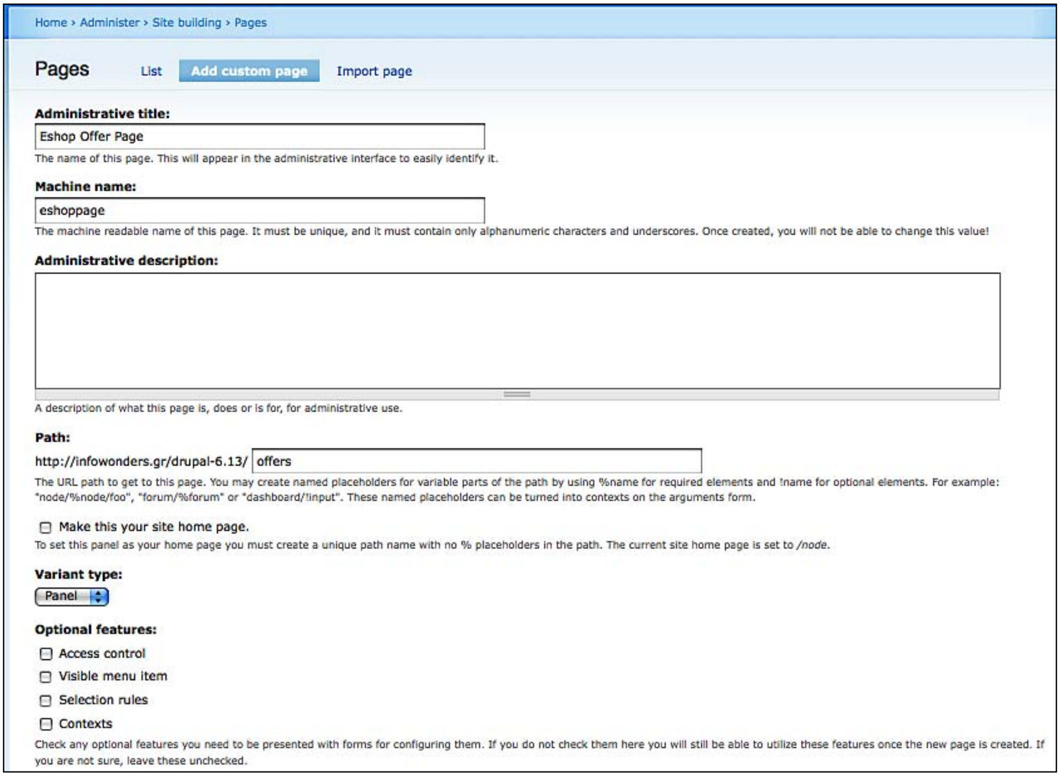
For the first question, in a real-life example we would need to include in our analysis a number of factors like what merchandise we want to promote, where a promotion area should be placed, what lifetime these promotion screens should have, how often they will be updated, and what other information will we show. Questions like these cannot be addressed without the presence of substantial data and reports. Large organizations, in order to perform minor changes to their corporate site profiles or their online shops, tend to get really sophisticated with direct marketing research, focus groups, and in general high-cost approaches. We on the other hand, having all these valuable statistics and reports, which we have set up in *Chapter 7, Managing Customers and Orders*, can take these decisions based on solid data for customer trends and behavior trends. Suppose that our customers produce a high click rate for featured products and camera electronics. We create some views using the views module and position them in our starting page in areas we think will be most visible to ensure a higher click rate.

To do so, we need to navigate to **Administer | Site building | Pages**. Here we can see all the active panels pages that are of type system. Another very useful feature with Panels is that you can use it to override specific Drupal core screens such as the node edit screen, or even get some new cool ways of displaying Taxonomies. You can add variants and context-dependent functionality by clicking on **Edit**, but we will not consider this at the moment.

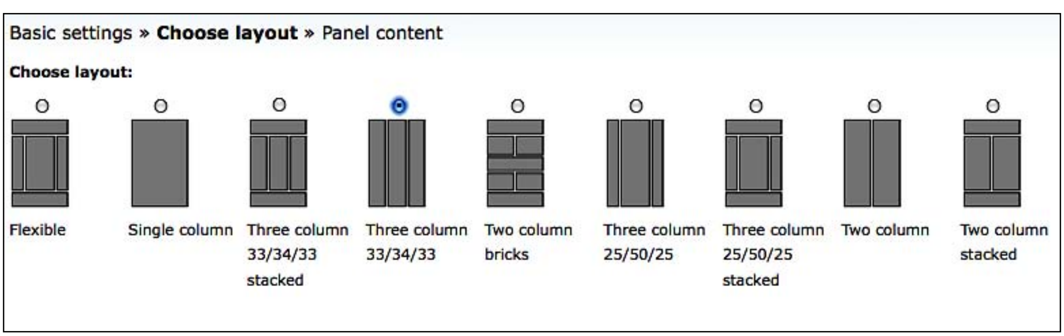


Here you have two different options for creating the panel layout you need: simple panel pages and custom pages that stand for more advanced panels with all the context-aware functionality we have been discussing. So, click on the **Create a new page** link and you will see the new pages creation screen, where we need to add the following information to the form:

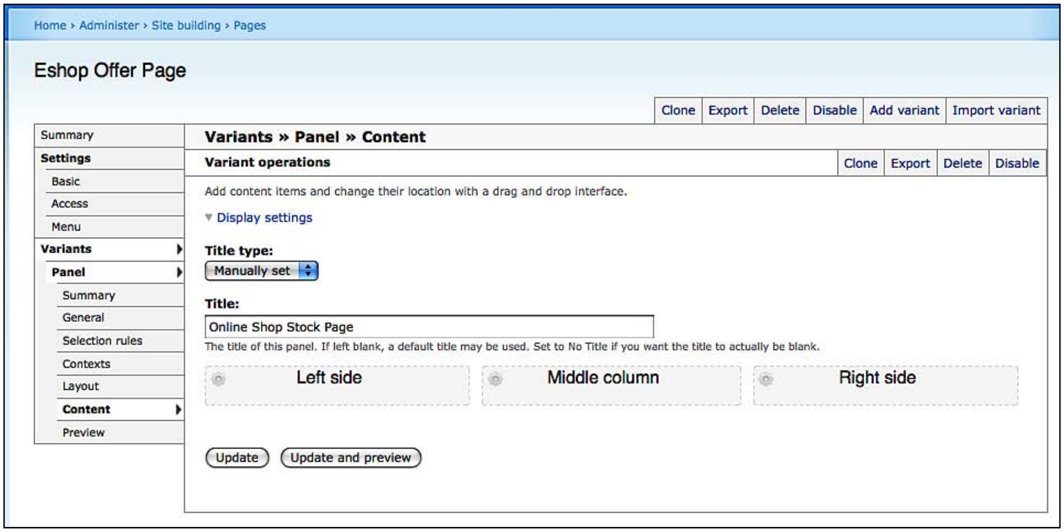
- **Administrative title**: This is the name of the page and will help you to refer to this page from the "pages" screen.
- **Machine name**: This is the machine-readable name of the page. It must be unique, and it must contain only alphanumeric characters and underscores. Once created, you will not be able to change this value.
- **Administrative description**: This is a description of what this page is, and does, for administrative use.
- **Path**: This is the URL path to get to this page. You may create named placeholders for variable parts of the path by using `%name` for required elements and `!name` for optional elements. For example: "`node/%node/foo`", "`forum/%forum`" or "`dashboard/!input`". These named placeholders can be turned into contexts in the arguments form.
- **Make this your home page**: This is a very useful option that helps you use your customized panel page as your home page.
- **Variant type**: For this choose the default value, **Panel**.
- **Optional features**: These are the additional configuration options for your panel screen. You can associate different behavior to different roles, languages and so on. In our case, as we are creating a promotional page with access from all, we will leave this blank.



After filling out the details click on **Continue**. Just like every Drupal wizard screen, keep in mind that nothing is being saved until you finish all the screen sequences. You then have to choose your preferred layout and we will choose the **Three column** view as we would like to have results from three different views displayed in our promotional page.



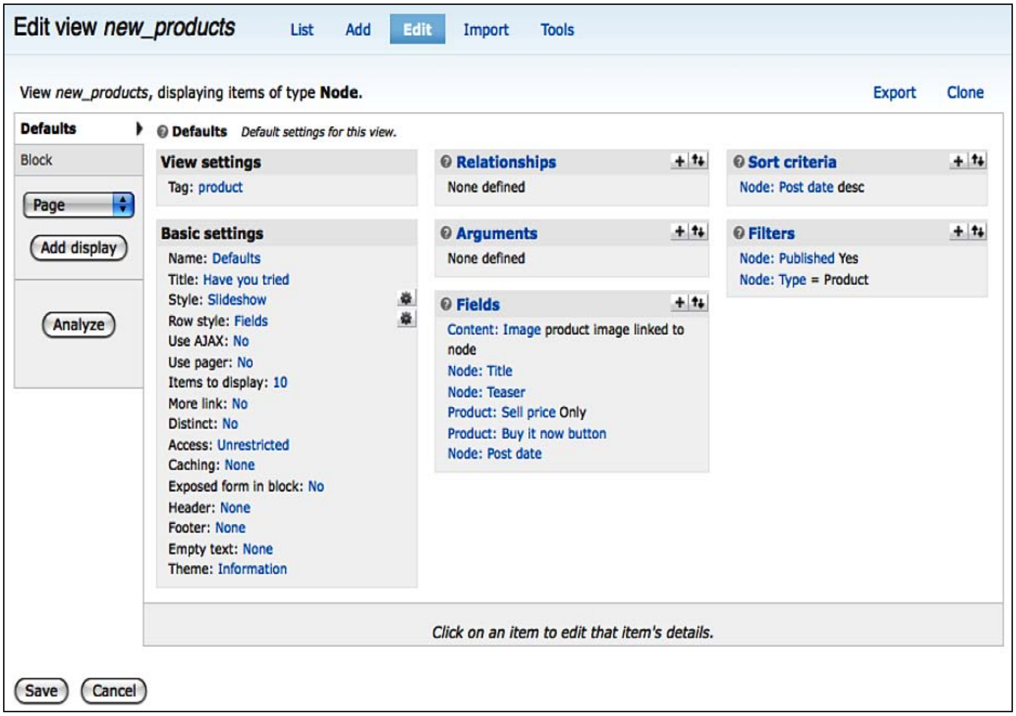
Following the decision on the panel layout, in the next screen we will also be asked to choose a name for the page and configure caching settings. Caching is extremely important for panels as they consist of different views and rendering time can start increasing if we are not a little careful. Usually we will define cashing time and attributes in the Views module but you can do it in the panel section globally in the same way you are doing it in Views, that is by defining data lifetime before the next rendering. So, after you enter the title click on **finish** and your panel definition is ready. You will be redirected to the panel edit page, which is depicted in the following image. The resemblance to Views screens is obvious, while the configuration options span from content layout settings to access rules and positioning.



The next thing we want to do is to try to populate the areas we have assigned to our three-column panel. We will be using two system views from Ubercart and we will manually create another one ourselves for our example.

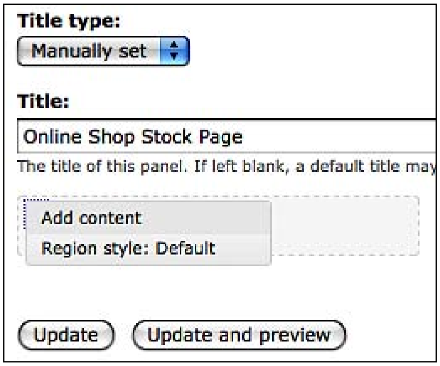
It should be clear by now that the rich usability of panels in Drupal emerges from the integration with the Views module. Literally the best content that you would like to add in your panel areas is a views block. Therefore, we will step right into creating the three views we will assign to the different areas of the panel. Navigate to the views creation page, **Administer | Site building | Views**. To create a view with the most visible products of the day do the following:

1. Click on the **Add view** button.
2. Add to the view a name, the identification "daily products" and the description "daily most viewed products" and click on **Next**.

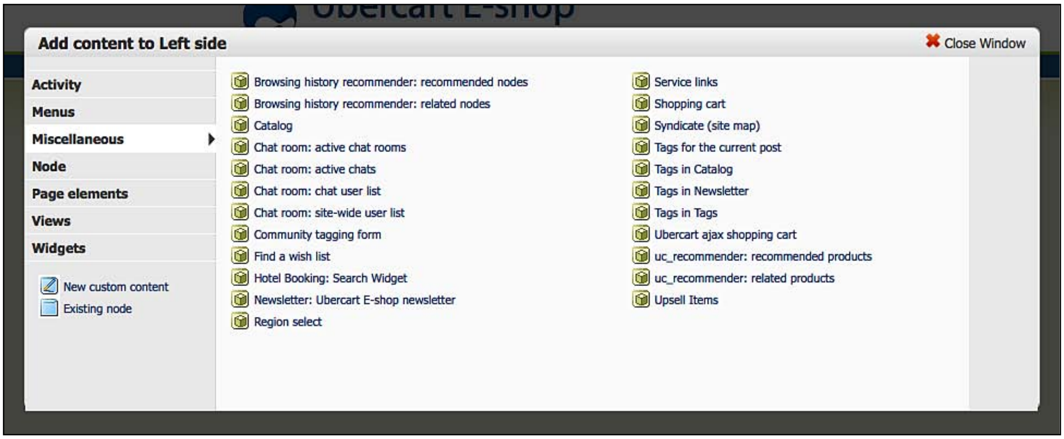


3. In the view configuration screen add a block display by selecting block from the drop-down option list on the top left of the screen, and then click on **Add display**.
4. Ignore the error generated and continue to add a sorting criterion. Click on the **add** icon on the sorting criteria box and in the option list that appears choose "Node Statistics", click on **View today** and choose ordering as descending, then click on **Add**.
5. Then we continue to add a filter. Click the add icon on the filter box and in the option list that appears choose "Node", click on **Node:Type** and choose "is One Of" and the type "Product" then click on **update default display**.
6. At Block Settings box click on caching and choose Cache Once for everyone.
7. Finally, as we are using fields display options (the default) we need to define these fields for our view. We choose to display the product title and the product teaser in our view. To do so click on the **Add** button on the fields box and select **Node** from the drop-down list and then click on **Node:Title** and **Node:Teaser**. Afterwards, click on **Add**. Then in the title field click on **Link this field to its Node** option. Click on **update default display** and everything should be set.

8. After we have created our custom view we will go through and populate the content areas in our panel; we go back to **Administer | Site building | Pages** and click on the page we just created. Go to **Variants | Content** and then click on the little gear-like icon on the top left of the content screen. You will be provided with the following option:

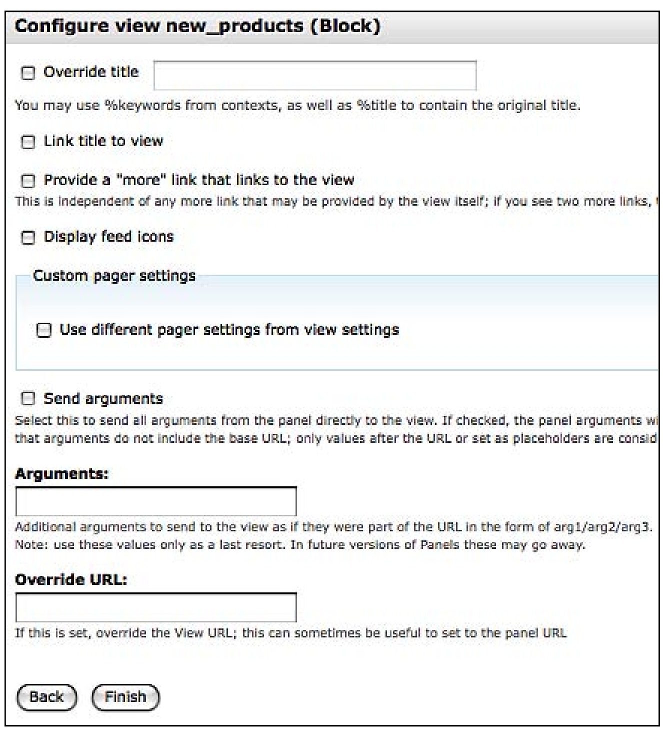


9. Choose **Add content** and a pop-up modal window will appear with all the available blocks of your site. The blocks are categorized so feel free to change between categories and see all blocks available to you. Also keep in mind the option to add new custom content, which stands for manually added HTML and adding existing nodes inside the panel block. We will choose to add from the **Views** tab the following:
    - popular_products
    - new_products
    - dailyproducts



In order to add the view to our panel display we perform the following steps for each view on our panel:

1. Choose the view you want to add to the particular panel area.
2. On the modal pop-up window choose the view you want to add. Usually custom views are found in the views tab. Click on the view you want to add.
3. Afterwards, you will be prompted to choose a display type for your current view. Choose block and click on **Continue**.
4. The next screen provides a large number of configuration options. In our case we will narrow down our changes only to the block title and the view title. You can see the administration options in the following image. Some very interesting options are:
    - The feed icons, especially if this is a feeds view—we recently created a node view in our shop.
    - The argument passing capabilities and how Panels module lets you associate URL parameters to view exposed filters.
5. After you have finished configuring the look and the feel click on **Finish**. Congratulations! You have successfully associated your panel area with the view we created in the previous steps.



After you have finished adding all the three views the result will be very elegant and will definitely resemble the work of an experienced web designer.

Creating custom functionality using Views

As we saw in *Chapter 4, Managing Categories, Products, and Attributes*, Ubercart creates a default Catalog page to help site visitors to browse through categories and subcategories. This is the default functionality for most e-stores and it's absolutely necessary for ours. Customers are used to this way of navigation and it's very convenient for them. If you analyze the stats of your store you'll confirm that the two most common methods that visitors use to locate your products are browsing the product catalog and searching through the search form. Now, if you want to use advanced marketing methods and suggest specific products to your clients, you have to create custom functionality. We'll show you how to enable new custom views and how to create your own.

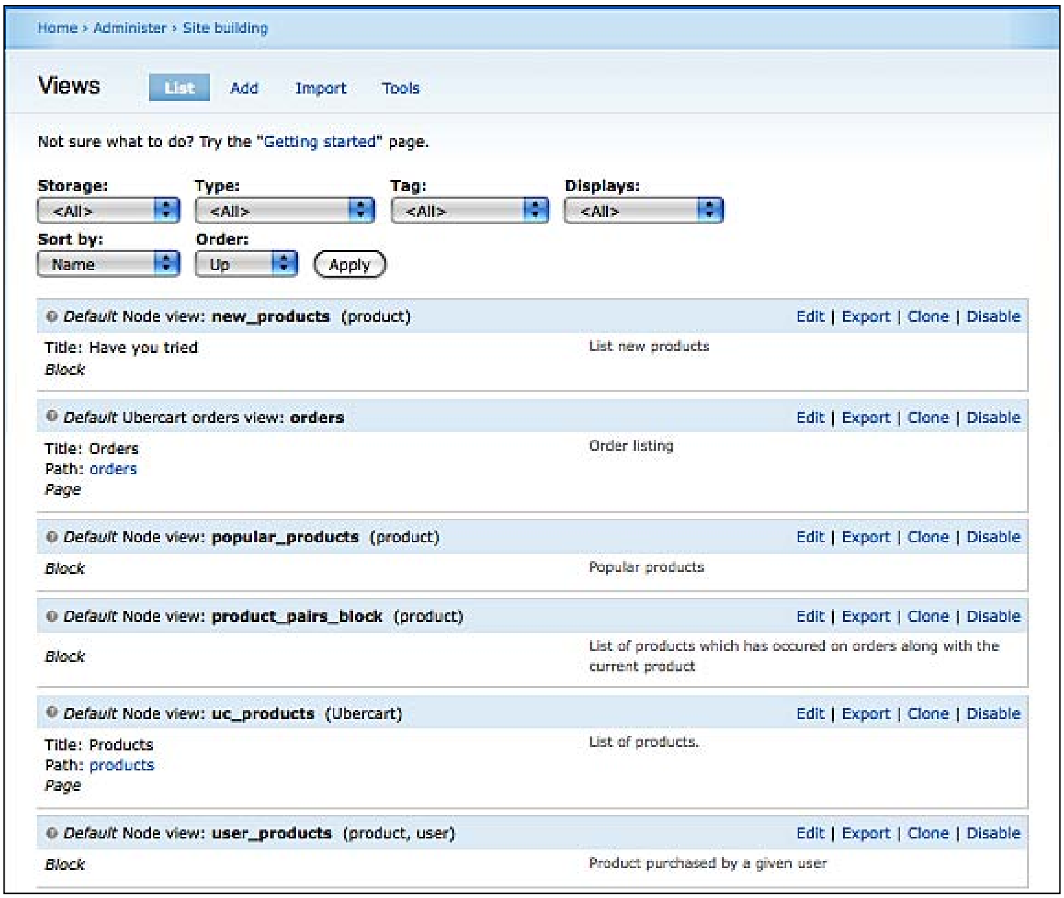
Before we start working, we need to download and enable two new modules, Drupal Views Slideshow and Ubercart Views.

- First navigate to `http://drupal.org/project/views_slideshow` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder. Then, activate the module from **Administer | Site building | Modules**.
- Navigate to `http://drupal.org/project/uc_views` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder. Then, activate the module from **Administer | Site building | Modules**.

Ubercart Views is a module that creates some very useful views for Ubercart. It also lets you create your own ones for your store, using the user interface for views. Let's see the five default views that this module provides. To view or edit them go to **Administer | Site building | Views**.

- **New products**: This view creates a custom block that shows the latest products of your shop as a slideshow. If you want to enable this block, go to **Administer | Site building | Blocks**, and find the block named **new_products**. Assign it to a region and click on **Save blocks**.
- **Popular products**: This view creates a custom block that shows the products with the most sales in your shop as a slideshow. If you want to enable this block, go to **Administer | Site building | Blocks**, and find the block named **popular_products**. Assign it to a region and click on **Save blocks**.

- **Product pairs**: This view creates a custom block that suggests products based on the items that you put in your shopping cart. If you want to enable this block, go to **Administer | Site building | Blocks**, and find the block named **product_pairs_block**. Assign it to a region and click on **Save blocks**.
- **User products**: This view creates a custom block that shows the products with the most sales for this particular user. If you want to enable this block, go to **Administer | Site building | Blocks**, and find the block named **user_products**. Assign it to a region and click on **Save blocks**.



- **Store orders**: This view creates a page, with a more convenient list of the orders in your store. You can access it at `http://www.mystore.com/orders`.

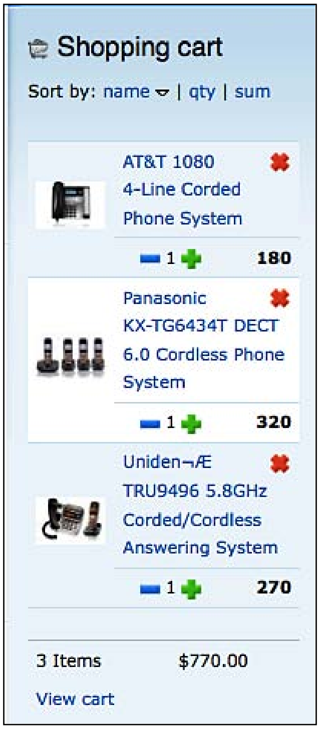
Improving the shopping cart

Sometimes, the shopping cart is the most critical part of an electronic shop. The customer may add a lot of products in it, but if it's difficult to see its contents or to make changes, he or she will leave our store. If we improve our shopping cart, we can reduce cart abandonment and maximize our profit.

