Documentation for Drupal Website: drupal 5.x

Ana Willem

Table of Contents

Strategy

This manual is designed specifically for organizations who have just built a website, but might not have complete operating instructions for it. It is written to help you form a deep understanding for the opperating principles that make your site do what it does.

This manual is designed to engender a working knowledge and familiarity with drupal. It is different from a lot of tutorials in the sense that it is not a 'How-to' as such (although there is a 'how-to' section). Drupal is a content management system (more on this later), the operative word of which is 'system'. This manual is designed to develop and understanding of the system, as opposed to an ability to copy how things get done.

It is designed this way because a drupal website is not just a straightforward collection of pages.

Your website is a coordinated system in which parts of that system are created in a variety of different ways. The idea behind this manual is to give you a sense of the system so that you not only understand how to create or modify content, but also so that you understand the basics of how the system works. Ideally you will be able to fully understand how/when/where things happen, and can make intelligent decisions for how to create and edit the items within the context of the system of your website.

This manual will not tell you everything you need to know. There is a drupal.org website for that, not to mention lots of books. The intention is not to make you an expert, but to enable you to have a starting point for configuring your site and a solid basis for broadening your understanding.

I have written this as a generic manual with any information that is specific to your site - indented. The intention behind this is that there are few drupal manuals out there that are generic and not tied into a specific site. I wanted to make this manual available to others as well in order to share the knowledge.

You will benefit immensely by having your browser window open to your website and following along while you read this.

Introduction

When you put in someone's URL into your browser window, you see pictures and text and moving objects. In fact, all web pages exist as text files that refer to other files such as pictures, documents, etc. All these files exist in folders on a computer that is always on (server). The URL is associated with that computer and those files so that when you type in their address, you end up accessing those files.

There was a time (before Web 2.0) when the only folks who could alter anything on the web were programmers who knew these markup languages.

The text file, written in HTML, tells your browser program (Firefox, Safari, Netscape, Internet Explorer, etc) how to visually depict a web page. An example of what this looks like:

```
<!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="en" lang="en">

    <head>
        <title>Irssi - The client of the future</title>
        <link rel="stylesheet" href="/css/main.css" type="text/css" media="all" />
```

```
        <meta name="author" content="The Irssi Project" />
<script src="http://www.google-analytics.com/urchin.js" type="text/javascript">
</script>
<script type="text/javascript">
_uacct = "UA-1191860-1";
urchinTracker();
</script>
    </head>

    <body>

        <div id="container">
            <div id="header"><center><img src="/images/irssitop.png" alt=""
/></center></div>
            <div id="navigation">
                <ul>
                    <li class='firstnav'><a href='/about' id='about'>about</a></li>
<li class='nav'><a href='/support' id='support'>support</a></li>

<li class='nav'><a href='/bugs' id='bugs'>bugs</a></li>
<li class='nav'><a href='/documentation' id='documentation' class='selected'>documentation</
a></li>
<li class='nav'><a href='/download' id='download'>download</a></li>
<li class='nav'><a href='/news' id='news'>news</a></li>
<li class='nav'><a href='/themes' id='themes'>themes</a></li>
<li class='nav'><a href='/scripts' id='scripts'>scripts</a></li>
                </ul>
            </div>
            <div id="contentbox">

                <div id="sidebar">
```

etc...

Eventually, programs were developed that enabled non-coders, with some training, to modify websites. Dreamweaver is an example of such a program as is FrontPage. These programs create visual representations of the web site, and simultaneously translate them into a text .html file. Because of the way they work, the code they create is often cumbersome and difficult to read. In addition, they require the downloading and uploading of files from the site in order to modify and/or test them. They also require for the user to purchase and install the program. This can be cumbersome.

Web 2.0, using programing languages like php and mysql, creates a database coupled with a user interface that enables regular folks to change the look and structure of the website, including the content and the way that content populates, in a way that does not require the use or knowledge of any programming languages. It is easier to use than programs such as Dreamweaver because there is no need to download content in order to modify it. Everything happens on the web, and so the user does not need a special program, and websites can be modified from any computer with an internet connection.

Drupal

Drupal is a content management system (CMS). It is a kind of program that is installed on the server or in the files where your server is. There are many different kinds of content management systems out there. Many of them are proprietary, which means that they cost money and are not customizable by the user. Drupal is open source, which means that the code it is both modifiable by the user, and the application/program is free.

What this means for you is that not only are users without programming experience able to modify content on the site, but it also means that the specific programs themselves can be modified to suit the special needs of the website.

Drupal has two types of areas where most of the 'work' of the website is done. The first area is a folder that contains 'Themes' for your site, and the second folder holds 'Modules'.

Themes

Themes are what make a website look like it does. They operate both from the perspective of what colors, images and fonts are used on your site, but also from the standpoint of the structure of your pages. In order to modify the theme, you must know CSS, HTML, and php. What this means is that modifications to the themes are possible, but only by a programmer. The look of the site should not, however, be confused with the content.

Drupal themes can cost money, but a lot of them are free. Often, when creating a custom theme, a programmer will download a free theme that is close to what is wanted, and then will customize that theme according to the client needs.