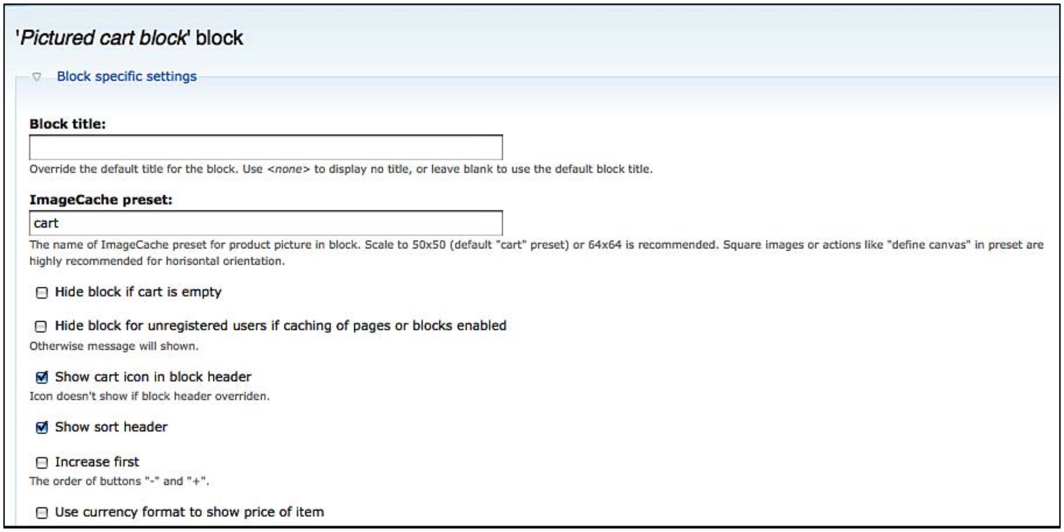
Pictured cart module

This module creates an enhanced version of the Ubercart cart block. The default version is rather basic with few options, but this module adds new and very useful functionalities:

- It shows the images of the products.
- It can be oriented horizontally or vertically.
- The user can sort the contents of the cart by name, quantity, or price.
- It shows extra content such as product description or attributes.



To install it, browse to `http://drupal.org/project/uc_pic_cart_block` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. Then go to **Home | Administer | Site building | Blocks**, find the block named **Pictured cart block** and assign it to a region. Don't forget to disable the default cart block.

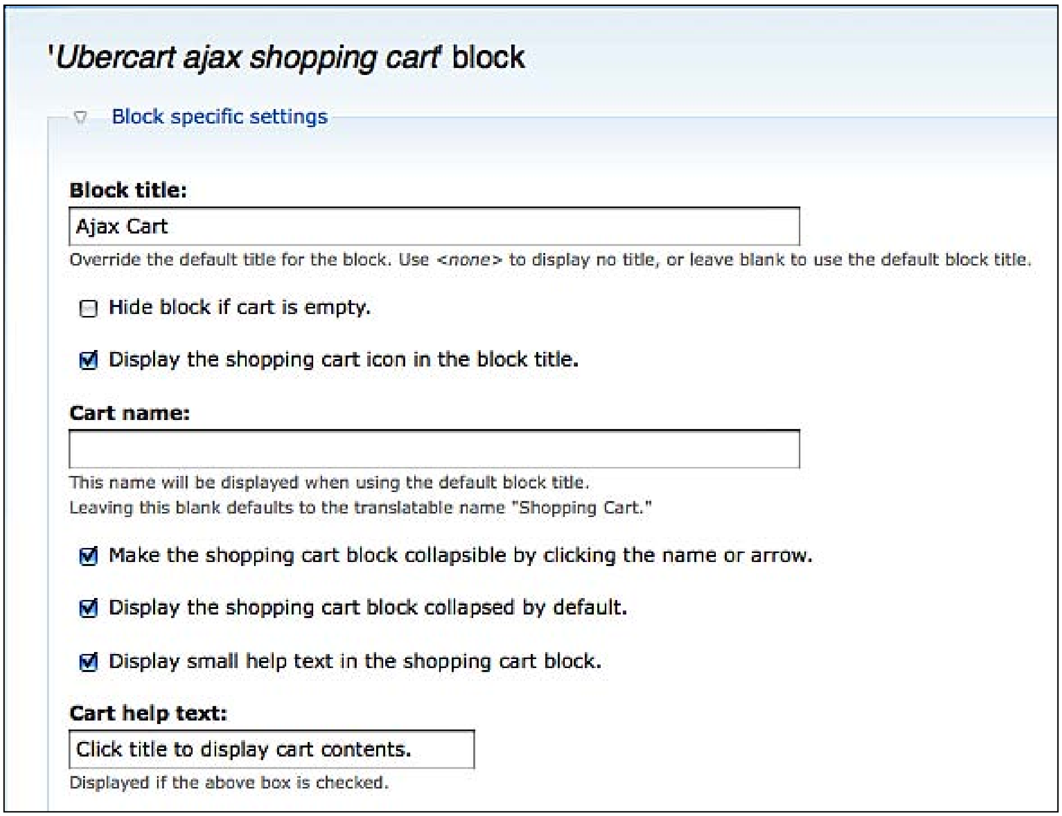


To configure it, click on **configure**. Apart from the standard block options, you can define format and visibility settings of the cart.

Ajax Cart module

This module creates a different version of the default cart block. It uses AJAX, so when the customer adds a product to the cart, it updates its contents without reloading the page. Unfortunately, it is not compatible with the pictured cart module, so you have to select which one is better for your needs.

To install it, browse to `http://drupal.org/project/uc_ajax_cart` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. Then go to **Home | Administer | Site building | Blocks**, find the block named **Ubercart ajax shopping cart** and assign it to a region.

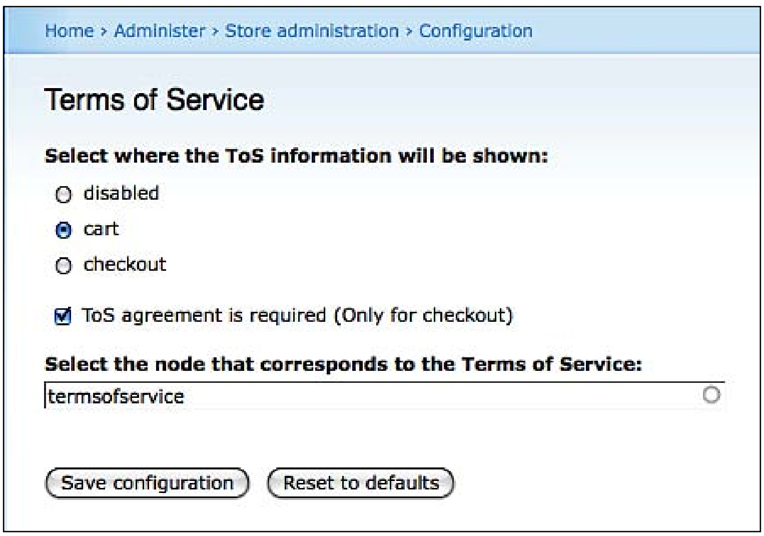


Don't forget to disable the default cart block. To configure it, click on **configure**. There are not only the standard block options, but format and visibility settings as well.

Terms of Service module

This is a very simple module. It just adds a terms or service text in the cart or checkout page. The customer has to accept these terms in order to proceed

To install it, browse to `http://drupal.org/project/uc_termsofservice` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Store administration | Configuration | Terms of Service**. Select the page in which the terms of service will be shown, whether agreement is required, and a node that contains them, and click on **Save configuration**:



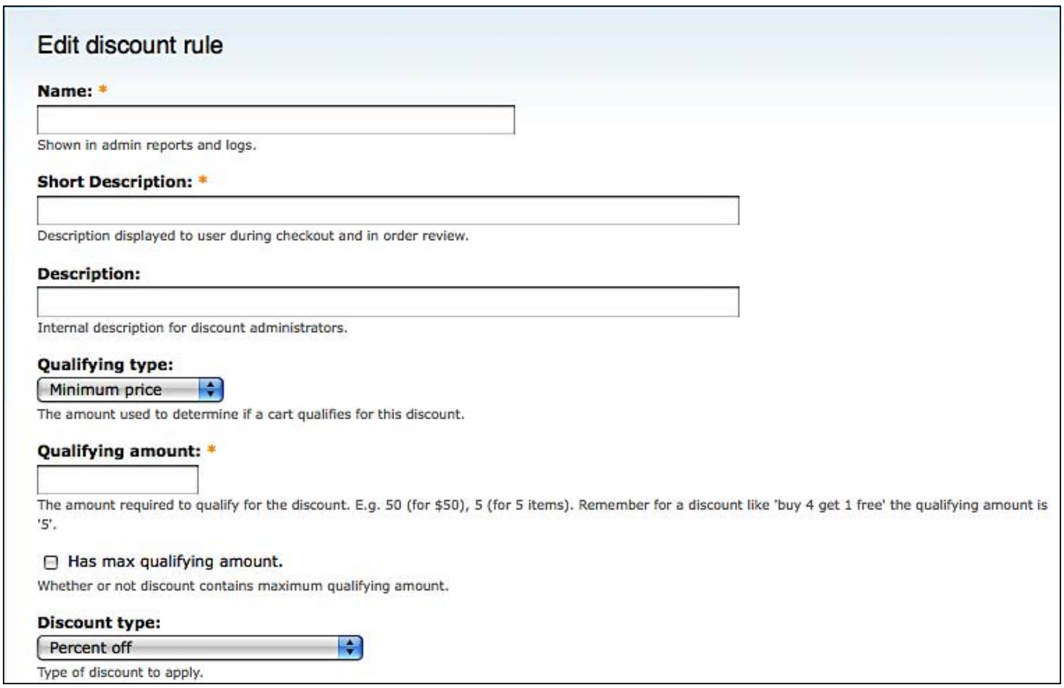
You also have to enable the terms and conditions agreement pane, so go to **Home | Administer | Store administration | Configuration | Cart Settings | Cart Panes**.

Sales and price administration

Offering discounts and special prices to selected users is a move that will make your clients love your store even more and boost your sales. Fortunately, there are some very useful modules for Ubercart that will add this extra functionality to your store.

Ubercart Discounts module

To install the **Discounts module**, browse to `http://drupal.org/project/uc_discounts_alt` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer** | **Site building** | **Modules** to enable it. To configure it, go to **Home** | **Administer** | **Store administration** | **Configuration** | **Discount settings**. If it is your first visit to this page, there will be no discounts yet, so click on **Add new discount** and the following page will appear. You can create a new discount rule as simple or as complex as you like, by using the following settings:



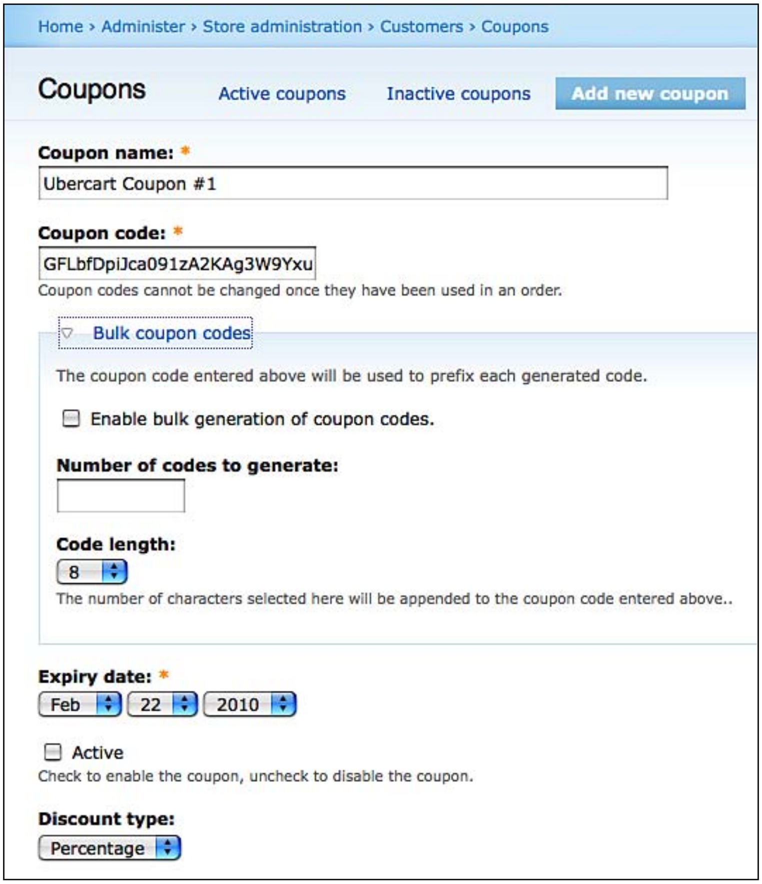
- **Name**: This will be the name of this discount. Enter a descriptive title here.
- **Short description**: Enter a short description for the discount. The customer can read it during checkout and order review.
- **Description**: This description is shown only to administrators.
- **Qualifying type**: You can determine if an order is qualified for this discount based on a minimum price or minimum quantity.
- **Qualifying amount**: Enter the required price or quantity here.
- **Has max qualifying amount**: Turn on this checkbox if this discount has a maximum qualifying amount.

- **Discount type**: This is the type of the discount we will offer to the visitor. The available options are: Percent off, Fixed amount off, Fixed amount off per qualifying item, Free items, Percentage off per qualifying item.
- **Discount amount**: This is the amount of the discount. It is related to the discount type.
- **Require code(s) to activate discount**: If you want the customers to use codes to activate their discounts, turn this checkbox on.
- **Discount codes**: Enter here the discount codes, one code per line.
- **Filter type**: Define how you want to filter you products: by product names, by category terms, or by SKU.
- **Products – Terms – SKUs**: Depending on your previous choice, here you can select your products, terms, or SKUs for this discount.
- **Filter based on roles**: Turn this on if you want to use this discount only for selected roles.
- **Require single product to qualify**: Turn this option on if a single product is required to meet the requirements to qualify for the discount.
- **Maximum times applied**: Define how many times this discount can be applied in the same order, usually **1**.
- **Can be combined with other discounts**: Turn this on if this discount can be combined with other ones in order to give the customer a bigger final discount.
- **Max uses**: How many times this discount can be used in your store. Enter **0** for unlimited.
- **Max uses per user**: How many times this discount can be used by a single user. Enter **0** for unlimited.
- **Discount expires**: Select whether or not this discount will be applicable if other discounts are already active.
- **Is published**: Turn this on to publish this discount.
- **Weight**: If there is more than one discount, the weight defines which one is applied first.
- Click on **Submit** to create this discount. Now your discount is created and your customers will be able to see it during checkout if they meet the requirements that you set.

Ubercart Discount Coupons module

Coupons are a very useful marketing tool. You can distribute coupon codes using printed coupons or by placing ads and promotions on affiliate websites. Customers love promotions and discounts and it's a very effective way to make them shop at your store.

To install the **Discount Coupons** module, browse to `http://drupal.org/project/uc_coupon` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Store administration | Customers | Coupons**, and the following page will appear:



- **Coupon name**: Enter a descriptive name for this coupon.
- **Coupon code**: Enter the coupon code. The customers enter this code to receive the discount.
- **Enable bulk generation of coupon codes**: If you enable this option, the module automatically creates bulk coupon codes.
- **Number of codes to generate**: Enter how many codes you want to be generated.
- **Code length**: Select the length of the generated codes.
- **Expiry date**: Enter the expiry date of this coupon.
- **Active**: Select if this coupon is active or not.
- **Discount type**: The available selections are **Percentage** or **Price**.
- **Discount value**: Enter the discount value. It is of course related to the discount type.
- **Minimum order limit**: Enter the minimum order total that qualifies for this coupon.
- **Maximum number of redemptions (per code)**: How many times this code can be used in your store. Enter **0** for unlimited usage.
- **Maximum number of redemptions (per user)**: How many times this code can be used by a single user. Enter **0** for unlimited usage.
- **Product classes**: Select one or more product classes on which this coupon can be used.
- **Applicable products**: You can restrict the usage of this coupon to only for selected products by selecting a product name.
- **Applicable SKUs**: You can restrict the usage of this coupon to only for selected products by selecting a product SKU.
- **Applicable taxonomy terms**: You can restrict the usage of this coupon to only for selected catalog terms.
- **User restrictions**: You can restrict the usage of this coupon to only by selected users.
- **Role restrictions**: You can restrict the usage of this coupon to only by selected roles.
- **Wholesale permissions**: Leave the default option, both wholesale and retail.
- Click on **Submit** to create this coupon.

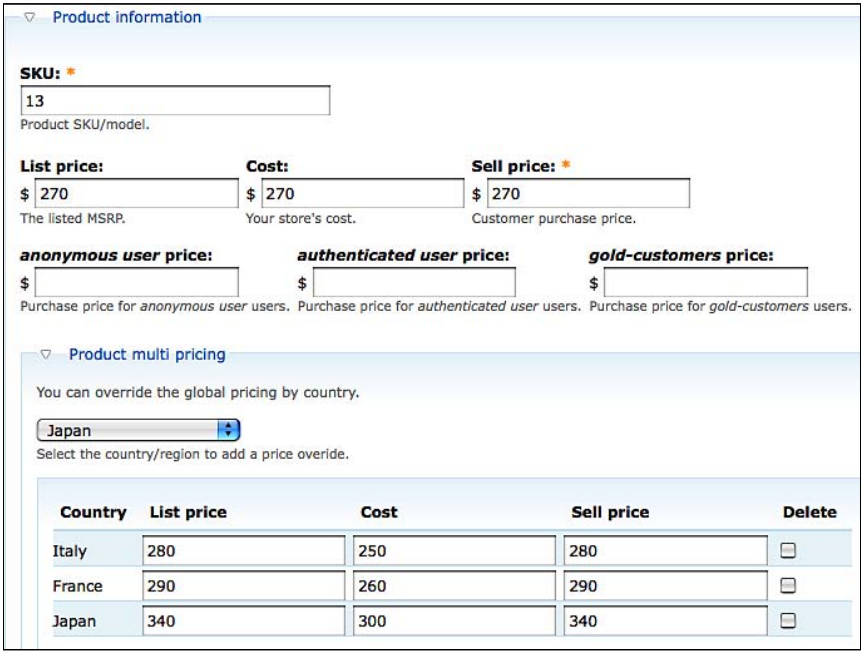
Now, there is a new region in the checkout page, where the customer can enter the **Coupon code** and get his or her discount. This region can be seen in the following screenshot:

After entering the code click on **Apply to order** to apply the coupon to your order.

Ubercart Multiprice module

The **Ubercart Multiprice** module solves a very important problem for international retailers. If you sell your products to many different countries, you may want to sell them at different prices for every country. It creates a block where the customer selects the country and it shows the correct price. Even better, it can combine with the IP to country module (`http://drupal.org/project/ip2cc`) and automatically detect the customer's country from his or her IP address.

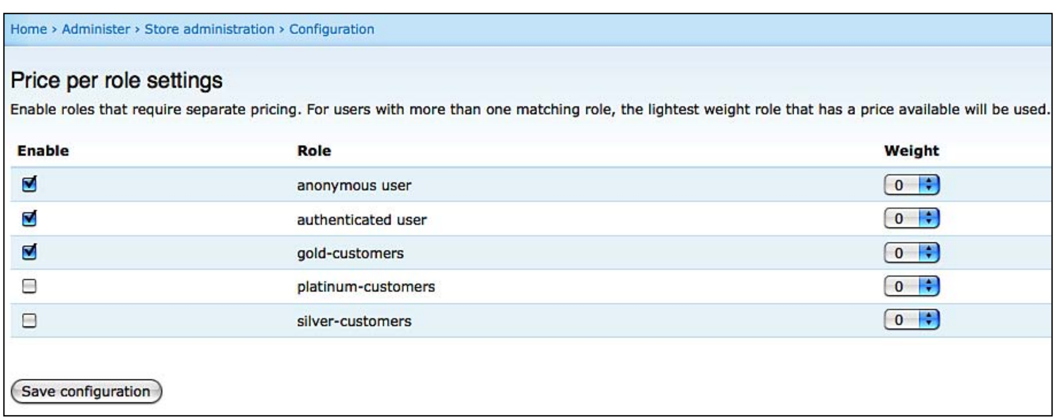
To install it browse to `http://drupal.org/project/uc_multiprice` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. Then go to **Home | Administer | Site building | Blocks**, and enable the **Region select** block. When you create a new product or edit an existing one, there is a new region just below the default price fields, named **Product multi pricing** as shown in the following screenshot:



Here you can select a different price for any country you want by overriding the default prices. When the visitor selects a country with an assigned price from the block that you enabled, the page reloads and shows the price for this specific country.

Ubercart Price Per Role module

With the **Price Per Role** module, you can assign different prices to different roles. For example, you can create a new role named "gold-customers" and can offer them special prices on selected products.

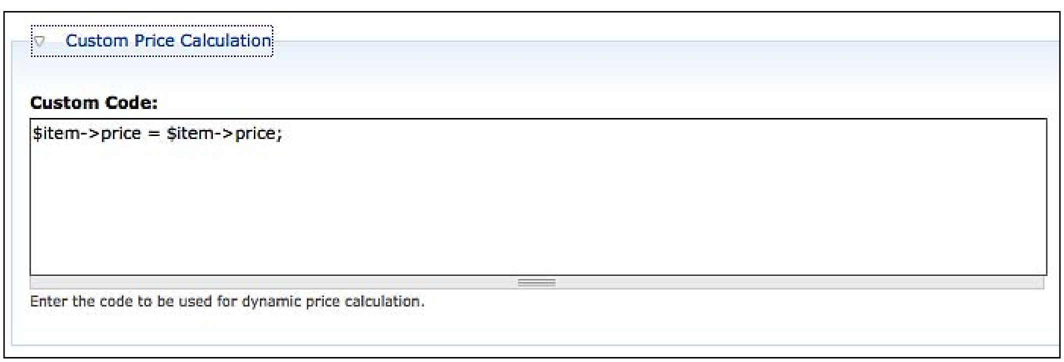


To install it, browse to `http://drupal.org/project/uc_price_per_role` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer** | **Site building** | **Modules** to enable it. To configure it, go to **Home** | **Administer** | **Store Administration** | **Configuration** | **Price per role settings**, and you will get a settings page as shown in the previous screenshot. There you select the roles with special prices.

When you create a new product or edit an existing one, there are new price fields for every selected role. Each visitor will now see the right price, related to his or her role.

Ubercart Custom Price module

This is a module for advanced users. It creates a blank field where you can add custom PHP code to adjust the price of your product. To install it, browse to `http://drupal.org/project/uc_custom_price` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer** | **Site building** | **Modules** to enable it.



When you are creating a new product, or editing an existing one, there is a field named **Custom Price Calculation**, where you can insert your custom code as shown in the previous screenshot.

Summary

In this chapter we showed you how to offer a better browsing experience to your visitors, and how to use enhanced marketing techniques to attract more clients and to make them spend more money in your store. In the next, final chapter of this book, we are going to give you advice on four very important topics: Search Engine Optimization, Internet marketing, Performance, and Security.

10

Optimizing and Promoting Your Store

Well, here we are, at the last chapter of our little adventure. Your shop is completed, everything is in place, and maybe it's already online. You can consider the project completed and start selling without problem. At this point, we ask you to do us (and yourself) a little favor. Invest a little more time in learning four very important tasks:

- Search engine optimization
- Internet marketing
- Performance
- Security

We are going to implement most of this functionality by using and customizing some third-party Ubercart modules, so the process is rather simple. We are also going to show you how to use some very useful tools for site marketing and promotion, most of them provided by Google.

Search egin optimizatio

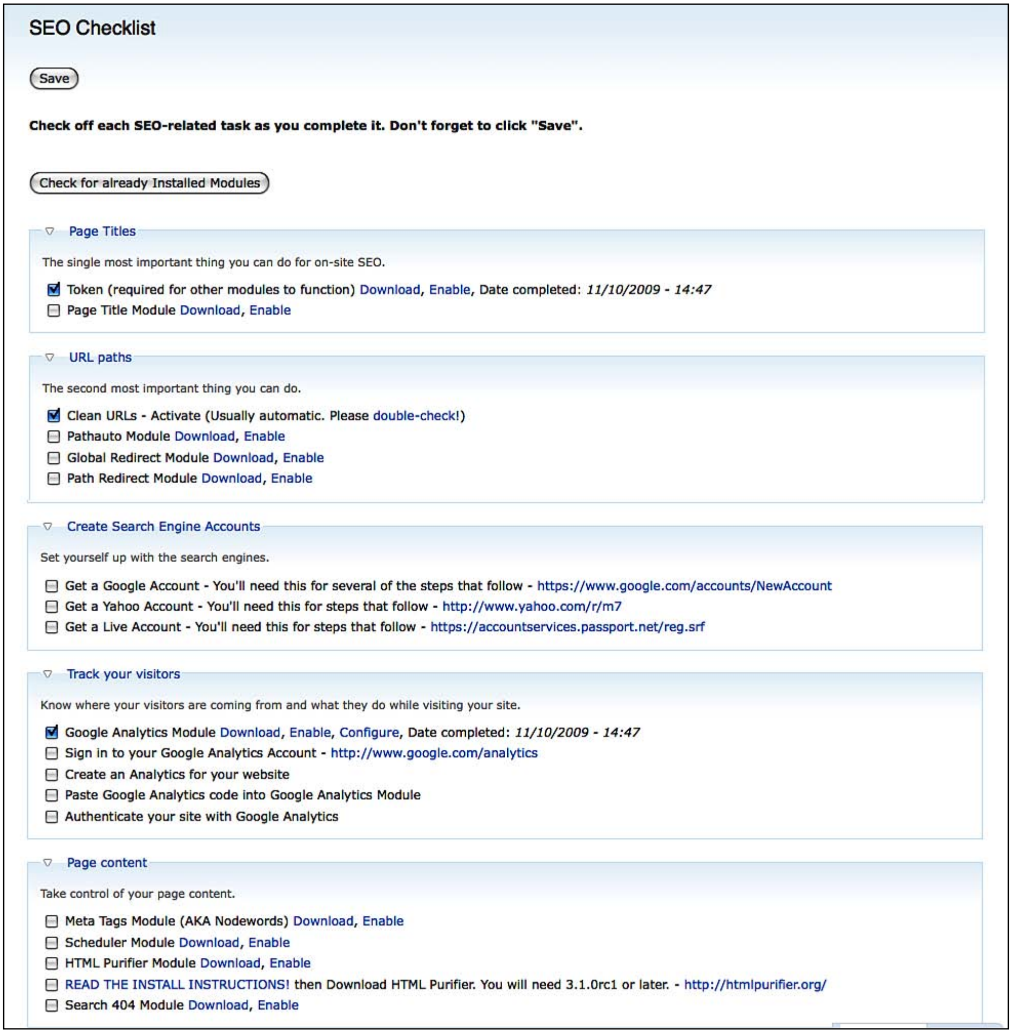
You have many ways to promote your website, but the main traffic source will always be search engines. Search engine optimization helps your site to improve its position in the natural search results, thus generating more traffic and attracting visitors who search for your products.

Our guide for the search optimization process will be the amazing Drupal SEO Checklist module. As its name suggests, it doesn't add any new functionality to our site, but it suggests useful modules and checks that they are installed and properly configured. So, our first step will be to set up this particular module. After that, we'll start to check actions of the checklist, until all actions are completed.

The Drupal SEO Checklist module

Let's start by installing and exploring this module. Browse to `http://drupal.org/project/seo_checklist` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. To view the SEO checklist and start working with it, go to **Home | Administer | Site configuration | SEO Checklist**.

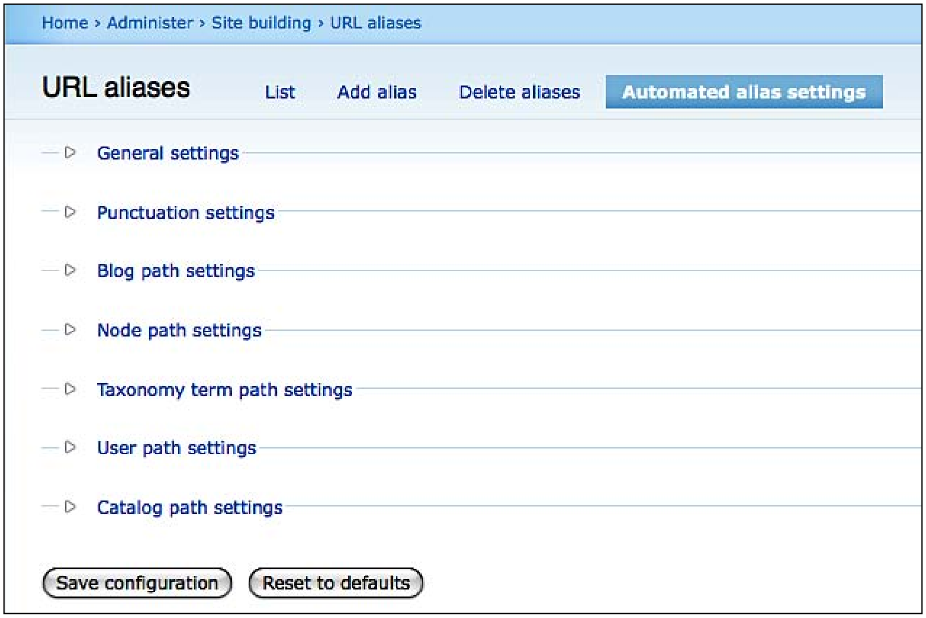
Your first action in this page is to click on **Check for already Installed Modules**. It scans your Ubercart installation and automatically checks off all the installed modules. Don't forget to click on **Save**. Right now you have only a few of them installed, so we'll start to install and configure most of them, one by one.



Pathauto module

This module automatically creates path aliases for our nodes, categories, and users. It generates search engine-friendly URLs and improves the ranking of our pages. Browse to `http://drupal.org/project/pathauto` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Site building | URL aliases**.

If you click on **Add alias**, you can manually create an alias for an existing path. Enter the name of the existing system path in the first field, the name of the path alias in the second field, and then click on **Create new alias**.

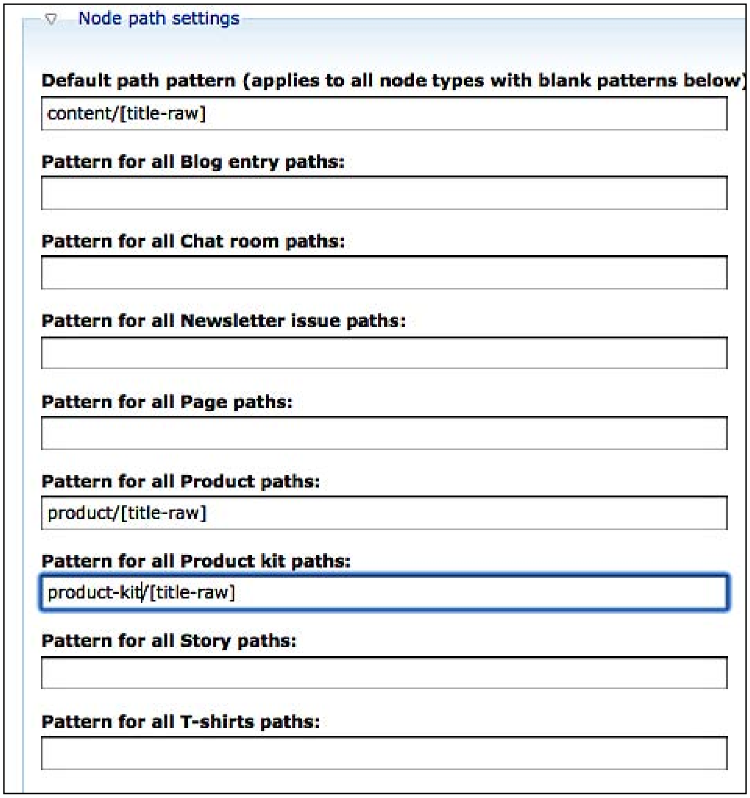


If you click on **Automated alias settings**, a new page will open with dozens of available settings. Here, you can fine-tune your alias settings just once and then you can leave Pathauto module to automatically create a new alias each time you create a new node, category, or user, based on the criteria that you entered. The available setting groups are general settings, punctuation settings, blog path settings, taxonomy term path settings, user path settings, and catalog path settings.

You don't have to alter every available option, because most of them are good enough for your store. We're just going to show you how to make a minor customization, and create a custom alias for your product and product kit pages.

Click on **Node path settings**. The collapsible pane opens. Find the **Pattern for all Product paths** field, and add **product/[title-raw]**. This creates a custom alias by using the word product followed by a slash and the title of the product. Then find the field **Pattern for all Product kit paths** and add this: **product-kit/[title-raw]**. This creates a custom alias by using the word product-kit followed by a slash and the title of the product kit.

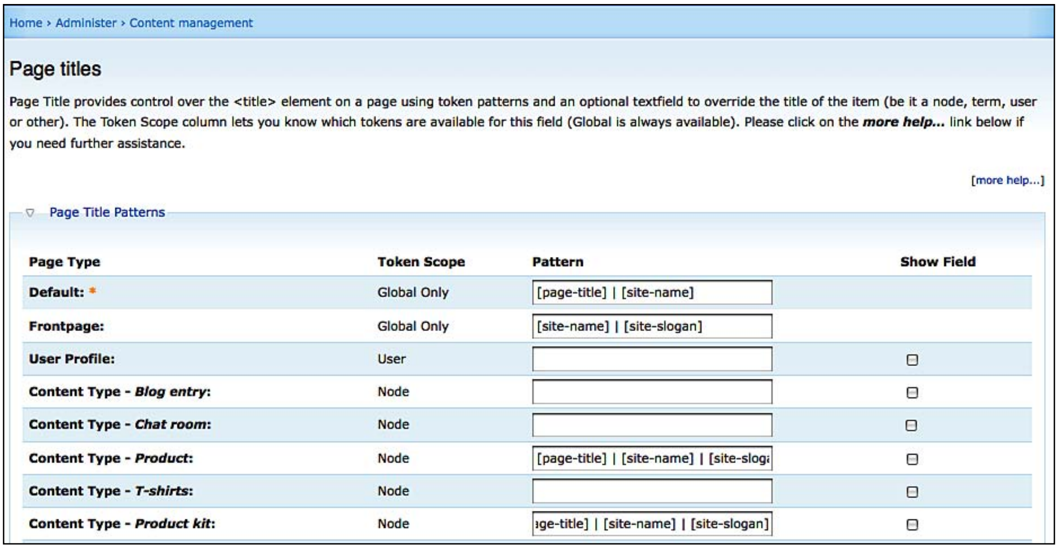
Finally, click on **Save configuration**. Now, every time you insert a new product or product kit, a custom alias will be created based on the above settings.



Page Title module

This module automatically creates <title> elements for our pages. It also enables administrators to create a page title different from the node title. Browse to http://drupal.org/project/page_title, and right after you download the module, upload it and unzip it to your site's /sites/all/modules folder, and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Content management | Page titles**.

Here, we are going to create custom patterns for the titles of product and product-kit pages. We are going to use the node title, followed by the site name and the site slogan. Find the field named **Content Type – Product** and add this custom text: **[page-title] | [site-name] | [site-slogan]**. Do the same for **Content Type – Product kit**.



Now, each time you create a product or product kit, a custom page title will be created. From experience we know that it is not enough. There are times when you need to enter by hand a page title that is totally different from the node title. To enable this feature, go to **Home | Administer | Content management | Content types | Edit Products**, enable the **Show field** checkbox at the bottom of the page, and click on **Save content type**.

Now, if you go to **Home | Create content | Product**, you will see a new field named **Page title**, where you can enter your custom title of the page.

Global Redirect module

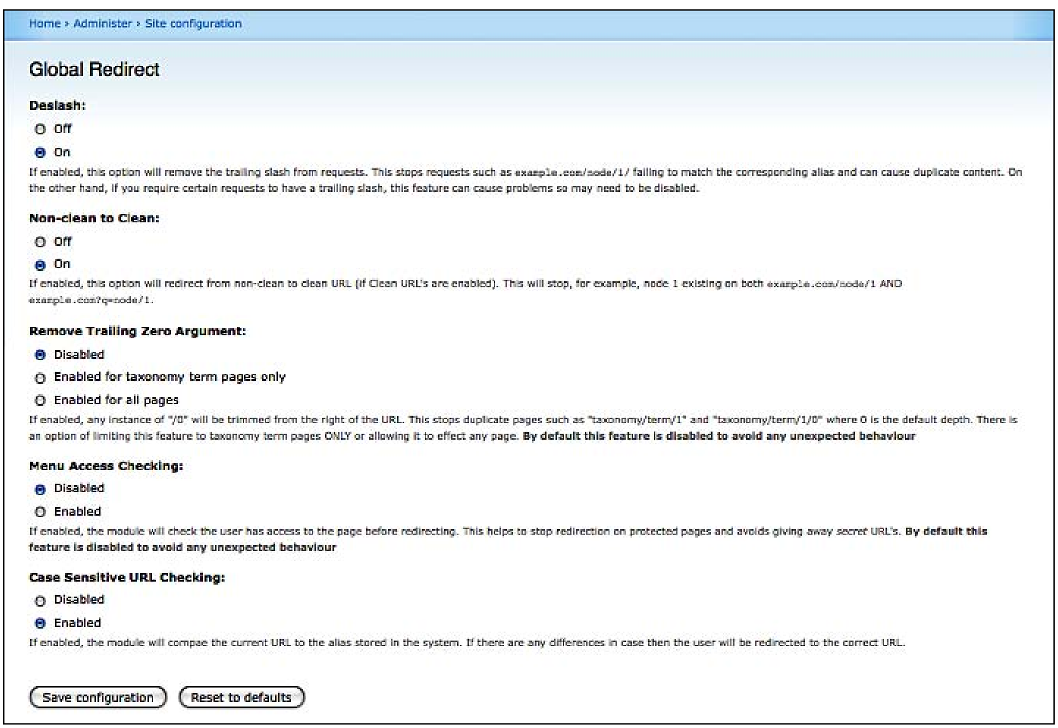
This is a simple but very useful module. In Drupal, especially when you're using clean URLs and Pathauto module, the same content can be reached using different URLs. For example, `node/34`, `node/34/`, `index.php?q=node/34`, and `products/ipod32mb` are different URLs that may target the same page. This might cause problems, because if Google and the other search engines locate different URLs with the same content, they may consider you a spammer and punish you with the duplicate content penalty and lower rankings. Global Redirect module does 301 redirects from all the alternative URLs to the best ones available, thus preventing the duplicate content penalty.

To install it, browse to `http://drupal.org/project/globalredirect` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder. To configure it, go to **Home | Administer | Site configuration | Global Redirect**.

There are five options on this page:

- **Deslash**: This option removes the trailing slash from requests.
- **Non-clean to Clean**: Enabling this option will redirect from non-clean to clean URLs.
- **Remove Trailing Zero Argument**: This option trims any instance of `/0` from the right of the URL.
- **Menu Access Checking**: This option will check if the user has access to the page before redirecting.
- **Case Sensitive URL Checking**: When enabled, it compares the current URL with the stored alias and checks if there are any differences.

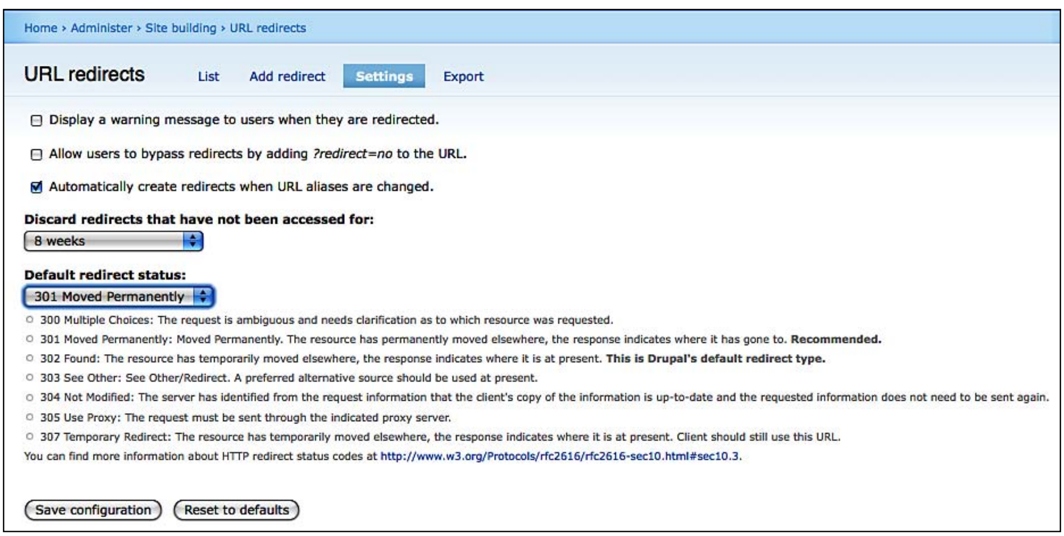
These options define how the variations of Drupal URLs will be handled by this module. You don't have to change anything in this page, because the default options work perfectly for most cases.



Path redirect module

This module also uses 301 redirects, but for a different reason. It helps you to create redirects from old or deactivated pages to new ones. For example, you may want to migrate from your old static site to a new Drupal site. Your pages will now have different URLs. That means that people who have bookmarked your pages or websites having links to your website will now receive a "Page not found" message. The most important problem is that search engines will also show broken links for some time. Then, when they start indexing your new page, you'll lose all the good ranking of the old pages and you'll have to start your SEO efforts from scratch. By using this module, you can easily redirect your old pages to the new ones without losing important traffic. To install it, browse to `http://drupal.org/project/path_redirect` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Site building | URL Redirects**.

In this first page, you can see a list of the available redirects. As we haven't created any redirect yet, this list is still empty. Click on the **Settings** tab to define how exactly the redirects are going to work.



If you enable the first checkbox, the users will see a warning message every time they are redirected. This is rather annoying in most cases, so leave this unchecked. If you enable the second checkbox, you allow your visitors to bypass redirects by adding `?redirect=no` to the URL. This is not useful for your store, so leave this also unchecked. If you enable the third checkbox, this module automatically creates redirects when URL aliases are changed. This is a life saver in most cases, because if it is not checked, you have to return here and change the details of the redirection every time that you change a URL alias.

In the next menu, we can not only select whether we want to disable inactive redirects, but also define after how many days a redirect that is not accessed should be considered as inactive.

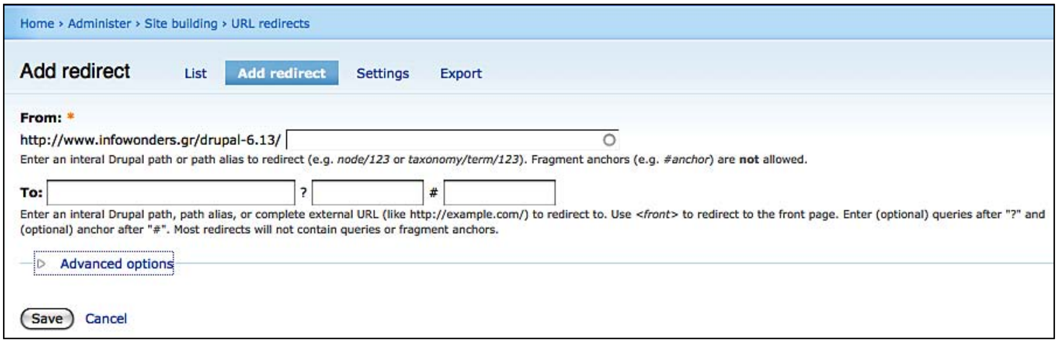
In the last menu, you can select the default redirect status. The available options are:

- **300 Multiple Choices**
- **301 Moved Permanently**
- **302 Found**
- **303 See Other**
- **304 Not Modified**
- **305 Use Proxy**
- **307 Temporary Redirect**

The most commonly used option is **301 Moved Permanently**, so select this option for your redirects. Finally, click on **Save configuration** to store the changes.

To create a new redirect, click on the **Add redirect** tab. In the **From** field, enter the internal Drupal path or path alias that you want to redirect. In the **To** field, enter the URL that you want to redirect to. Here, you can use a new internal Drupal path or path alias, or even an external URL. Click on **Save** to store the redirect.

If you want to export your redirects, click on the **Export** tab, select the **Export format**, (CSV or XML) and click on the **Export** button.



Meta tags module

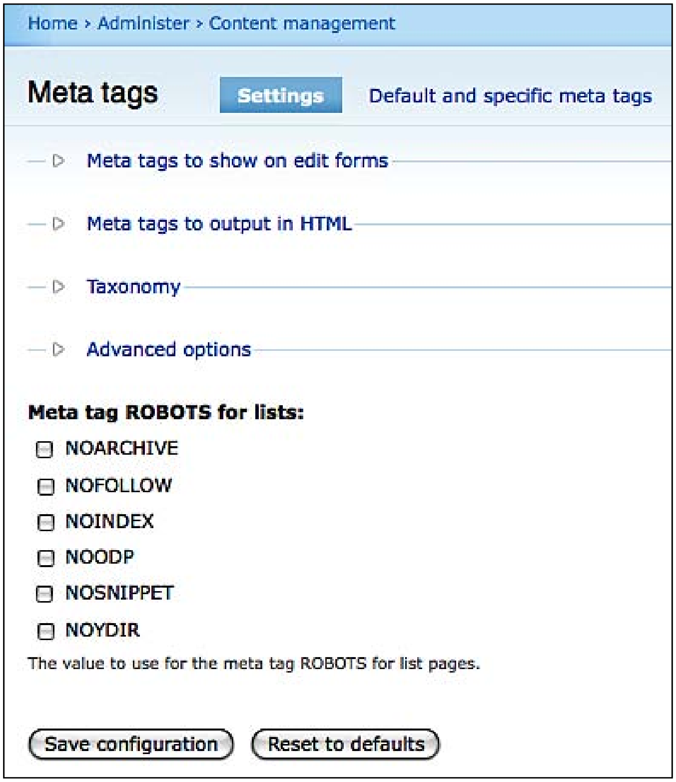
Meta tags are special elements in the HTML code of your web pages. They are located in the header of the page, and they are not visible to site visitors. Only search engines can view them. Their purpose is to give the search engines information about the content and the details of your web pages, in a structured and predefined way. There are many meta tags available, but the most common and useful are the following ones:

- **Keywords**: This meta tag lists keywords that are related to the content of the web page. Nowadays, more search engines, including Google, do not take it into account.
- **Description**: This meta tag gives a short description of the page and appears in the search results of Google.
- **Copyright**: If your work is copyrighted, you can use this to declare it.
- **Robots**: This meta tag defines if the page will be indexed by search engine spiders or not.