> All of the content in the pages and different sections on the site (except the header movies) can be modified by non-programmers. Including which logo is used (but not where that logo appears).
>
> There are certain aspects of the look of the SCHR site that are not modifiable by non-programmers. Because the SCHR site uses flash movies in the header, it required special programming to position this movie. The flash movies themselves cannot be modified, but the when different flash movies appear on which pages can. In addition, which elements appear in the header can be modified by the user, but not the way they look.
>
> SCHR uses 2 themes. Multiple themes are used so that the front page can look differently from the inside pages in terms of layout and look of the pages. One theme is used only on the front page, and the other is for all of the internal pages. This is made possible using a 'module' (which we will go over later) called 'sections'. The fact that it is governed by a 'module' means that there are likely ways to configure the way this function operates without needing to know any programming languages.

The place themes are typically altered or modified happens in two places. The first is under the administration menu - Administration → Site building → Themes - and the second (site slogan, mission, etc) in the – Administration → Site configuration → Site Information section.

In order to access the Administration section of the site, you will need permissions. We will go over that lower down in this document.

Site Information

Associated with the themes is the 'Administer → Site configuration → Site information' section. It is a good page to get used to as it has basic information about your site on it. Among other things, you can set up how the site footer reads, what the slogan/mission for the site is, and who the site email goes to and is from. In addition, there is a section where you can tell it what the front page of your site should be.

Modules

Modules are building blocks that contain specific functions for the site. Example: if you have been on a website that asks you to enter a combination of letters and numbers (to ensure that you are not a web robot) – in drupal, this is a piece of functionality that is governed by a module. Being able to see printer-friendly pages is governed by another module. More complicated functions such as e commerce are often governed by a group of modules that are packaged together.

Some modules come with the initial installation of drupal. These are called 'core' modules. Other modules have to be installed separately. These are called 'contributed' modules. They have been programmed by different people or groups, and have been 'contributed' to the general drupal community for free. There are probably over 1000 contributed modules – so if there is something that you need the website to do, chances are there is a module that will do it. Modules can be turned on and off for the site by going to 'Administer → Site building → Modules'.

Modules often need to be installed and set-up/configured. The configuration and set-up that needs to be done depends on the module. Configuration options for modules (if there are any) are listed under the 'Administration → Site configuration' menu.

Content/Nodes

In drupal, there are a variety of different types of 'content' that you can work with. 'Content' in this sense, is any single item that is created in a drupal website. There are some 'content types' that come with your drupal installation (page, story, book).

> The content types that have been created specifically for the SCHR website are jobs and resources. The 'Jobs' content type is designed to hold information about employment opportunities at SCHR. The 'Resources' content type is designed to hold media, articles and other information resources about SCHR or SCHR's work.

Some content types are created by the installation of new modules. Examples of this are the 'Product' or 'Product Kit' content types that come with the 'Ubercart' module (e commerce). The 'Events' module creates a content type called 'Event'. The 'Webform' content type, which is part of the 'webform' module, enables the creation of custom forms that can be downloaded into a spreadsheet format.

> SCHR uses the Webforms content type to gather information about Volunteers. It uses the 'Events' content type to keep track of events that will be sponsored by SCHR.

Content types that exist or that come with the drupal installation can have additional fields added to them. This is enabled by a special module called 'CCK'.

> CCK is the module that was used to create the 'Jobs' and “Resources' content types. It also enabled the creation of additional fields for the 'Events' content type.

Each piece creates a 'node'. A node is a non-specific term for a content item. Nodes are the way that the underlying drupal database keeps track of certain pieces of content. Any time a page or webform, or event, or any other kind of content is created on the website, it is assigned a number (like 304), and a default path like '/node/304' or a URL like 'mysite.com/node/304'. Again, node 304 could be a page, or it could be another content type such as an event.

Paths/URLs

Paths and URLs are similar. They tell your browser what page you are on. Understanding the difference between a path and a URL will enable you to correctly configure different parts of your website, and it will also enable you to understand a principal way that drupal functions.

Drupal categorizes web pages in two ways: pages that exist within the context of your website, and pages outside of your website. Every web page has a 'root' URL. The root constitutes the common denominator of your website. In other words, it is the base URL that all other URLs in your site use to define themselves.

> For SCHR, the root URL is 'http://schr.org' or 'http://www.schr.org'. Any and every additional page in your site will start with that 'root'.

In drupal, your 'root' URL is understood. Anytime you specify a location that does not start with a 'root' URL designator (mysite.com or whatever), drupal will assume that the location is within your site. This is called a path. There are sometimes when you use a 'path' that you will need to begin the pathname with a single '/' (forward slash) in front of it. Generally when drupal asks you for a 'path' it will assume the leading forward slash (/).

There are times, specifically when you are linking to other pages in your site, when you will need to include the leading forward slash in the pathname - '/node/304' - instead of 'node/304'.

The rule of thumb is: 1) first check if it tells you how it wants you to write the path. 2) if it doesn't specify then try it first including the leading slash and if that doesn't work 3) try it without that leading slash. There are reasons to do it in this order that are too complicated to explain here – but they have to do with the fact that paths can be 'relative'. You can find more information about this if you google 'relative path names'.