This module allows you to enter some general rules for creating meta tags in every page, but it also gives you the opportunity to manually create meta tags for every node, every view, and every taxonomy term.

To install it, browse to `http://drupal.org/project/nodewords` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Content management | Meta tags**.

Let's explore all the available options of the Settings page. As you can see in the following screenshot, there are a few collapsing panes and we'll start to explore them one by one:



The **Meta tags to show on edit forms** option is actually a big list of all the available meta tags. You have to select which meta tags you want to be visible when an administrator edits a node. Select **Keywords**, **Description**, and **Robots**. The **Meta tags to output in HTML** option actually presents the same list of meta tags, but now you have to select which meta tags you want to be used in your code. Select **Keywords**, **Description**, and **Robots**. In the **Taxonomy** area, you can select the vocabularies that contain terms that can automatically be used as keywords in the pages of your website. Select the **Catalog** vocabulary, because adding the catalog terms as keywords is a helpful option. The **Meta tags creation options** area has many options available:

- **Enable the user profile meta tags**: Turn this on if you want to use meta tags in the users' profile pages.
- **Repeat meta tags for lists**: If there are big lists in content, for example in a view, Drupal uses pagination to divide the content into easy-to-read pages. If you turn this on, the same meta tags will be used in every page, but if you turn it off, it will be used only in the first page. It's better to keep it off, because sometimes search engines punish your website if you use the same meta tags in many pages.
- **Use front page meta tags**: The front page of your store is not defined from the start. It can be a list of your latest products, a text with your company profile, or a combination of them together with photos, custom views, and so on. If you want to use specific meta tags in your front page, no matter what its content is, you have to enable this option.
- **Maximum meta tags length**: Here you can define the maximum length, in characters, for the meta tags input field. The default value is fine, so you don't have to change anything here.

The **Meta tags content generation options** area defines how a meta tag content is generated from the node content. Lets see this in more detail:

- First, you have to select if you want to generate meta tag content when the meta tag content is empty, or if you don't want to generate meta tags content at all. The automated creation of meta tag content is useful, so go on and enable it.
- In the second option, you define the source of generation for meta tags content. The most useful option is the third one, generate meta tags content from the node teaser, or the node body when the node teaser is empty.
- In the next option, enable the checkbox **Replace the tag IMG content with the attribute ALT**.
- Finally, leave the two last options empty, as you don't have to filter the text in the node teaser or to use a custom regular expression.

The **Taxonomy options** area defines how a meta tag content is generated from taxonomy terms.

- **Auto-Keywords Vocabularies**: Selecting one more vocabularies from this list automatically adds their terms as keyword meta tags from nodes.
- **Base URL**: Enter here the base URL. It will be used for canonical URLs.
- **Global keywords**: This is a list of keywords that will be added on every page of your site.
- **Meta tag ROBOTS for lists**: We mentioned before what lists are. Here you can choose one or more meta tags to use for the Robots meta tag for lists. For your store, there is no need to choose any of them.
- **ICRA validation content**: ICRA (`http://www.icra.org`) is part of the Family Online Safety Institute and its purpose is to protect children from potentially harmful material. Using their tools, you can label your site, but you have to use some validation content. Enter this content here and it will automatically be added to your site.

> There is a very nice article about the Robots meta tag, at the Official Google Webmaster Central Blog: `http://googlewebmastercentral.blogspot.com/2007/03/using-robots-meta-tag.html`.

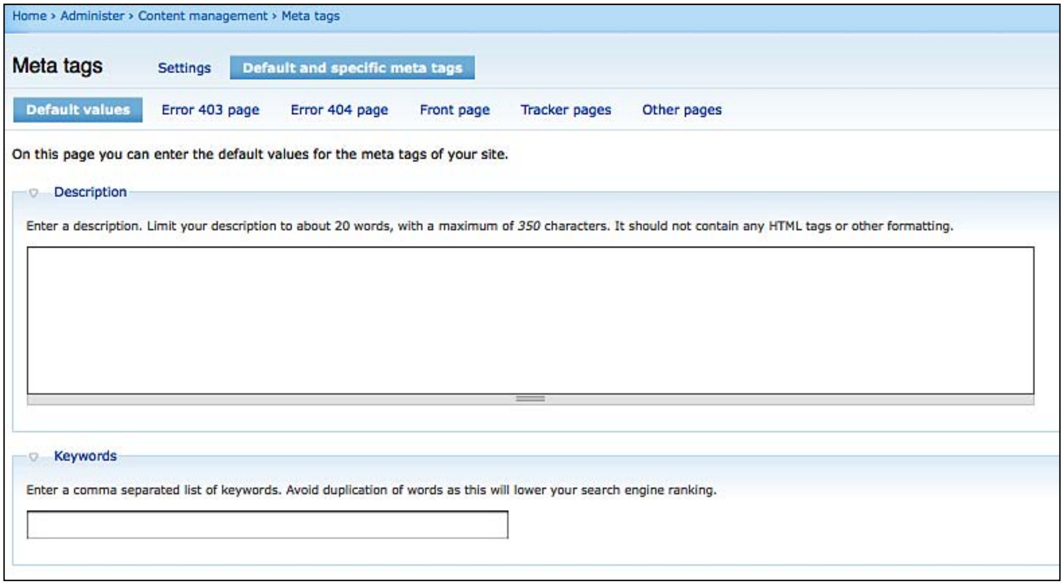
Finally, click on **Save configuration** to store your settings.

Now, click on the **Default and specific meta tags** tab. Do you remember the meta tags that we chose in the settings page? They were **Description**, **Keywords**, and **Robots**. Here, you can insert default values for these meta tags in the **Default values** section.

There are also separate sections for **Error 403 page** (access denied), **Error 404 page** (page not found), **Front page**, and **Tracker Pages**. If that's not enough for you, you can go to the **Other pages** section, where you can add meta tags to any page of your website.

Now that we're finished with the settings, let's see this module in action. We'll edit a vocabulary and a product, to see what changes it causes to the edit form. Go to `/admin/content/taxonomy/edit/vocabulary/1` to edit the Catalog vocabulary. There is a new pane, named **Meta tags**, where you can enter values for the three meta tags that we selected previously.

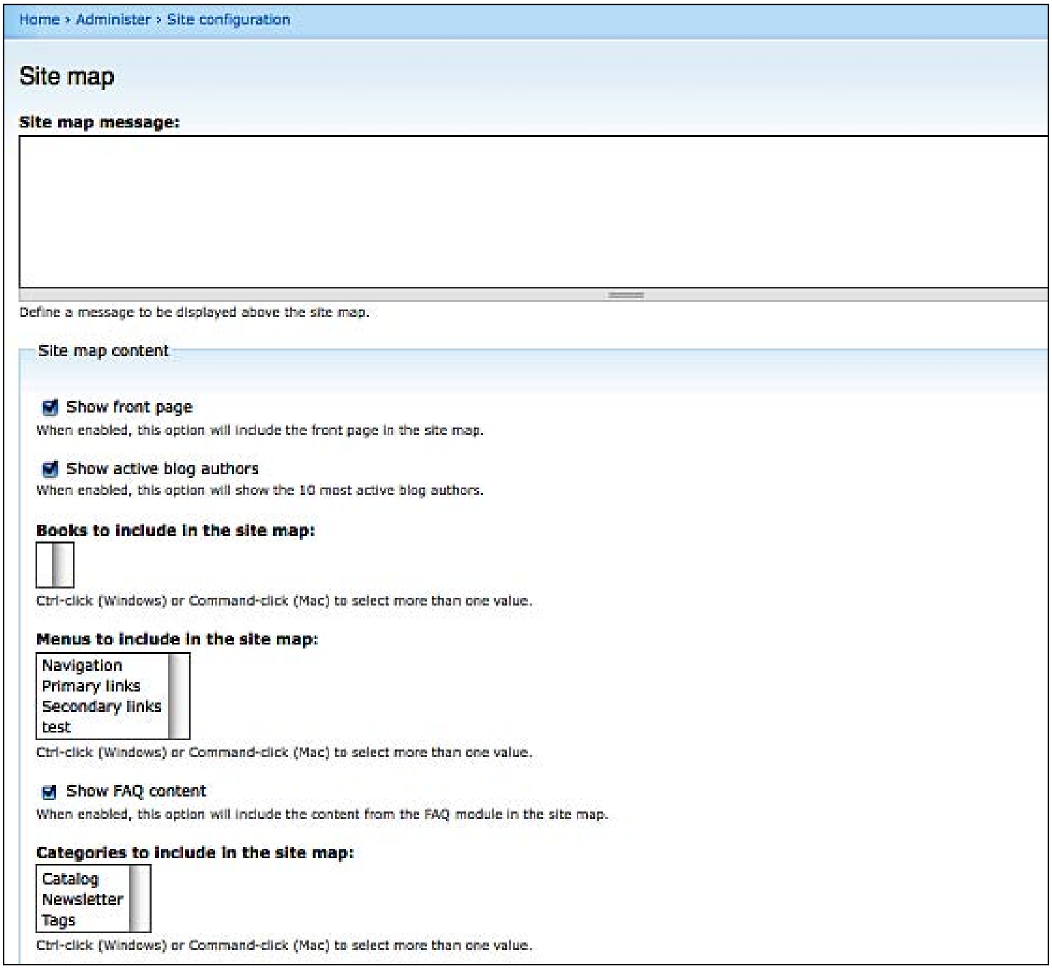
The same happens when you edit a product or create a new one. Usually, the most useful combination is to enter some default meta tags from the settings page for the entire website, some new meta tags for every vocabulary, and finally extra meta tags for every node, if you think that it's necessary.



Site map module

Site map module creates a page with a site map in a readable form. The visitors of your site can view the structure of it, but it's also accessible by search engines.

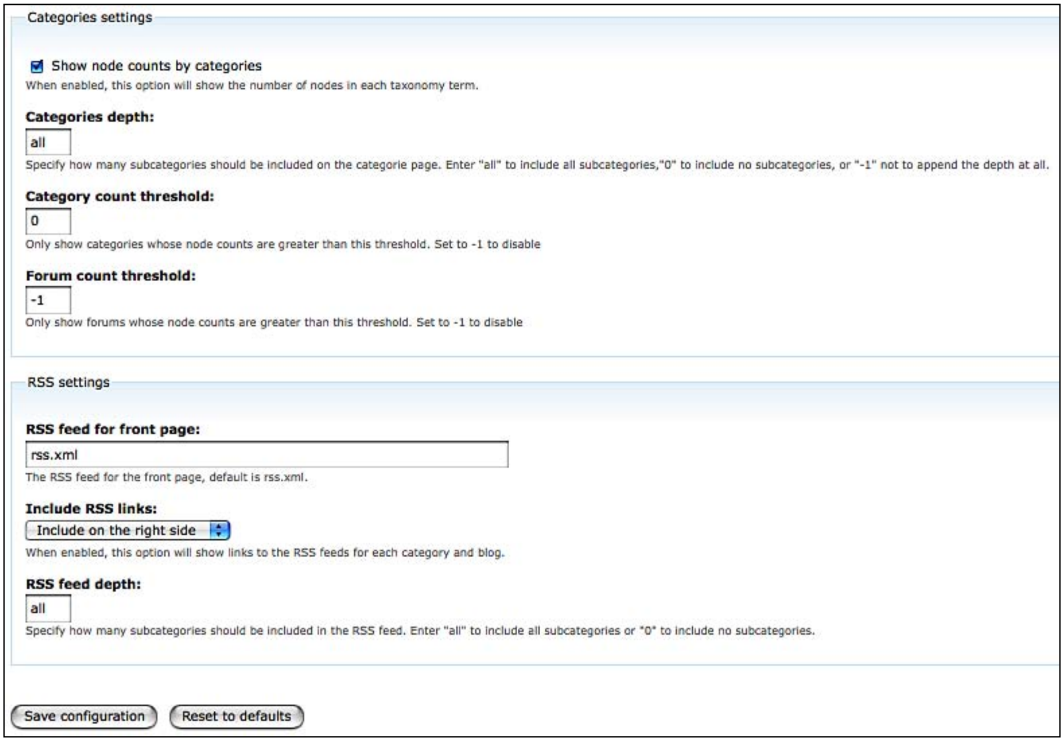
To install it, browse to `http://drupal.org/project/site_map`, and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Site configuration | Site map**. You'll see a screen similar to the following:



In the **Site map message** field, write a message that explains what exactly it is, so that your visitor can immediately understand the scope of the Site map page. In the **Site map content** pane, choose from the following the content of your store that will be displayed in the site map:

- **Show front page**: If you want to include the front page in the site map, then check this option.
- **Show active blog authors**: This option is useful only if your site contains blogs from multiple authors. In this case, the site map will show the 10 most active blog authors.
- **Books to include in the site map**: In Drupal, a book is a set of pages tied together in a hierarchical sequence, perhaps with chapters, sections, subsections, and so on. You can use books for manuals, site resource guides, Frequently Asked Questions (FAQs), or whatever you like. If your site contains books, you can select one or more of them from here to include in the site map.
- **Menus to include in the site map**: You don't want to expose all of your menus to the site map, because maybe some of them have links to administration pages or protected pages. Select at least **Primary links**, but skip **Navigation**.
- **Show FAQ content**: The FAQ module creates a page full of Frequently Asked Questions together with the answers to them. It's not a bad idea to include them in the site map, so enable this option.
- **Categories to include in the site map**: Here, there is a list that shows all the taxonomy terms of your site and you select which one of them you want to show in your site map. In most cases, including all the categories is a good idea.
- **Do not include site map CSS file**: Turn this option off, because we don't want to load the included CSS file.

The **Categories settings** pane gives you the option to define exactly how the categories will be shown in the site map.



Here, you will get the following options:

- **Show node counts by categories**: If you turn this option on, Site map will display a number next to every category, which shows the number of the nodes that it includes. It's a useful feature, so turn this on.
- **Categories depth**: There is no reason to leave any category out of the site map, so leave this option to **all**.

- **Category count threshold**: If you enter a number here, only the categories whose node counts are greater than this threshold will be included. If you enter **-1**, this feature is disabled. Enter **0** to include only categories that contain nodes.
- **Forum count threshold**: If you enter a number here, only the forums whose node counts are greater than this threshold will be included. If you enter **-1**, this feature is disabled. Enter **0** to include only forums that contain nodes.

This module creates not only a page, but also an RSS feed for your site. In the **RSS settings** pane, you can define all the details of this feed.

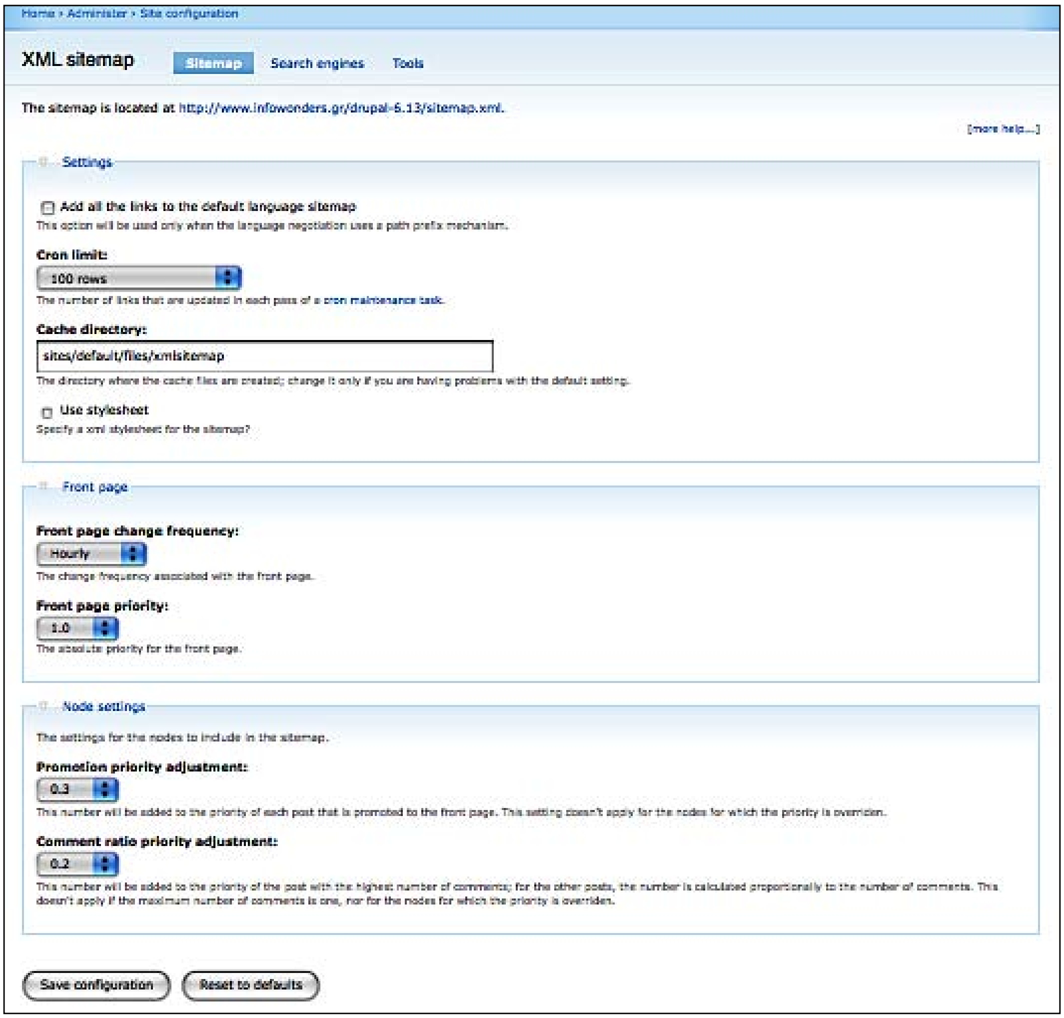
- **RSS feed for front page**: This is the name of the file for the RSS feed of the front page. Keeping the default name is a good idea.
- **Include RSS links**: If you turn this option on, a little icon will be displayed in every category and blog. By clicking on it visitors to the site can subscribe to this category and blog.
- **RSS feed depth**: We want to include all the subcategories in the site map, so select **all**.

Finally, click on **Save configuration** to store the settings. That's it, your page map is created; take a look at it by browsing to `/sitemap`. Don't forget to go to **Home | User management | Permissions** and set the permissions for this page.

XML sitemap module

XML sitemap is a protocol for site map creation. It was first introduced by Google, but it's now been adopted by most big search engines, such as Yahoo, Bing, and Ask. It creates a list of all the pages of your site, together with some useful metadata, such as the date that a page was last updated. Search engine crawlers can read this file, thus helping to submit as many pages of your website as possible. You can find a description of the XML schema for the sitemap protocol at `http://sitemaps.org/protocol.php`. XML sitemap module automatically creates a site map compatible with the above protocol.

To install it, browse to `http://drupal.org/project/xmlsitemap` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. You have to enable not only the main module, but also all the other related modules: XML sitemap engines, XML sitemap node, XML sitemap taxonomy, and XML sitemap user. To configure it, go to **Home | Administer | Site configuration | XML sitemap**. You will see a screen similar to the following:



Here, the **Sitemap** tab contains all the settings for the XML sitemap file.

Within this tab, first there is the **Settings** pane, with general settings about this file, which are described as follows:

- **Add all the links to the default language sitemap**: If your site is multilingual, this option will allow you to add all the links, including those of the other active languages, to the site map of the default one. Usually, you want this option turned off.
- **Cron limit**: Cron is a small application that executes specified commands. It's necessary for the maintenance of your site. Here, you can define how many links will be scanned each time that cron runs. The default value is 100 rows, but if your store has thousands of products, you can set this value higher.
- **Cache directory**: The directory where the cache files are created. Leave the default directory.
- **Use stylesheet**: Turn this on if you want to specify an XML stylesheet for the site map.

The **Front page** pane has two options:

- **Front page change frequency**: If the content of your home page changes frequently, you want search engines to visit it more often. So the default value, **hourly**, ensures that the most recent content of your home page will be displayed in search engines at any given time.
- **Front page priority**: Priority declares the importance of a page compared to the other pages of your website. The front page is usually the most important page of your website, so leave this set to the highest possible value, **1.0**.

The **Node settings** pane contains general settings for the nodes of the site map.

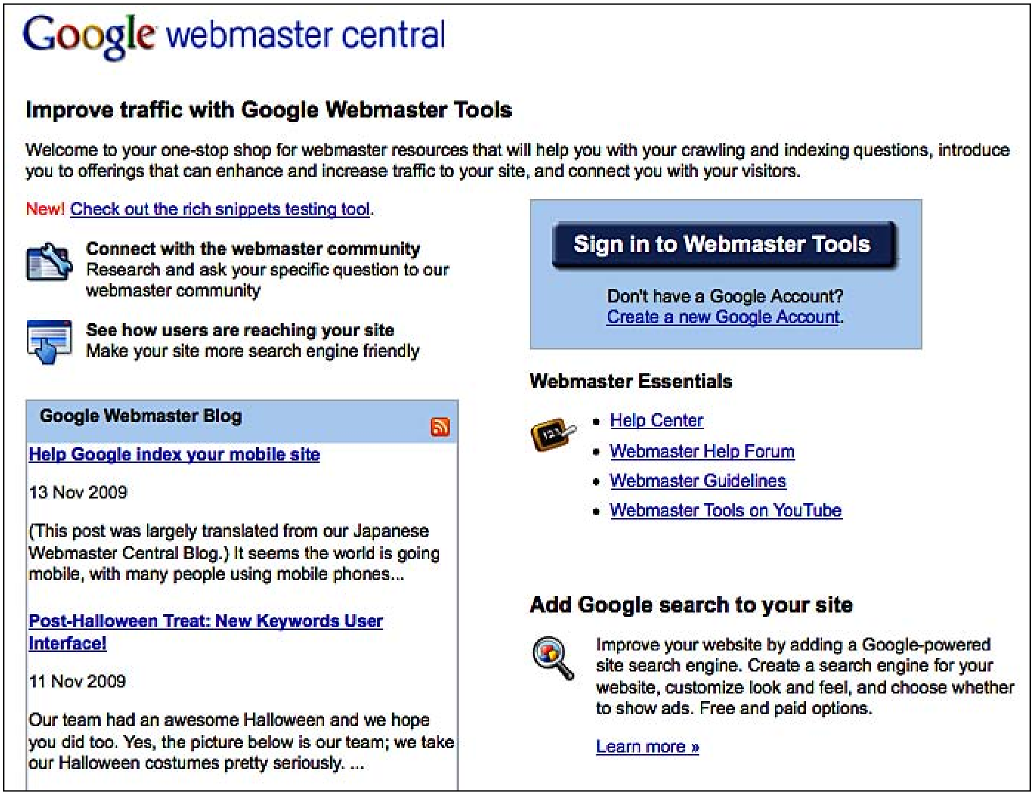
- **Promotion priority adjustment**: This field defines the priority value that will be given to any node that is promoted to the first page. Leave the default value.
- **Comment ratio priority adjustment**: This field defines the priority value that will be given to the node with the most comments. Leave the default value.

Click on **Save configuration** to store the settings. The XML site map file is now created.

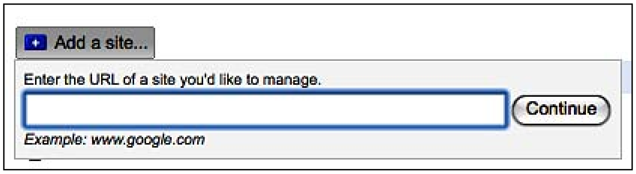
Submit your website to Google

Webmaster Tools is a free platform offered by Google. It contains valuable resources for search engine optimizations, useful tutorials, and some interesting tools that help you to submit your XML sitemap to Google and optimize it.

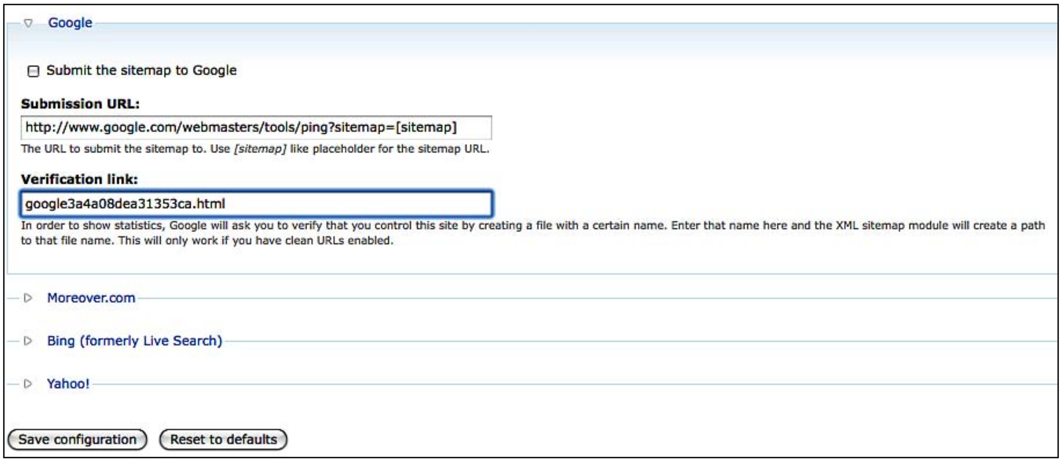
1. You can find Google Webmaster Tools at `http://www.google.com/webmasters/`. If you already have a Google username and password, you can sign in using them. If not, click on **Create a new Google Account** and create a new one.



2. On the home page you can see a list of the websites that you've already added to Google. If it is your first visit here, this list is empty. Click on **Add a site**, enter the URL of your website, and click on **Continue**.



3. Now, to verify that you are the owner of this website, Google suggests two ways: the first one is to insert a specific meta tag into the <head> section of your home page, and the second one is to upload a specific file to the root folder of your website. We are going to use the second method, but we don't need to upload the actual file, because the XML sitemap module will help us with it.
4. Return to your website, go to the XML sitemap settings page, and click on the **Search engines** tab. Open the collapsed pane named **Google**, and find the verification link field. Enter the name of the file that Google gave you (`google3a4a08dea31353ca.html`) and click on **Save configuration**.



5. Go back to the Google Webmaster Tools page and click on **Verify**. You'll receive a message that your site is verified and you can proceed.
6. Our last job is to submit our site map to Google. From the left menu, select **Site configuration | Sitemaps**.

7. Click on **Submit a Sitemap** and write `sitemap.xml` in the blank field. Click on **Submit Sitemap**. If you check the status of the submitted site map, you'll see the icon of a little clock. That means that Google is still reading the file. Come back in a few hours, because it takes a little time, especially when you submit your site map for the first time.

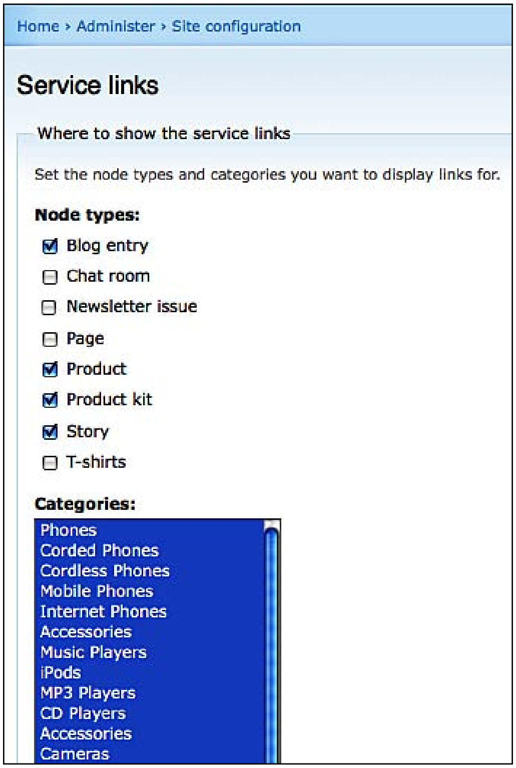
When the file is finally processed, you will see its status as completed. Now, you can explore and use all the sections of Google Webmaster Tools, such as **Your site on the web** or **Diagnostics**.

Internet marketing

A website without proper marketing is doomed to fail. No matter how high is the quality of your products or how low are your prices, if your potential clients don't know that you exist, your store will have very few visitors. In this section, we'll show how to implement some simple, but very important marketing strategies, aimed at promoting your store and attracting new and targeted visitors.

Service links module

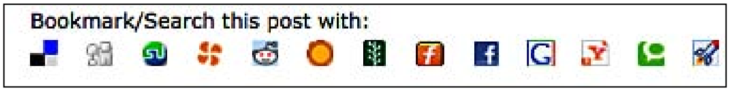
This module allows you to share your content in the most popular networking sites, such as Facebook, StumbleUpon, Digg, Delicious, Google, Yahoo, and so on. People spend most of their time at these sites, so it's a great opportunity to inform them about your company and your products. To install it, browse to `http://drupal.org/project/service_links`, and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Site configuration | Service links**.



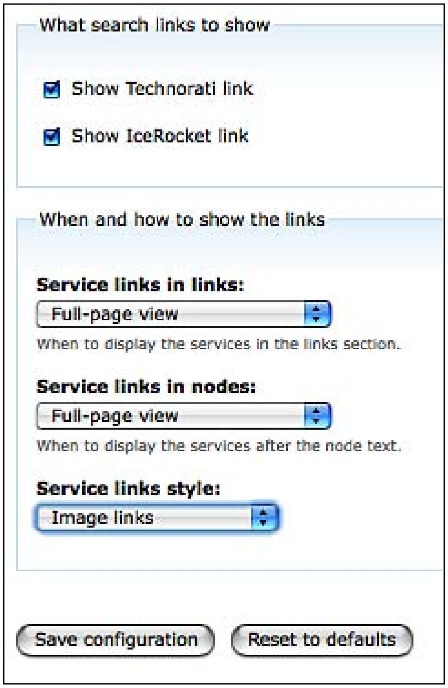
In the first section of the settings page, you have to define **Where to show the service links**, using the following two options:

- **Node types**: Select the node types for which you want to include service links. You don't have to include every node type, so select only **Blog entry**, **Product**, **Product kit**, and **Story**.
- **Categories**: Select the categories that you want to include service links. If there is no special reason to exclude a particular category, select all of them.

In the second section, you select **What bookmarks links to show**. There is a list of all the available sites, so select the ones that fit most to your company's profile.



In the third section, you select **What service links to show**. There are only two available options here, Technorati link and IceRocket link, so decide which you want to include.



In the last section, you define **When and how to show the links**, using the following settings:

- **Service links in links**: You define when to show service links in the links section. The available options are **Disabled, Teaser view, Full-page view,** and **Teasers and full-page view**.
- **Service links in nodes**: You define when to show service links in the nodes section. The available options are **Disabled, Teaser view, Full-page view,** and **Teasers and full-page view**.
- **Service links style**: Here you decide if you want to include **text links, image links,** or **image and text links**. It's up to you to decide which style is better for your website.

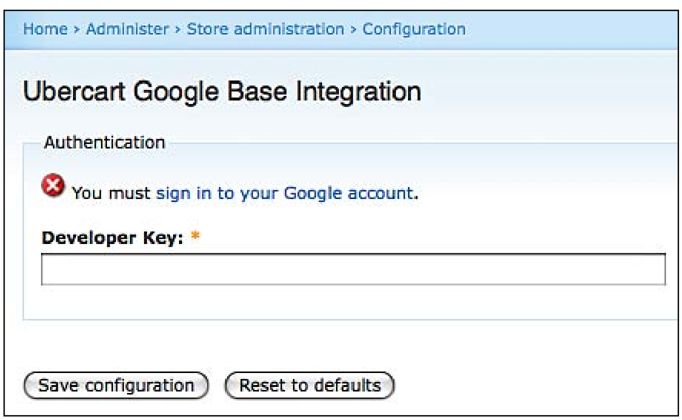
Finally, click on **Save configuration** to store the settings.

Now, when a person visits a product page from your store, he or she will see a new region with icons. These are the service links that allow the sharing of your content.

Google Base integration

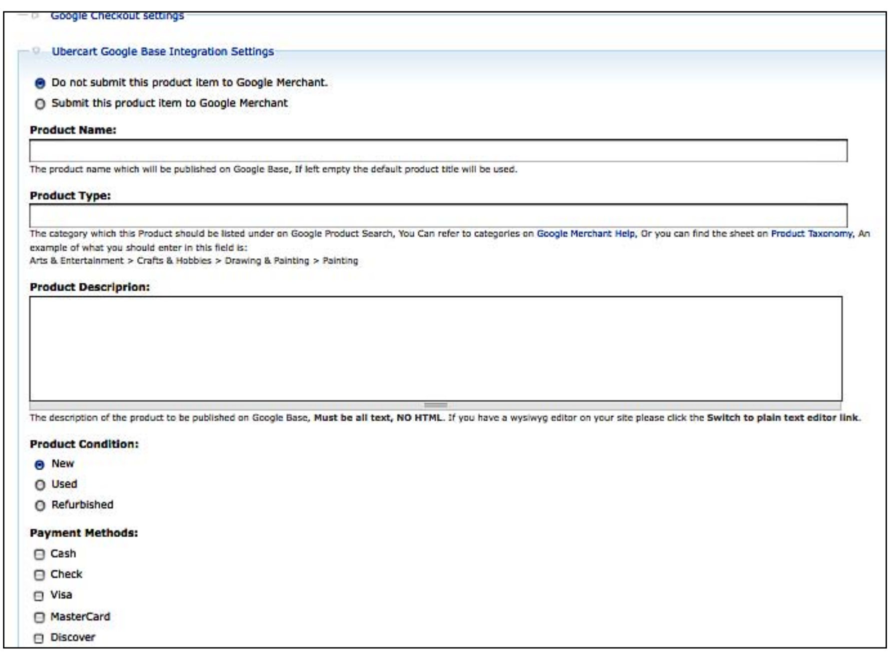
Google product search (`http://www.google.com/products`) is a price comparison service, created by Google. It doesn't actually sell the products, but when a visitor searches for a specific product, it shows a comparison table with photos, prices, and ratings. There are links to the retailers' websites, where people can actually buy the products. So, you understand that it's a great opportunity for you to include your products in it. Google offers an easy way to upload your products, called **Google Merchant Center** (formerly known as **Google Base**). We're going to use the Ubercart Google Base module, which allows you to complete the entire integration by following a few easy steps:

1. To install it, browse to `http://drupal.org/project/uc_gbase` and right after you download the module upload it and unzip it to your site's `/sites/all/modules` folder, then go to **Administer | Site building | Modules** to enable it.
2. To configure it, go to **Home | Administer | Store administration | Configuration | Ubercart Google Base Integration**. Click on **sign in to your Google account**. After you log in to Google using your username and password, Google Base asks for permission to create an account. Click on **Grant access**. You'll receive a conformation message that **you have successfully signed in to your Google account**.



3. Go to `http://code.google.com/apis/base/signup.html` and sign up for a Google API key. Copy this key, paste it on the settings page of your site, and click on **Save configuration**.

4. Now, when you submit a new product, there is a new section named **Ubercart Google Base Integration Settings**, with the following options:
    i. First, for every product you select if you want to submit it to Google Merchant or not.
    ii. **Product Name**: If you leave this empty, the default name will be submitted. If you want a different name, type it in this field.
    iii. **Product Type**: Enter the Google Product Search Category under which this product will be listed. If you are not sure what to enter here, consult the Google Merchant Center Help (`http://www.google.com/support/merchants/bin/answer.py?hl=en&answer=160081`).
    iv. **Product Description**: Enter the product description that will be displayed in Google Product Search, in plain text format.
    v. **Product Condition**: Choose if the product is **New**, **Used**, or **Refurbished**.
    vi. **Payment Methods**: Select a payment method for this product. The available methods are **Cash**, **Check**, **Visa**, **MasterCard**, and **Discover**.
    vii. When you click on **Save** to store your new product, it will automatically be submitted to Google Product Search, and it will be visible to millions of potential clients.



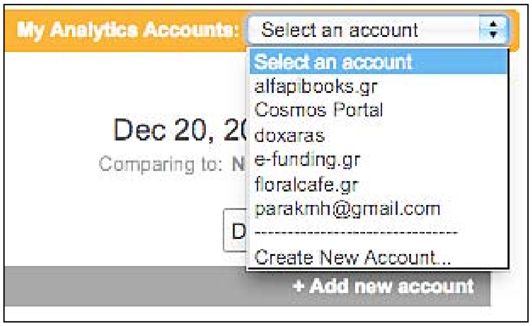
Google Analytics and goals tracking

Google Analytics is currently the number-one tool for monitoring online marketing, and is used in a cross-domain way. For instance, Google Analytics can provide a very clear answer to the question "What is the percentage of my clients that view a specific product, and eventually end up buying it?" In general, targeted analytics are ideal for online shops, as they provide valuable information used in various circumstances such as:

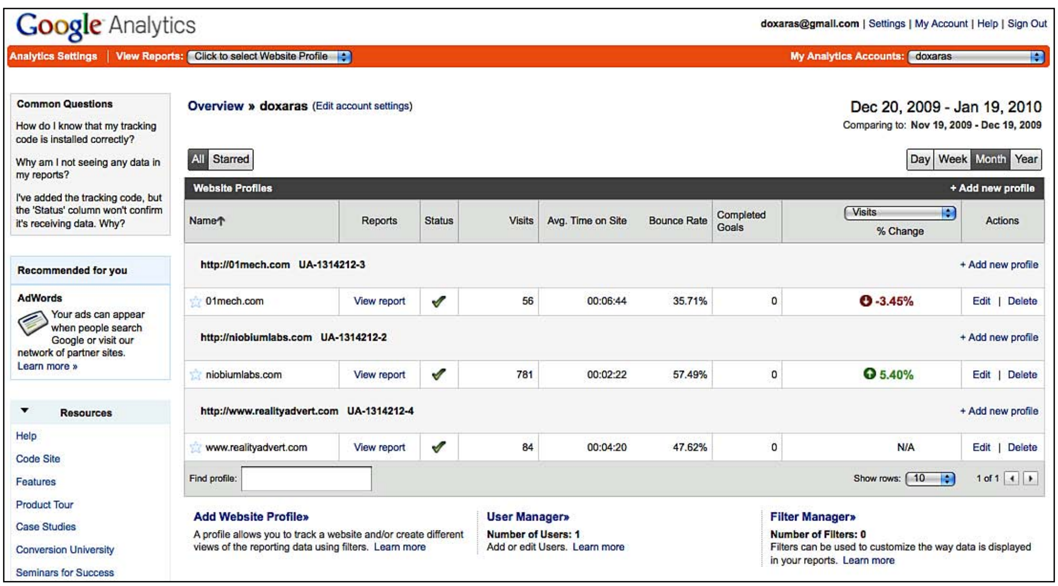
- Monitoring online campaigns
- Conversions and impressions statistics
- Tracking user behavior
- Product sales

First of all you need an analytics account to associate with your site. To do so, follow these steps for creating and configuring your account in Drupal:

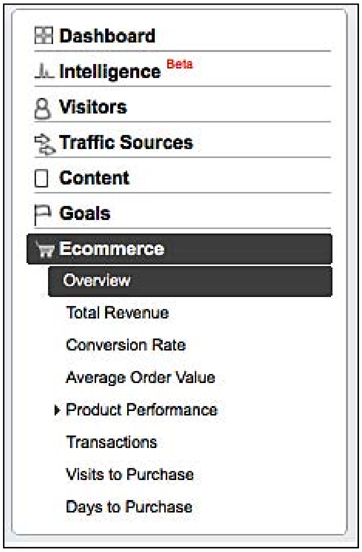
1. You need to have Google Analytics Drupal module installed. If you have not done so, refer to *Chapter 2, Installation of Drupal and Ubercart*, for details on the installation procedure.
2. After that, you need to create a Google Analytics account and link it to your Drupal site. To do this, navigate to `http://www.google.com/analytics` and provide your credentials to access the service. After you log in, you need to create a new site account. Click on the **Create New Account...** link on the top right of the page.



3. This will trigger a wizard that asks you for your site name (provide the full canonical name `www.myshopname.com`), a name to this account, and to set up the time zone. Keep in mind that if you already have a Google Analytics account, you should use the existing one because you gain more if you are an individual administrator of multiple analytics accounts with high traffic numbers.
4. Take the unique account number and navigate to **Administer | Site configuration | Google Analytics**, fill the appropriate field with this, and click on **Save configuration**.
5. The introductory analytics screen has some interesting areas that you should pay attention to. You can inspect the sites that you have configured for this particular account. In addition, you can manage users and grant permissions. A very common practice when it comes to Internet marketing campaigns, is to add your marketing consultant account to your analytics site to provide permissions for that person to access statistics, in order to design the marketing strategy. Finally, there is also an indication on the status of your site regarding the Google bot accessing it. For instance, if you have broken links and misconfigured analytics, you will see alerts in the status column. From the account view, you can access more detailed statistics by clicking on the **View report** link.



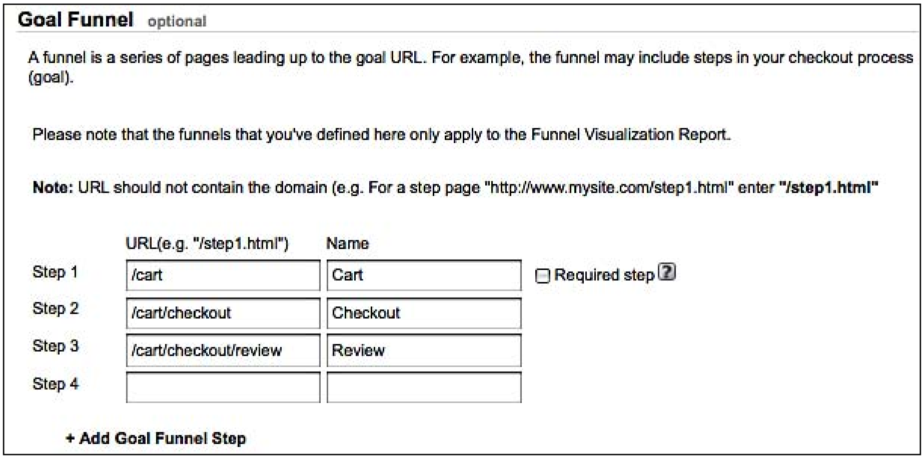
Another aspect of functionality that Google Analytics provides is the goal and target definition section. To access and define your goals, click on the **Edit** link next to the account you have just created. You will see the site profile page. It is very important in order to exploit the ready-made analytics functions that Google provides for online stores to define your site as an online store. To do so, click on the **Edit** link in the **Main Website Profile Information** bar, and then click on the option **Yes, an E-Commerce Site**. There you go! Now you will be able to have access to the e-commerce menu that Google Analytics provides.



As everything is set up now, we need to start defining our goals. Navigate to your site profile page and click on the **Add goal** link. The goal we will define follows Ubercart's checkout workflow and is an important factor in defining how many users initiate the checkout process after viewing the products. In general, this goal is the same for all Ubercart installations with the only difference being that if you do not have your portal in the root folder, it just adds the extra path.

1. Click on the **Add goal** link in the site profile page.
2. Fill in the name **Purchase** in the name fields, as we will describe a purchase workflow.
3. Select **URL Destination** as **Goal type**.

4. At the goal details section, choose head match and provide `[your_drupal_path]/cart/checkout/complete` as **Goal URL**.
5. Click on the **+Add Goal Funnel Step** link and enter the states as shown in the following screenshot:



6. Click on the **Save Goal** button.

After creating your target goal, you can monitor the conversions of your end users on a day-to-day basis and check instantly if your marketing campaign is fulfilling its purpose. Keep in mind that you are in the direct marketing market, so you should have a very solid idea on what is the revenue generated for every dollar you spend on marketing. In this objective, Google Analytics is your number-one alloy.

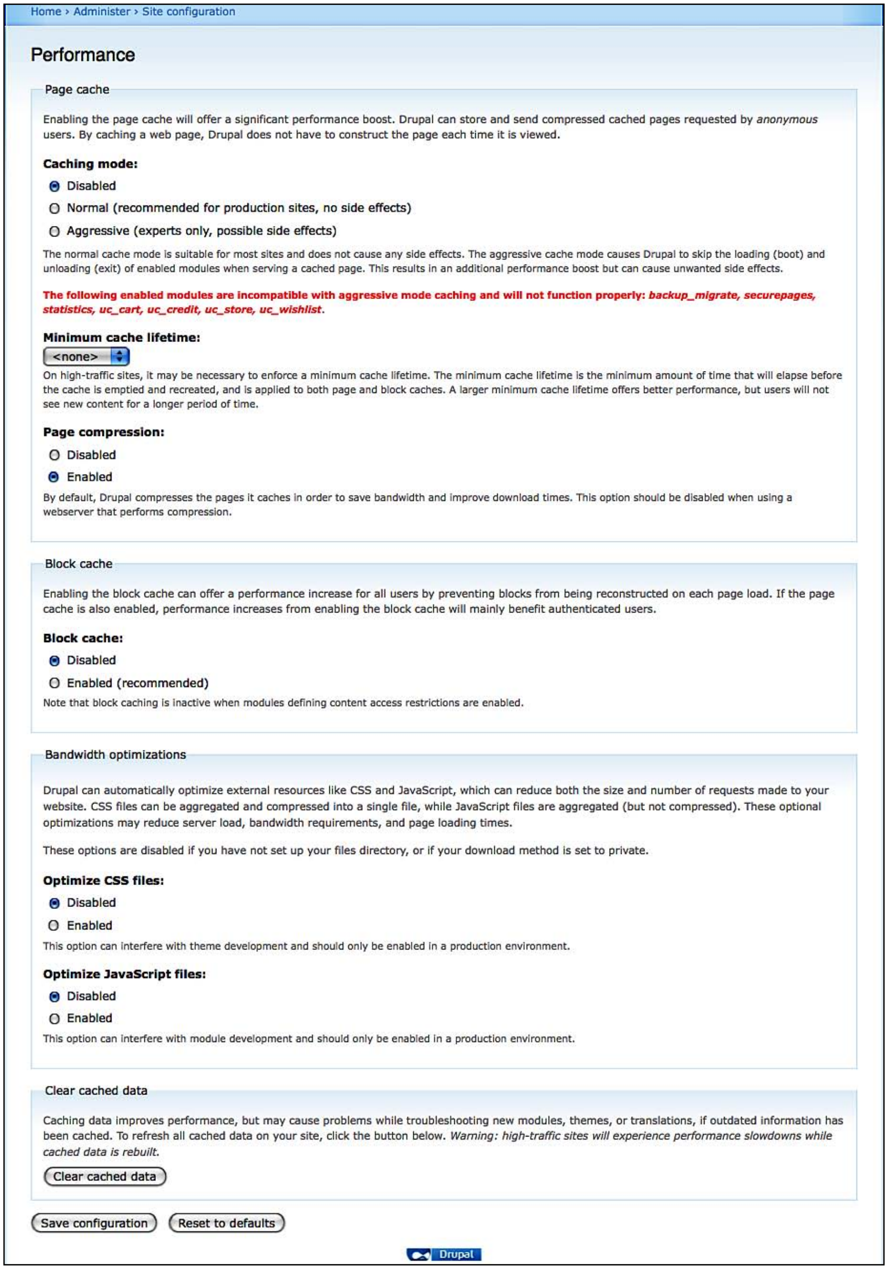
Another thing you should keep in mind is to be patient, as there is a particularly large aging factor in good Internet marketing campaigns, and set up goals for the next three, six, and twelve months and constantly track them in order to see if your marketing plan is solid or need modifications.

Improving performance

For the first weeks after you launch your new store, you don't have to worry about your site performance, because you will have few visitors. As the days go by, you'll see that traffic increases. Maybe an advertisement or an article in the local newspaper will cause a peak in you site's visitors—are you sure that your store is prepared to handle it? In this section, you will see how to improve the performance of your website and make it work faster using less resources. Keep in mind though, that the selected theme may also affect the performance, if it has lots of layout images, JavaScript, or effects.

Performance settings

You can improve your site's performance just by altering the cache settings and by compressing important files. Go to **Home | Administer | Site configuration | Performance**.



On this screen, the following options are available:

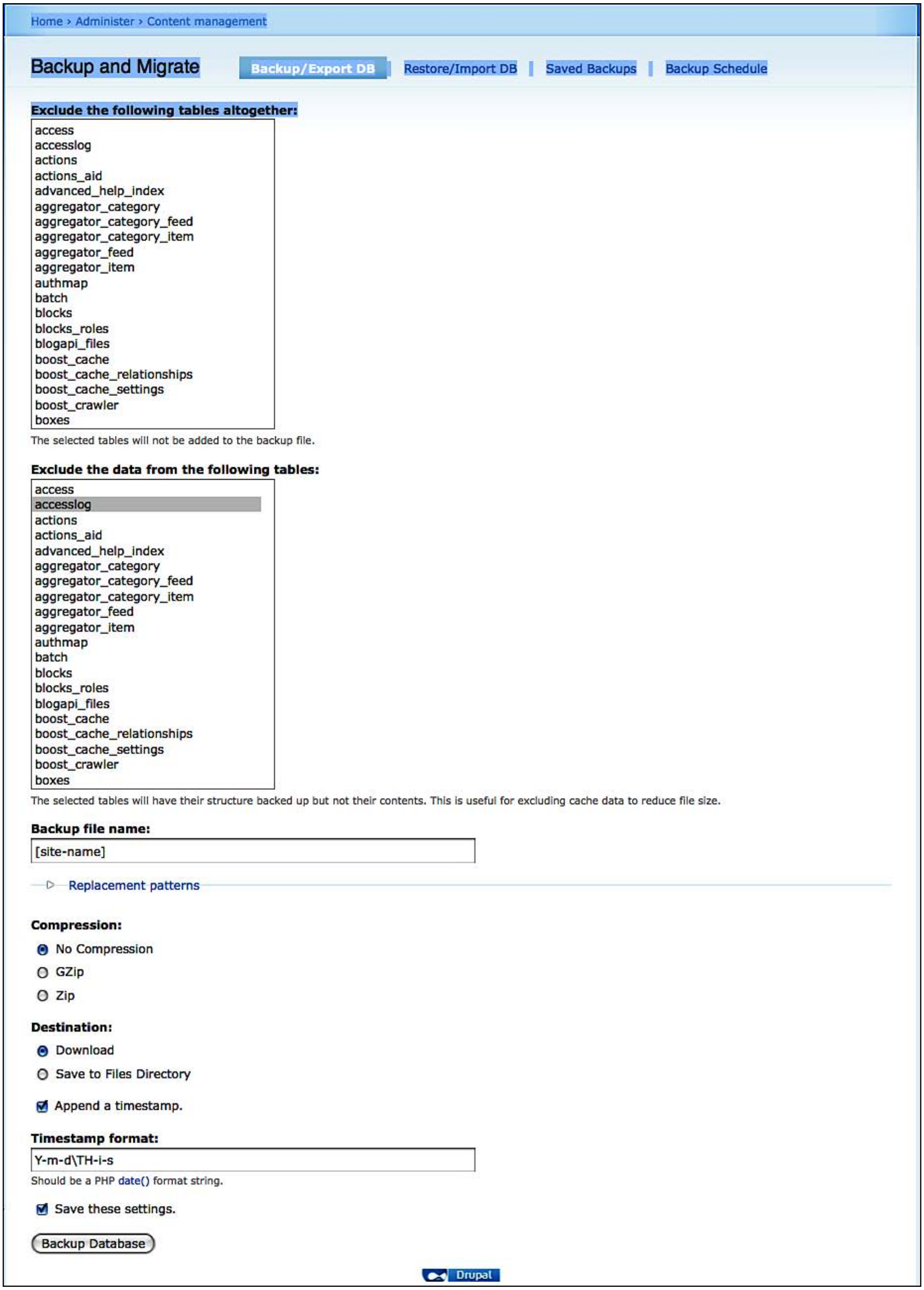
- **Caching mode**: When you enable caching, Drupal stores the latest version of your page and serves this to anonymous users. Using this method, it doesn't have to recreate the page on every request, saving precious system resources this way. There are three caching modes: **Disabled**, **Normal**, and **Aggressive**. For your store, normal caching mode is fine.
- **Minimum cache lifetime**: Using this option, you can set a minimum time before the cache is emptied and recreated. If you enter new content often, keep this lifetime low.
- **Page compression**: If this option is enabled, Drupal compresses the pages before caching.
- **Block cache**: If this option is enabled, Drupal caches not only pages, but also blocks.
- **Optimize CSS files**: If this option is enabled, Drupal compresses CSS files and reduces their size and number.
- **Optimize JavaScript files**: If this option is enabled, Drupal compresses JavaScript files and reduces their size and number.

Enhancing store security

Drupal is one of the safest Content Management Systems; that's why it is used by some of the biggest companies and organizations. In this section, we'll show you some simple ways to make your site safer, protect it from spams, and back up your data.

The Backup and Migrate module

The **Backup and Migrate** module simplifies the task of backing up and restoring your database. To install it, browse to `http://drupal.org/project/backup_migrate` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Content management | Backup and Migrate**.

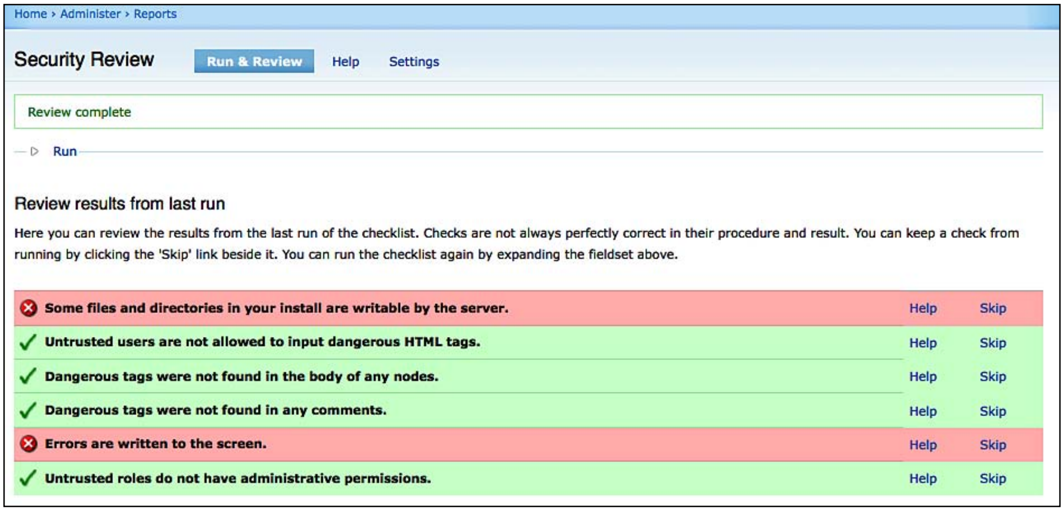

Home › Administer › Content management
Backup and Migrate
Backup/Export DB
Restore/Import DB
Saved Backups
Backup Schedule
Exclude the following tables altogether:
access
accesslog
actions
actions_aid
advanced_help_index
aggregator_category
aggregator_category_feed
aggregator_category_item
aggregator_feed
aggregator_item
authmap
batch
blocks
blocks_roles
blogapi_files
boost_cache
boost_cache_relationships
boost_cache_settings
boost_crawler
boxes
The selected tables will not be added to the backup file.
Exclude the data from the following tables:
access
accesslog
actions
actions_aid
advanced_help_index
aggregator_category
aggregator_category_feed
aggregator_category_item
aggregator_feed
aggregator_item
authmap
batch
blocks
blocks_roles
blogapi_files
boost_cache
boost_cache_relationships
boost_cache_settings
boost_crawler
boxes
The selected tables will have their structure backed up but not their contents. This is useful for excluding cache data to reduce file size.
Backup file name:
[site-name]
Replacement patterns
Compression:
No Compression
GZip
Zip
Destination:
Download
Save to Files Directory
Append a timestamp.
Timestamp format:
Y-m-d\TH-i-s
Should be a PHP date() format string.
Save these settings.
Backup Database
Drupal

In the first tab, you can manually create a backup of your database. You select which tables or table data you want to exclude, the name of the backup file, the compression type, the destination folder, and the time stamp format. Click on **Backup Database** to create the backup.

Click on the **Backup Schedule** tab to create a schedule for your backups. You need to enter the number of hours between backups and the number of backups to keep, and click on **Save configuration**.

Security Review module

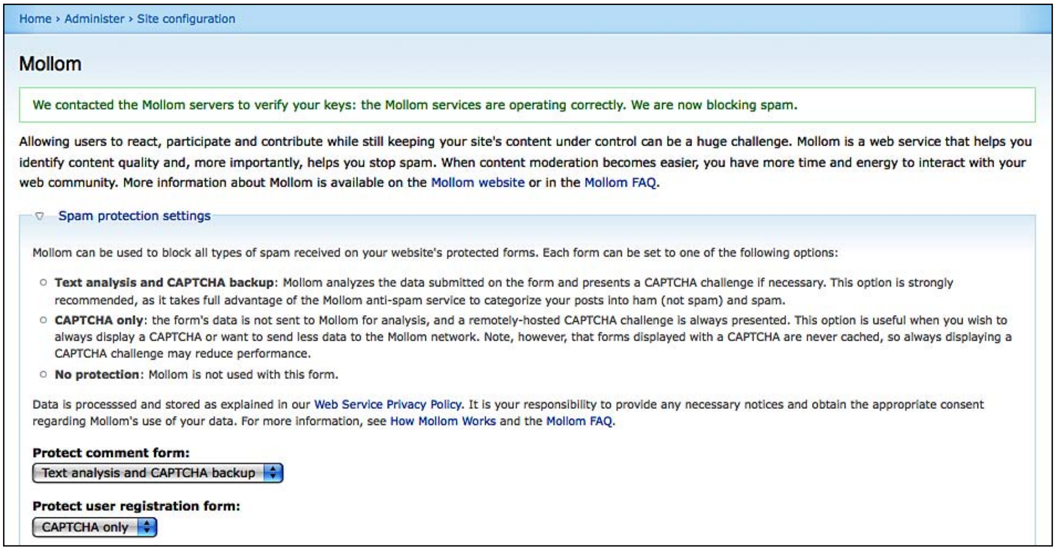
This is a simple module that checks your website and gives you a report about existing vulnerabilities. To install it, browse to `http://drupal.org/project/security_review` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder, and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Reports | Security Review**. Click on **Run checklist**. You will see a list of all the checks made by the module. The items marked with red color are possible security holes and should be corrected immediately.



Mollom module

Mollom is one of the best anti-spam tools available. It protects your site from several threats, such as spams in the form of comments, contacts, or password requests, or fake user accounts registration. The Mollom module automatically connects your site with Mollom, just by following a few easy steps.

1. To install it, browse to `http://drupal.org/project/mollom` and right after you download the module, upload it and unzip it to your site's `/sites/all/modules` folder and go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Site configuration | Mollom**.
2. Go to the Mollom website (`http://www.mollom.com`), and click on **Create account**.
3. Enter username, e-mail address, and personal information, and click on **Create new account**. Soon, you will receive a message with your username and password.
4. Log in to your new account and click on **Site manager**.
5. Click on **Add new site**. Select **Get Mollom Free**. Enter the required information for your site and click on **Next**. Review your settings and click on **Complete subscription**.
6. Now you can see your site in the **Site manager** page. Click on **view keys**. Copy the **Public key** and **Private key** values.
7. Return to the Mollom settings page of your website and paste the **Public key** and **Private key** values. Click on **Save configuration**.
8. You will see a message at the top of your page that Mollom services are operating correctly and that they are now blocking spam.
9. At the bottom of the page there are settings about every type of protection that Mollom offers. The default settings work fine in most cases, so just click on **Save configuration**.



Summary

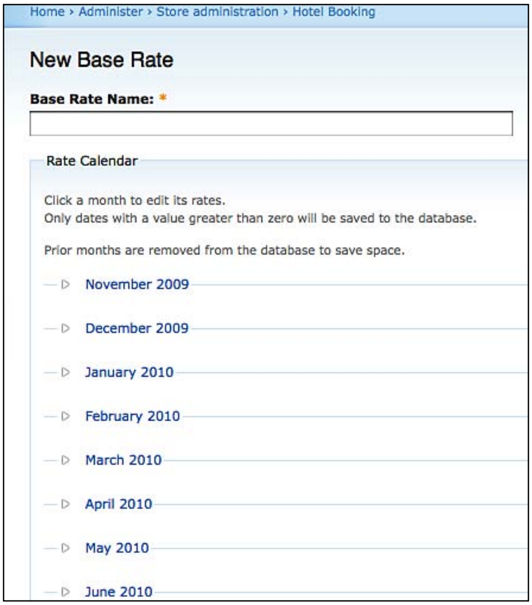
In this final chapter, we showed you how to optimize your store for search engines, how to implement some easy marketing techniques, and how to make your site faster and more secure. You have built a superb store, using a state-of-the-art CMS combined with an amazing store application. Of course, you'll have to keep improving your store, adding more products, and optimizing your content, but don't forget to be proud of your website, love your customers, and enjoy the whole process!

A

Hotel Bookings System for Ubercart

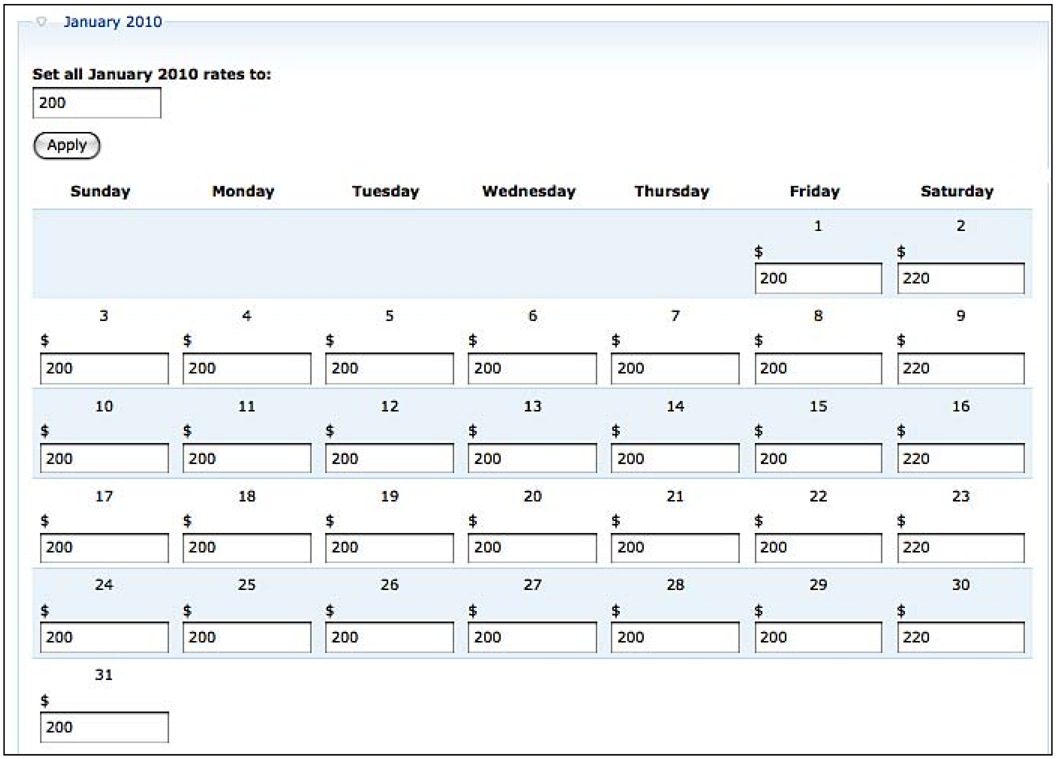
Ubercart can now be used for hotel bookings, thanks to the **Hotel Booking System for Ubercart**. To install it, browse to `http://drupal.org/project/uc_hotel`. After you download the module unzip it, and upload it to your site's `/sites/all/modules` folder. Now go to **Administer | Site building | Modules** to enable it. To configure it, go to **Home | Administer | Store administration | Hotel Booking**. Here you can set up all the options for your hotel using several available tabs as follows:

- **Base Rates**: Here, you can create and manage the base rates for your hotel. Let's create a new one. Click on **Add Base Rate**. A new page will open.



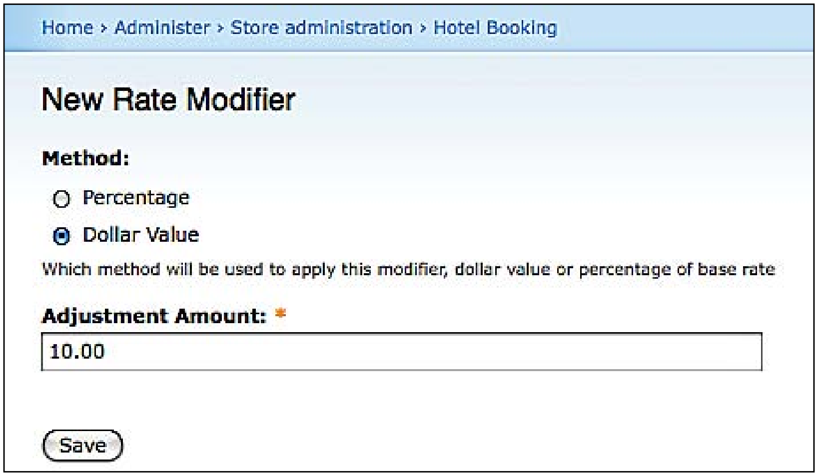
This page has the following fields, which you need to fill:

- ◦ **Base Rate Name:** Here, you can enter a descriptive name for this base rate.
- ◦ **Rate Calendar**: Using the rate calendar, you can set up the base rates for the next 12 months. Click on a month to open its calendar. If you want to set a single rate for all the days of this month, enter it in the **Set all month rates to** field and click on **Apply**, else enter a different value for each day of this month.



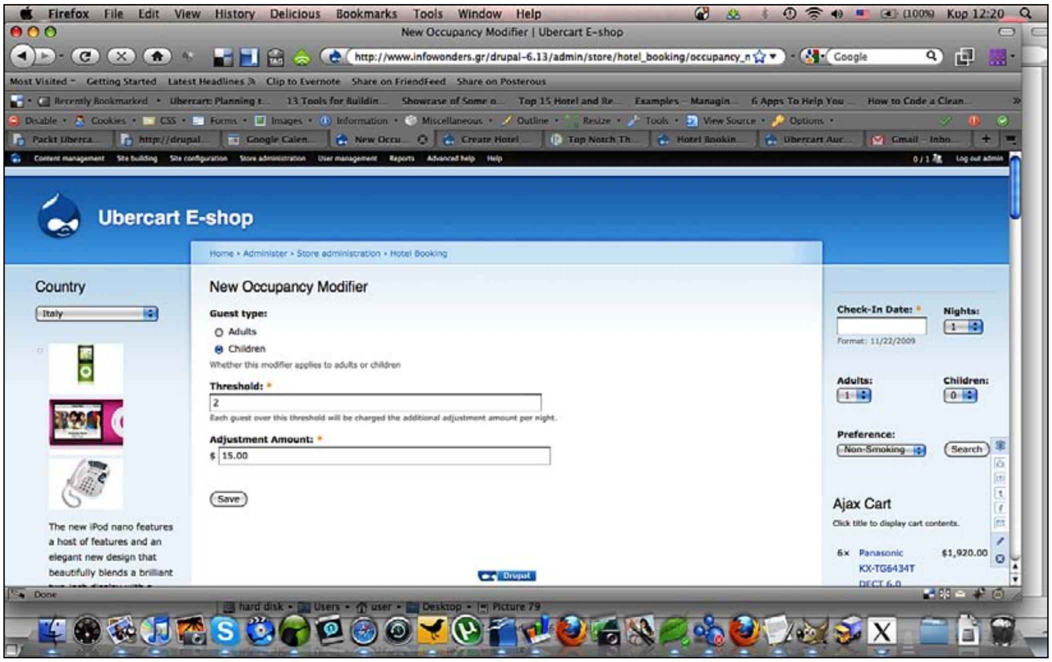
- ◦ Repeat this procedure for all the following months and click on **Save**.

- **Rate Modifiers**: By using the rate modifier, you can enter a price adjustment.
    - Click on **Add Rate Modifier** to create a new one.
    - Select whether your preferred method is **Percentage** or **Dollar Value**.
    - Enter an **Adjustment Amount**.
    - Click on **Save** to save the configuration.

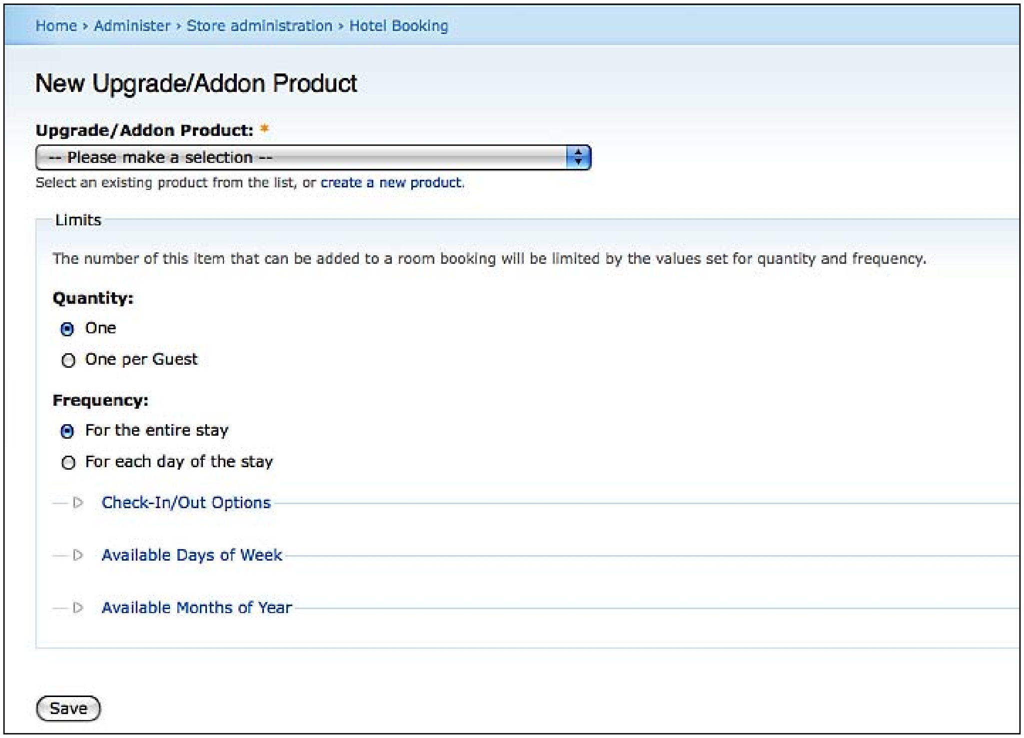


- **Occupancy Modifiers**: The occupancy modifiers alter the cost of each room, for each additional guest in a room over the **Threshold**. Now click on **Add Occupancy Modifier** to create a new one.
    - Select which **Guest type** this modifier applies to: **Adults**, **Children**, or **Occupants**.
    - Enter a guest **Threshold**. Only guests above this threshold will be charged with the adjustment amount.
    - Enter the **Adjustment Amount**.

- ° Click on **Save** to save the configuration.

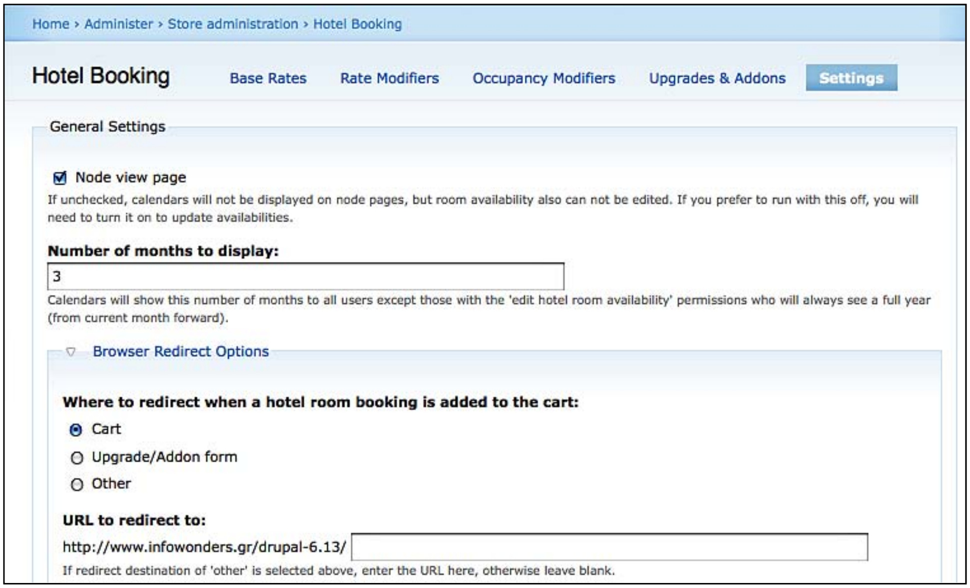


- **Upgrades & Addons**: Here, products can be added as upgrade/addon products for your rooms. Click on **Configure New Upgrade/Addon Product** to add a new one.
  - ° **Upgrade/Addon Product**: Select this from the drop-down menu of the products in your site.
  - ° **Quantity**: Select whether the quantity will be **One** (per room) or **One per Guest**.
  - ° **Frequency**: Select whether this product will be available **For the entire stay** or **For each day of the stay**.
  - ° **Check-In/Out Options**: If this product is available for each day of the stay, you can define whether it will be **Available on Check-In date** or **Available on Check-Out day**.
  - ° **Available Days of Week**: Select for which days of the week this upgrade/addon will be available.
  - ° **Available Months of Year**: Select for which months of the year this upgrade/addon will be available.
  - ° Click on **Save** to store it.



- **Settings**: In this section you can set up the general settings for the hotel booking system.
    - **Node view page**: With this option turned off, you will not be able to view calendars on node pages, or update room availabilities, so it's better to keep it turned on.
    - **Number of months to display**: Here, you can define the number of months that site visitors will see in their calendars.
    - **Browser Redirect Options**: Here you can select the page to which the user will be redirected, when a room is added to the cart. You can select between the **Cart** page, the **Upgrade/ Addon form**, or you can select **Other**, and enter a custom **URL to redirect to**.
    - **Upgrade/Addon Page settings**: If you are not satisfied with the default settings, enter a custom **Upgrade/Addon Page Title** and an **Upgrade/Addon Page Text**.
    - **Search Result Options**: In this section you can select the **Sort Order** of the search results. The available options are: **Lowest price rooms first** and **Highest price rooms first**.

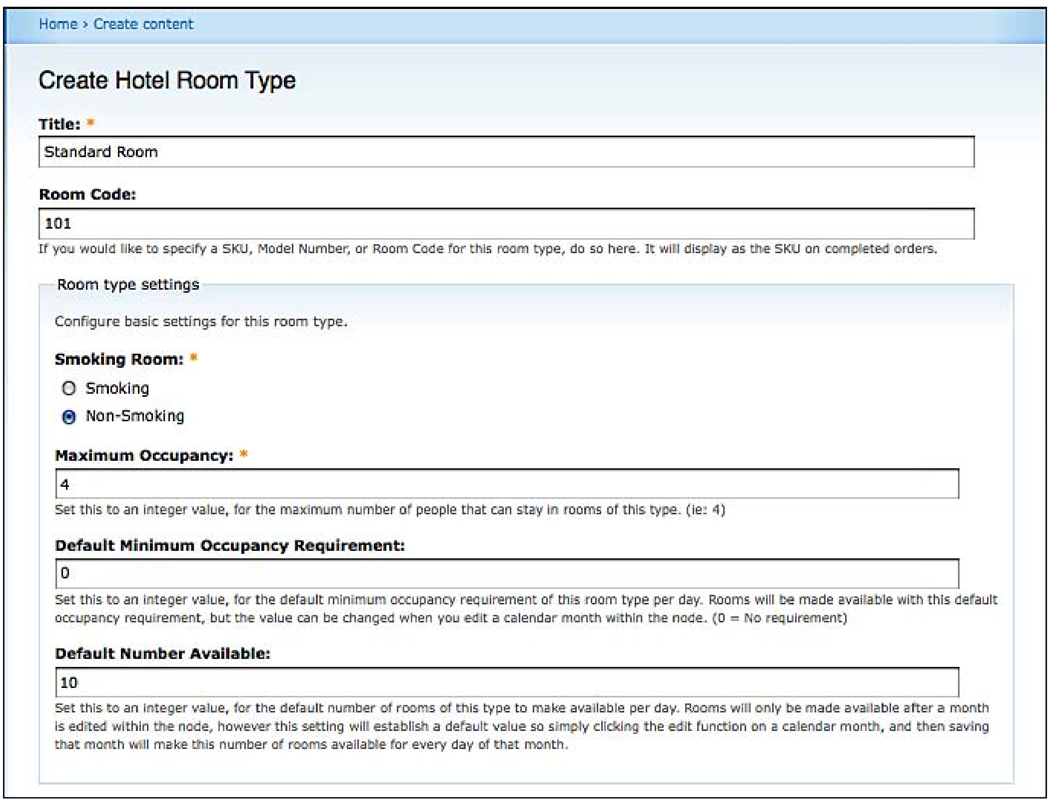
- **Expiry Settings**: If you haven't set an expiry date for the availability of your rooms, you can select when you want your available rooms to auto-expire, in relation to this date. The available options are: **Yesterday**, **Today**, and **Tomorrow**.
- **Teaser in cart**: Select **Yes** if you want to show the teaser text in the cart page.
- **Teaser in cart, checkout, and order panes**: Select **Yes** if you want to show the teaser text in cart, checkout, and order panes.
- **Reward Membership Prompt**: If you are using a reward membership system, enter a text in this field to prompt the user to enter his or her reward number during checkout. Leaving this field empty disables it.
- Finally, click on **Save configuration** to store these settings.



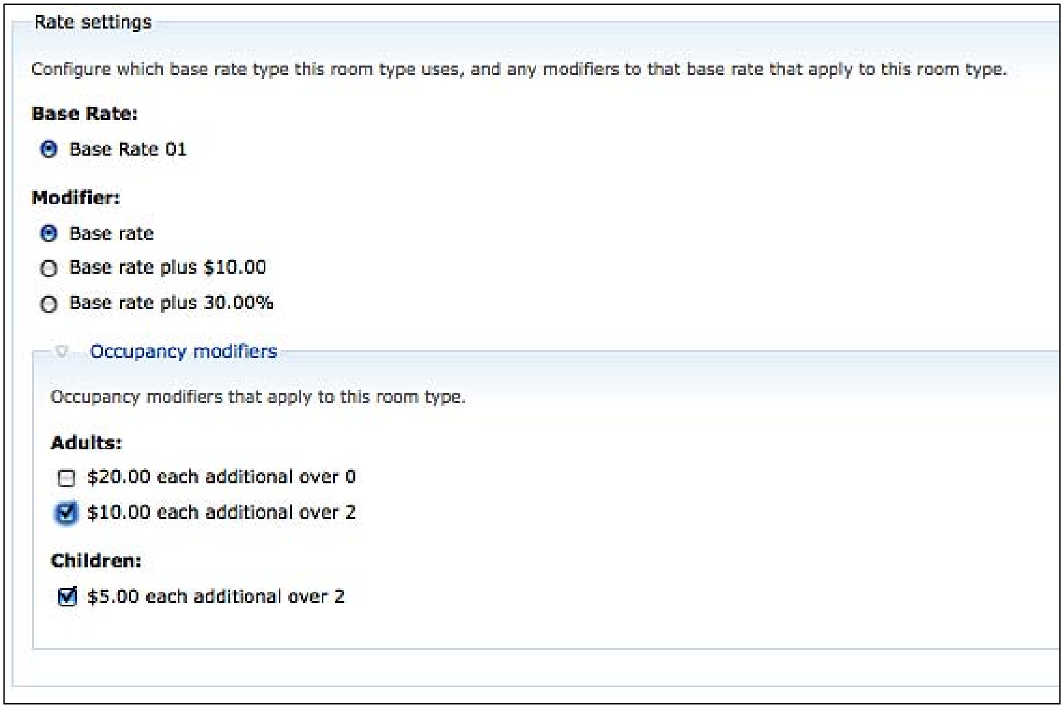
Creating Hotel Room type

In this module we will create a new content type, named Hotel Room. After you have finished with the settings, you have to create your rooms. Go to **Home | Administer | Create Content | Hotel Room Type**. You may notice that it is the default product insertion form, but contains some new fields, required to update a single product page to a hotel room page. Let's create a new room type to explore all the available options.

- **Title**: This is the name of the room type that the clients see when they want to book a room. Enter **Standard Room**.
- **Room Code**: The room code is optional, but is useful when you have many types of rooms. Enter **101**.
- **Smoking Room**: Select whether this type of room is **Smoking** or **Non-Smoking**.
- **Maximum Occupancy**: Here you can define the maximum people that can stay in this type of room. Enter **4**.
- **Default Minimum Occupancy Requirement**: If there is a minimum occupancy requirement, enter a number here. We don't need it, so just leave the default value, **0**.
- **Default Number Available**: If you enter a default number of available rooms here, you don't have to enter them manually for each calendar day. You just need to select a month and this value is entered for each day of this month. Let's say you have 10 rooms of this type, so enter **10** in this field.

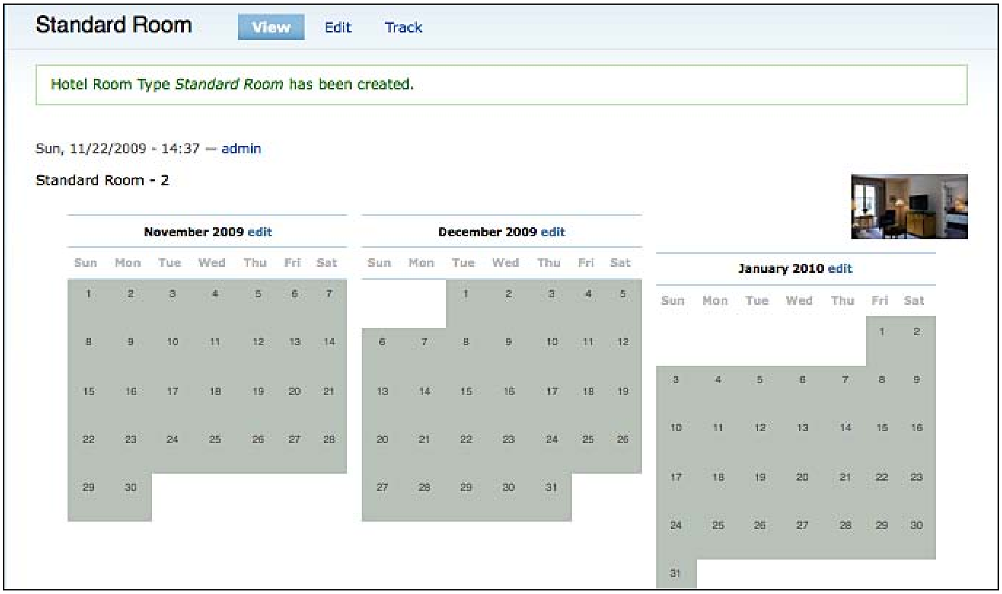


- **Base Rate**: Select one of the base rates that you defined in the settings section.
- **Modifier**: Select one of the price modifiers that you created in the settings section.
- **Occupancy modifiers**: Insert one of the occupancy that you created in the settings section.



- **Image**: Insert an image for this hotel type.
- **Body**: In this you can write a short or long description, so that the clients can understand exactly what this type of room offers.
- Click on **Save** to create it.

After the creation of the room, you will see a calendar for the next 12 months. This is used to enter the availability and the restrictions for each room on a daily basis.



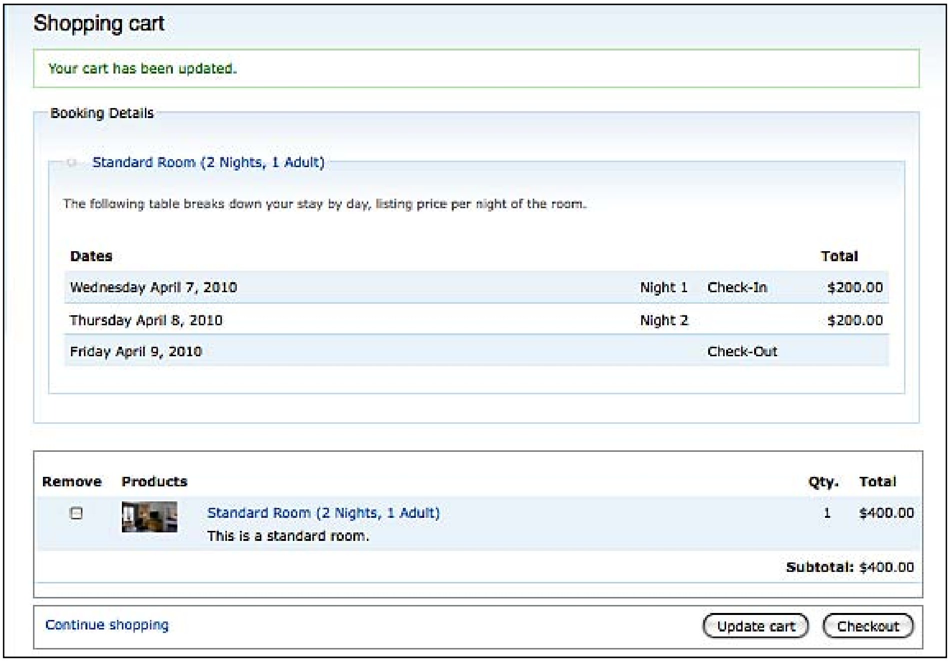
Click on **edit** to enter or to see the details for each day of the selected month. If you enter a value in the **Default Number Available** field, it will be populated for each day. If you click on the **Restrictions** link, you can select a required minimum stay, the required minimum guests, and whether these rooms are available for check-in or check-out. Click on **Submit** and repeat this procedure with the remaining months.

The Hotel Booking block

Now that you have created the room types and defined all the details, you have to enable the room booking functionality for your clients. Go to **Home | Administer | Site building | Blocks**, and enable the block named **Hotel Booking: Search Widget**. There, the visitor selects the **Check-In Date**, the nights of stay, the number of guests that are **Adults** and **Children**, and whether he or she wants a smoking or non-smoking room, and then clicks on **Search**.

During the next step, the visitor can see a list of all the available room types, with the prices for each day, the total cost, total cost per person, and the average cost per night, and can select one by clicking on **Book This Room**.

The client is then transferred to the shopping cart page, where he or she can see the booking details and make changes to them, or click on **Checkout** to continue with the payment.



In the final step, the visitor enters the billing information, payment details, and completes the order.

Summary

This is only a small example of how Ubercart can be used not only for selling products, but also for offering services. New modules are created every day that constantly add new features to it.

B
Modules Used in the Book

It would be impossible to complete the book and help you to create your e-shop without using the following great modules from some amazing developers. If you're using these modules in your project, you can help them by donating, writing a review at `http://www.drupalmodules.com`, helping to do translation, reporting bugs and issues, or just by saying thanks to the developers!

In this chapter, we list the modules you can download when working through the exercises for each of the chapters, and brief you a bit about them.

Modules used in Chapter 2

In Chapter 2, we download and install Drupal, Ubercart, and all of the basic modules required to support or enhance our Drupal Ubercart e-store features. Some of these are:

Token

Token allows other modules to place small bits of text into specific placeholders.

URL

`http://drupal.org/project/token`

Maintainers

Jeff Eaton, Greg Knaddison, Mike Ryan

Content Construcion Kit (CCK)

The Content Construction Kit is one of the most important modules in Drupal 6. It allows the creation of custom fields for every content type from the user interface.

URL
`http://drupal.org/project/cck`

Maintainers
Karen Stevenson, Marc Ferran, Moshe Weitzman, Yves Chedemois

Filefield
Filefield creates a field for CCK used to upload files.

URL
`http://drupal.org/project/filefield`

Maintainers
Nathan Haug, Andrew Morton, Jakob Petsovits

Imagefield
Imagefield creates a field for CCK used to upload images.

URL
`http://drupal.org/project/imagefield`

Maintainers
Nathan Haug, Andrew Morton

ImageAPI
ImageAPI cooperates with other modules such as imagecache, and supports image manipulation libraries such as GD2 and ImageMagick.

URL
`http://drupal.org/project/imageapi`

Maintainer
Andrew Morton

ImageCache

ImageCache gives you the ability to create predefined image dimensions and easily transform your images to one or more of these dimensions.

URL

`http://drupal.org/project/imagecache`

Maintainer

Andrew Morton

Thickbox

Thickbox brings the Thickbox jQuery add-on (`http://jquery.com/demo/thickbox/`) to Drupal. It shows texts, forms, images, or videos in a hybrid modal.

URL

`http://drupal.org/project/thickbox`

Maintainer

Fredrik Jonsson

Google Analytics

Google Analytics adds its tracking code to your store.

URL

`http://drupal.org/project/google_analytics`

Maintainers

Alexander Hass, Mike Carter

Views

If CCK facilitates the import of data, Views makes it easier to present them. It allows the creation of powerful queries without writing a single line of code.

URL

`http://drupal.org/project/views`

Maintainers

Earl Miles, Daniel Wehner, Derek Wright, Daniel F. Kudwien

Modules used in Chapter 4

In Chapter 4, we create categories and subcategories of products, make customizations, and prepare the products catalog. To do it all, we make use of the following modules:

Node import

Using Node import, you can import massive amounts of data from text files.

URL

`http://drupal.org/project/node_import`

Maintainers

Robrecht Jacques, Daniel F. Kudwien, David Donohue, Neil Drumm

Community Tags

Community Tags allows the users of your store to tag your products.

URL

`http://drupal.org/project/community_tags`

Maintainers

Omar Abdel-Wahab, Ronny López

Tagadelic

Tagadelic cooperates with Community Tags, and creates a block with the most popular tags from your store.

URL

`http://drupal.org/project/tagadelic`

Maintainers

Bèr Kessels, Rob Loach

Modules used in Chapter 7

In Chapter 7, we move on to manage our order process and customers. We also integrate features that help us build good customer relationships, and thus, a good business reputation. To achieve that, we make use of the following modules:

CiviCRM

CiviCRM is an open source CRM application. This module synchronizes the user accounts of Drupal with the customers of CiviCRM.

URL

`http://drupal.org/project/civicrm`

Maintainer

Donald A. Lobo

UC Wish List

UC Wish List gives your customers the opportunity to create their wish lists and store them for later use.

URL

`http://drupal.org/project/uc_wishlist`

Maintainer

Ryan Szrama

Simplenews

Simplenews is a module for the creation of newsletters. It supports multiple mailing lists and users can subscribe and unsubscribe to them easily.

URL

`http://drupal.org/project/simplenews`

Maintainers

Erik Stielstra, Rob Roy Barreca

Tweetbacks

Tweetbacks shows the posts from Twitter that refer to pages or products from your store.

URL

`http://drupal.org/project/tweetbacks`

Maintainer

Christopher S. Charabaruk

Twitter

Twitter allows users to display their Twitter posts in their account or to post a tweet when they create new content at your site.

URL

`http://drupal.org/project/twitter`

Maintainers

Jeff Eaton, James Walker

Modules used in Chapter 9

In Chapter 9, we work towards enhancing our UI to present our customers with an intuitive UI. To achieve that, the following modules would be very useful:

Browsing History Recommender

Browsing History Recommender creates recommendations based on browsing history.

URL

`http://drupal.org/project/history_rec`

Maintainer

Daniel Zhou

Relevant Content

Relevant Content creates blocks showing relevant content, based on the taxonomy of the current product.

URL

`http://drupal.org/project/relevant_content`

Maintainer

Nicholas Tompson

Ubercart Products Recommender

Ubercart Products Recommender creates a block for product recommendations, based on the order history of your store.

URL

`http://www.drupal.org/project/uc_rec`

Maintainer

Daniel Zhou

Panels

Panels is a layout builder for Drupal, allowing a site administrator to create customized layouts for multiple uses.

URL

`http://www.drupal.org/project/panels`

Maintainers

Earl Miles, Sam Boyer, Daniel F. Kudwien, Bernd Oliver Sünderhauf

Views Slideshow

Views Slideshow cooperates with Views to create custom slideshows for your products.

URL

`http://drupal.org/project/views_slideshow`

Maintainers

Adam Moore, Fearls Groove, Aaron Winborn

Ubercart Views

Ubercart Views allows you to create specialized views for Ubercart, in order to display user or order data.

URL

`http://drupal.org/project/uc_views`

Maintainer

Mads Peter Henderson

Ubercart Ajax cart

The Ajax cart gives your customers the ability to add products to the cart instantly, without refreshing the page.

URL

`http://drupal.org/project/uc_ajax_cart`

Maintainers

Erik Seifert, Joe Chellman

Ubercart Terms Of Service

Ubercart Terms of Service is a simple module that creates text with terms of service and a checkbox, which needs to be checked before submitting the order.

URL

`http://drupal.org/project/uc_termsofservice`

Maintainer

Pedro Cambra

Ubercart Discounts (Alternative)

The Ubercart Discounts module allows you to create discounts for your store.

URL

`http://drupal.org/project/uc_discounts_alt`

Maintainers

Ezra Barnett Gildesgame, Ryan Groe

Ubercart Discount Coupons

The Ubercart Discount Coupons module allows you to create coupon codes to provide discounts to your customers.

URL

`http://drupal.org/project/uc_coupon`

Maintainers

Dave Long, Blake Lucchesi

UC Multiprice

If you sell your products in many different countries, this module allows you to sell them at different prices for each country.

URL

`http://drupal.org/project/uc_multiprice`

Maintainer

Sjoerd Arendsen

Modules used in Chapter 10

In Chapter 10, we try our hands on various techniques to promote our products and e-store, and also optimize our store for security. The following modules would be very helpful for that:

SEO Checklist

SEO Checklist doesn't add any new functionality to our site, but it suggests useful modules and checks that they are installed and properly configured.

URL

`http://drupal.org/project/seo_checklist`

Maintainers

Dave Reid, Kris Vanderwater, Ben Finklea

Pathauto

Pathauto automatically creates path aliases for our nodes, categories, and users. It generates search engine-friendly URLs and improves the ranking of our pages.

URL

`http://drupal.org/project/pathauto`

Maintainers

Greg Knaddison, Frederik S. Olesen, Clay Butterfly, Mike Ryan

Page Title

This module automatically creates `<title>` elements for our nodes and users. It also gives authors the ability to create a page title different from the node title.

URL

`http://drupal.org/project/page_title`

Maintainers

Nicholas Thompson, John Wilkins, Robert Douglass

Global Redirect

Global Redirect creates page redirects and solves problems related to duplicate content and lower rankings by search engines.

URL

`http://drupal.org/project/globalredirect`

Maintainers

Nicholas Thompson, Alexander Hass

Path redirect

Path redirect also uses redirects, but for another reason. It helps you to create redirects from old or deactivated pages to new ones.

URL

`http://drupal.org/project/path_redirect`

Maintainers

Dave Reid, Alexander Hass, Steven N. Severinghaus, Jeff Robbins

Meta tags

Meta tags creates meta tags for your products and pages.

URL

`http://drupal.org/project/nodewords`

Maintainers

Alberto Paderno, Robrecht Jacques, Gábor Hojtsy

Site map

Site map creates a page with a site map in a readable form. The visitors of your site can view the structure of it, but it's also accessible by search engines.

URL

`http://drupal.org/project/site_map`

Maintainer

Fredrik Jonsson

XML sitemap

XML sitemap automatically creates a site map that is compatible with the XML Sitemap protocol.

URL

`http://drupal.org/project/xmlsitemap`

Maintainers

Dave Reid, Earnie Boyd, Alberto Paderno, Darren Oh

Service links

Service links allows you to share your content in the most popular networking sites, such as Facebook, StumbleUpon, Digg, Delicious, Google, Yahoo, and so on. People spend most of their time in these sites, so it's a great opportunity to inform them about your company and your products.

URL
http://drupal.org/project/service_links

Maintainer
Rob Loach

Ubercart Google Base Integration
Ubercart Google Base Integration allows you to submit your products to Google Merchant Center (formerly known as Google Base).

URL
http://drupal.org/project/uc_gbase

Maintainer
John Youssef Ibrahim

Backup and Migrate
Backup and Migrate simplifies the task of backing up and restoring your database.

URL
http://drupal.org/project/backup_migrate

Maintainer
Ronan Dowling

Security Review
This is a simple module that checks your website and gives you a report about existing vulnerabilities.

URL
http://drupal.org/project/security_review

Maintainer
Ben Jeavons

Mollom

Mollom is one of the best anti-spam tools available. It protects your site from several threats, such as spams in the form of comments, contacts, or password requests, or fake user account registrations.

URL

`http://drupal.org/project/mollom`

Maintainers

Dave Reid, Dries Buytaert

Modules used in Appendix A

Ubercart is suitable not only to sell physical products, but to build a Hotel Booking System as well. In Appendix A, we analyze the following module and we show you how to configure it:

Hotel Booking System for Ubercart

Ubercart can now be used for hotel bookings, thanks to the Hotel Booking System for Ubercart.

URL

`http://drupal.org/project/uc_hotel`

Maintainer

Will Vincent

Summary

Of course, the modules that we studied here are not the only third-party modules related to Ubercart. As it grows in popularity, new modules are added daily that add exciting new features, so visit `http://www.ubercart.org` or `http://www.drupalmodules.com` often and be informed about all the new releases.

C
Free and Commercial Ubercart Themes

In this appendix, you will find some very interesting free and commercial themes. Some of them are designed especially for Ubercart and some of them are general Drupal themes that fit well to an e-store design. You can install them and use them immediately, or customize them to fit to your needs.

Free themes

Theoretically speaking, you can use any of the free Drupal themes from the Drupal website. The themes that we suggest here are designed especially for Ubercart and are appropriate for an e-commerce installation.

Acquia Prosper

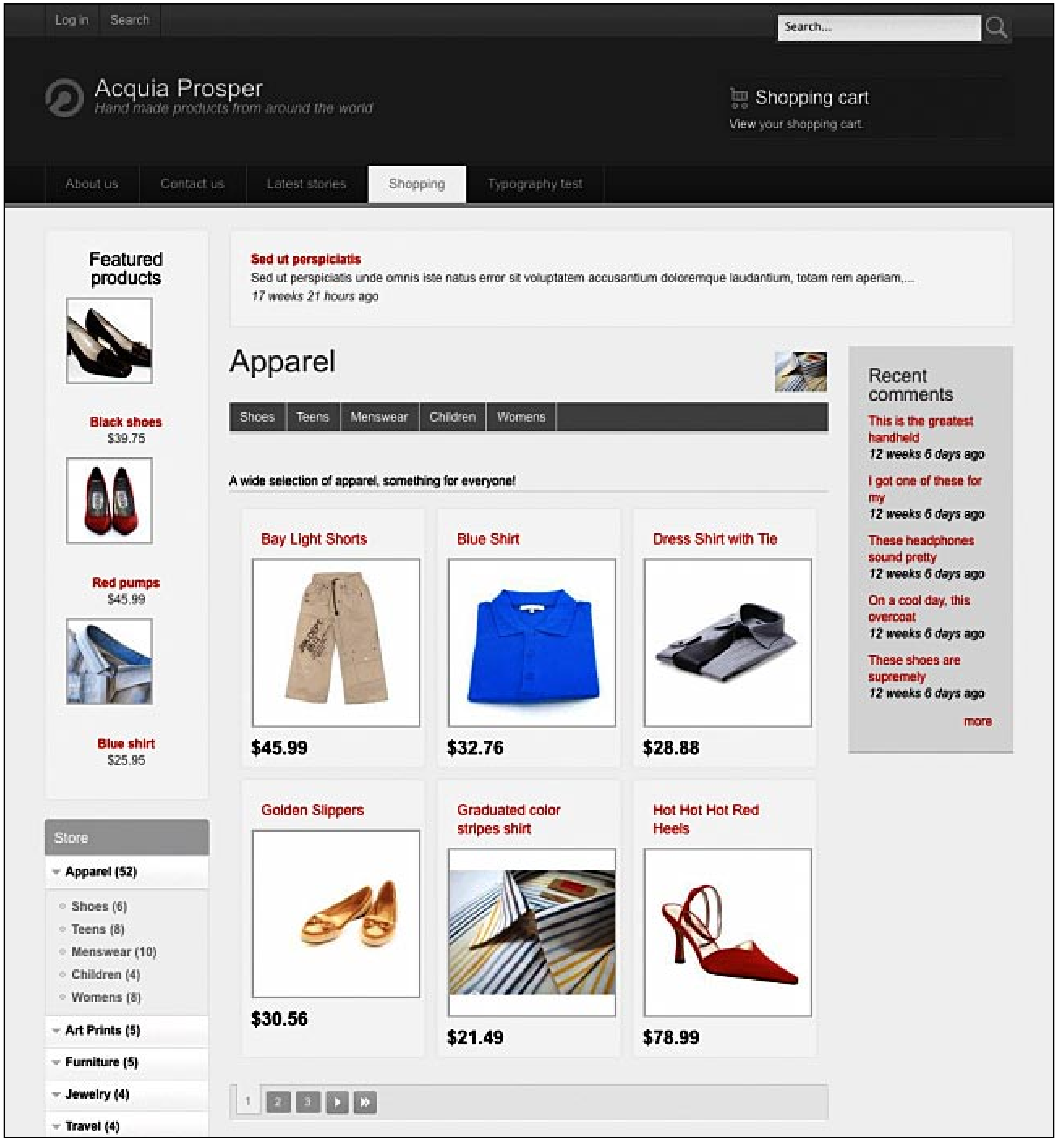
Acquia Prosper is a free theme, designed especially for Ubercart. It is not a standalone theme, but it's a sub-theme of Fusion Core base theme (`http://drupal.org/project/fusion`). It provides enhanced features for your site, such as custom shopping cart icons, custom product templates, and themed catalog grid view.

Creator

Acquia, Top Notch Themes

URL

`http://drupal.org/project/acquia_prosper`



AD Novus

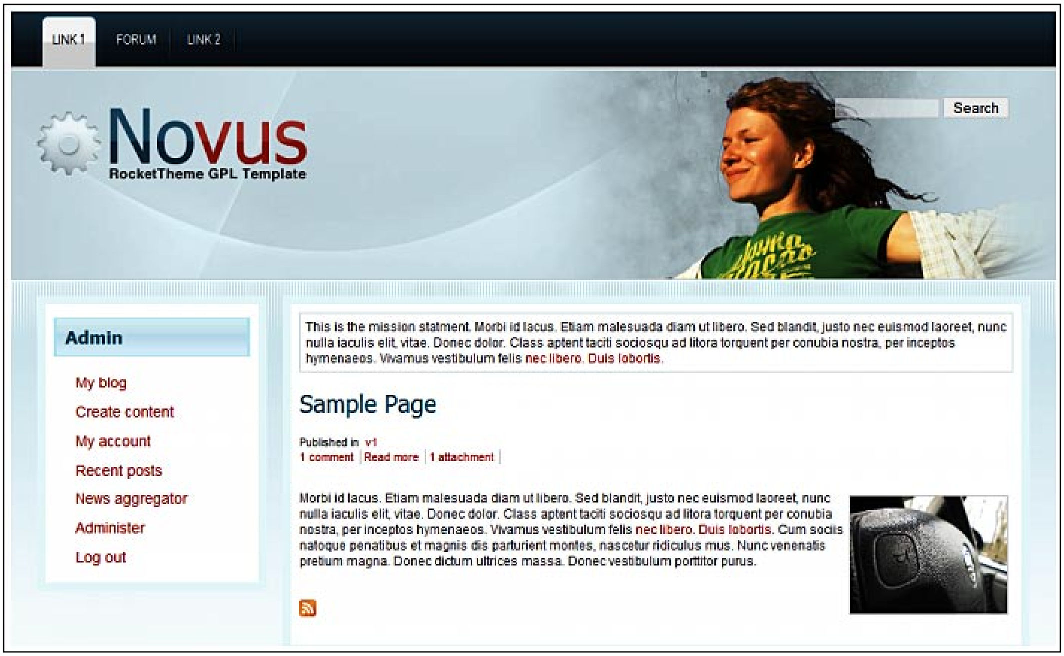
This is a general Drupal theme, but its simplicity is ideal for an e-commerce website. It offers fixed and fluid versions, support for one- two- and three-column layouts, and compatibility with W3C standards.

Creator

Avioso Designs

URL

`http://drupal.org/project/ad_novus`



Acquia Marina

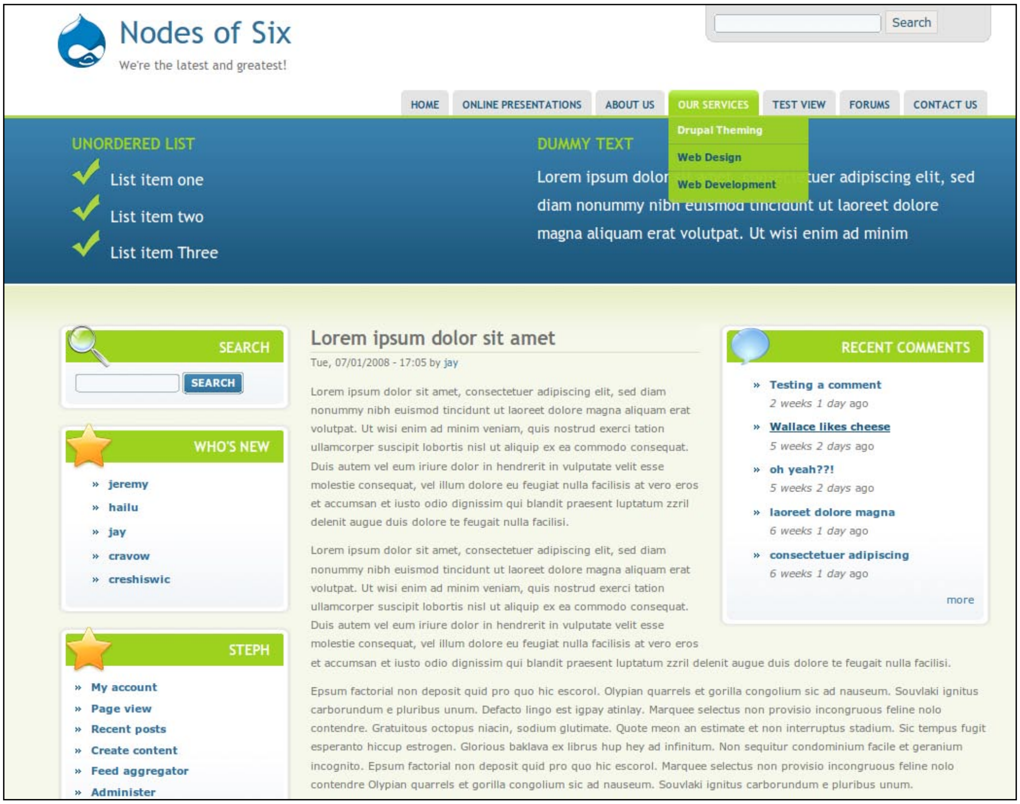
Acquia Marina is another general Drupal theme that works well with Ubercart. It provides one-, two-, or three-column layouts, 15 collapsible block regions, and fluid or fixed layouts. The new version 3 is even more advanced, providing right-to-left theming and point-and-click layout.

Creator

Acquia, Top Notch Themes

URL

`http://drupal.org/project/acquia_marina`



Commercial themes

If you want to have a unique presentation without paying a vast amount of money, then you can download a premium theme from a theme seller.

Market Share

Market Share is a commercial theme for Ubercart. It offers 16 Drupal regions and includes stock photos, logo, and layered Photoshop file. It also has a drop-down menu and custom **Add to cart** buttons.

Creator

Top Notch Themes

URL

`http://www.topnotchthemes.com/theme/1058`

Luxe

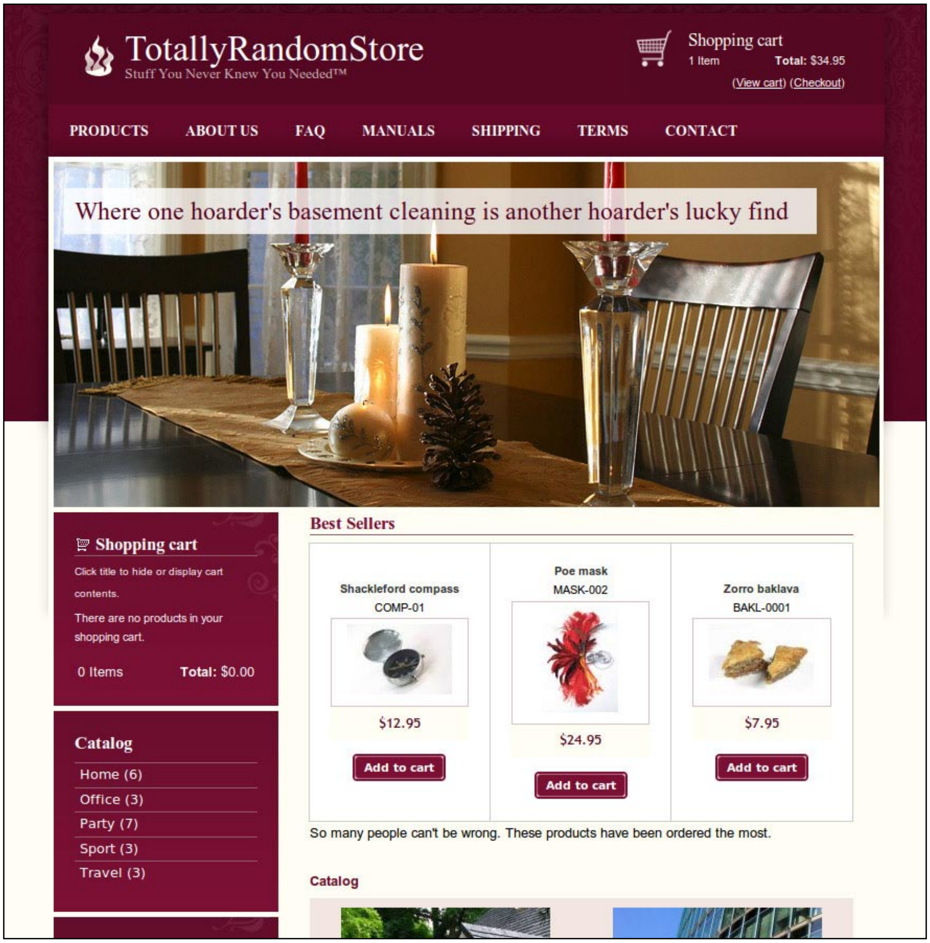
This theme has a fixed layout width of 900 px and provides one- or two-column layouts and 15 block regions. It includes custom theming for the most important elements of Ubercart, such as product pages, attributes, shopping cart, catalog grid, and checkout form.

Creator

Top Notch Themes

URL

`http://www.topnotchthemes.com/theme/1042`



New Media

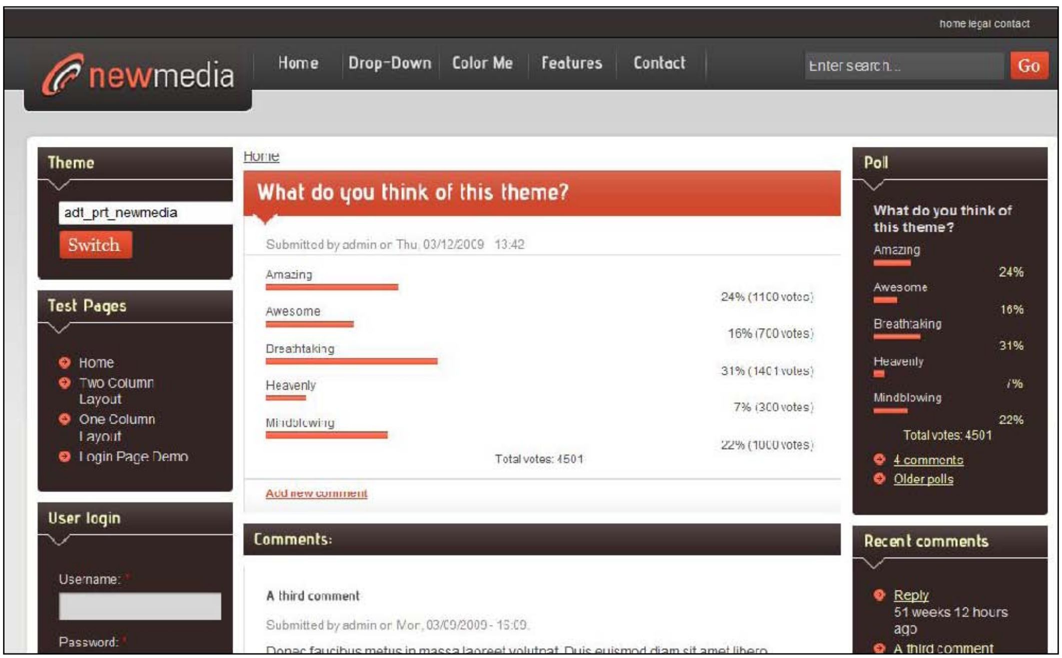
This theme also offers very interesting features, such as a jQuery color picker, many preloaded color schemes, configurable layouts, and advanced theme settings. It also offers multi-level drop-down menus and dynamic regions.

Creator

All Drupal Themes

URL

`http://www.alldrupalthemes.com/drupal-themes/newmedia`



Summary

All of the themes we discussed in this appendix are just a tiny selection, but they are selected to fit your needs and are suitable for your store. Of course, there are thousands of Drupal Themes available at `http://drupal.org/project/Themes`, and there are also dozens of companies providing free and commercial themes. The final choice is yours.

Index

A

B

C

D

E

F

G

M

N

O

P

Q

R

S

T

U

Thank you for buying

Drupal e-commerce with Ubercart 2.x

Packt Open Source Project Royalties

When we sell a book written on an Open Source project, we pay a royalty directly to that project. Therefore by purchasing Drupal e-commerce with Ubercart 2.x, Packt will have given some of the money received to the Ubercart project.

In the long term, we see ourselves and you—customers and readers of our books—as part of the Open Source ecosystem, providing sustainable revenue for the projects we publish on. Our aim at Packt is to establish publishing royalties as an essential part of the service and support a business model that sustains Open Source.

If you're working with an Open Source project that you would like us to publish on, and subsequently pay royalties to, please get in touch with us.

Writing for Packt

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to `author@packtpub.com`. If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.

About Packt Publishing

Packt, pronounced 'packed', published its first book "Mastering phpMyAdmin for Effective MySQL Management" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution-based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: `www.PacktPub.com`.

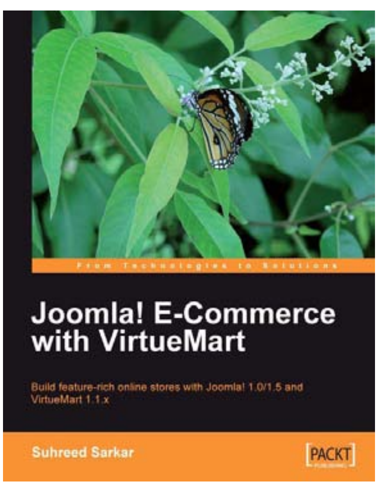
Drupal 7 First look

ISBN: 978-1-849511-22-3 Paperback: 228 pages

Learn the new features of Drupal 7, how they work and how they will impact you

1. Get to grips with all of the new features in Drupal 7
2. Upgrade your Drupal 6 site, themes, and modules to Drupal 7
3. Explore the new Drupal 7 administration interface and map your Drupal 6 administration interface to the new Drupal 7 structure
4. Complete coverage of the DBTNG database layer with usage examples and all API changes for both Themes and Modules

---



Joomla! E-Commerce with VirtueMart

ISBN: 978-1-847196-74-3 Paperback: 476 pages

Build feature-rich online stores with Joomla! 1.0/1.5 and VirtueMart 1.1.x

1. Build your own e-commerce web site from scratch by adding features step-by-step to an example e-commerce web site
2. Configure the shop, build product catalogues, configure user registration settings for VirtueMart to take orders from around the world
3. Manage customers, orders, and a variety of currencies to provide the best customer service
4. Handle shipping in all situations and deal with sales tax rules

Please check **www.PacktPub.com** for information on our titles