Menus

Menus represent a hierarchical list of content (like pages) and how they are listed with respect to other pages of the site. A menu is usually represented by a list of pages (or other types of content) with other 'sub-pages' listed under those 'primary' menu headings.

Menu's can be set in two places.

The first, and most advised, way to set up a menu item is when you create the page or content item. When you are creating a page, if you scroll down below the 'body' section, you will find a click-able (collapsed) item called 'Menu'. When you click on it, it will un-collapse and you will see different option for creating and configuring the menu item. The first heading enables you to give it a title. In addition, you will find a place to give the menu item a place in the hierarchy, and also a weight within that place.

The second way to create a menu item (and the place you will want to go when you edit existing menu items) is in the Administer → Site building → Menus. A common mistake is to go here first and create all of your menu items first. This is a bad idea because you cannot assign a piece of content to a specific already existing menu item from the content creation or editing pages. This means that you want to configure menu items here only after the content items have been created. There is a reason things are done this way.

There are some circumstances when you will be creating a menu item for a piece of content from the menu page (as with 'Views', for example). In these cases you will be asked to give the menu item a 'path' when you create it, as well as a place in the hierarchy of other menu items.

Because of the way that hierarchy works, creating 'primary' or 'root' level content makes placing 'sub-pages' and content items easier. Setting up how this hierarchy will look before you start creating content will make your job easier, and will save you having to repeat steps and edit pages several times. When you are choosing the level on the hierarchy where the menu item will appear, you are choosing the 'parent' item, or the item that you want your content item to be listed under.

So in the following example...:

Primary Heading A – weight: 0
- Sub-Heading A – weight: -7
    - 2nd Sub-Heading A – weight: -4
    - 2nd Sub-Heading B – weight: 3
    - 2nd Sub-Heading C – weight: 5
- Sub-Heading B – weight: -2
    - 2nd Sub-Heading A – weight: -5
    - 2nd Sub-Heading B – weight: -1
        - 3rd Sub-Heading A – weight: -8

Primary Heading B – weight: 6
- Sub-Heading A – weight: -8
    - 2nd Sub-Heading A – weight: -1
    - 2nd Sub-Heading B – weight: 2
    - 2nd Sub-Heading C – weight: 3
- Sub-Heading B – weight: -1

...the second primary heading is placed under the first one by giving it a 'heavier' weight. They are both listed in the same level of hierarchy (under the menu item itself). By contrast, while there are 2nd Sub-Headings with the same weight listed under different Sub-Headings. They are placed under different Primary Headings and Sub-Heading by choosing the 'parent' item in the hierarchical list. The weights place content items with respect to other pieces of content with the same 'parent'.

In oder to give a menu item a weight, you will need to check to see what the other menu weights are. This is unfortunate (and time consuming) but the weights of other items are not displayed on the menu list. The menu weights are arbitrary. In other words, they are only meaningful in relationship to one another. The numbers don't mean anything in and of themselves.

Menu's all have a name. There are two menus that come with your drupal installation: Primary Links and Navigation. The Primary Links menu item is where you want to place your main navigation for the site. The Navigation is a menu item that is only programed to be seen by folks who login to the site (also called 'authenticated' users). It is programed to automatically be called whatever that person's username is. The idea is that you would place things under that menu item that you only want users who are 'authenticated' to see. There are some items that will appear automatically. They can be disabled by going to 'Administer → Site building → Menus'.

Blocks

Web pages have different sections: header, left-hand navigation, footer, right-hand navigation, content – just to name a few. Which sections you have and where they are is defined by the theme that you installed. You can populate these different parts of the site with different types of information using blocks.

To configure blocks, you should go to the 'Administer → Site building → Blocks' section of the site. You will see a list of pre-defined blocks that come with your drupal installation and theme. Some blocks are created by modules, and become listed once the module is installed and configured. Others are built using other different functions and sections of your site (views, taxonomy/categories, etc). There is also an option to create your own blocks.

There are two places that blocks are configured. The first is on the page that lists the blocks. Next to each block are a couple of dropdown menus. One menu places the block in a certain area of the webpage. The second block determines the 'weight' or order in which the block content is listed in that section with respect to the other blocks.

The second way that blocks are configured is by clicking the 'configure' link at the end of the row. This is where you can give the block a title, tell it what different permission levels should be able to access or see it, and which pages should display it.

Categories

Categories (also sometimes referred to as 'taxonomy') enable the ability to label certain content as being similar. The reason this is valuable is that it gives you the ability to have a page that automatically populates 'teasers' (short descriptions) of all of the content that has been assigned a specific term in a category. The easiest way to demonstrate how categories work is to show them in action.

Let's say that I have a website-magazine (webzine) with articles about sculpture. This magazine, over the years will collect a lot of different articles. If I want to make it easier for my readers to find articles or information specifically about gallery installation art in New York, categories will enable me to do this. In this case, there might be two categories created: Form and Metropolitan Area. Form would categorize different types of sculpture and might include: gallery installations, public art, juried group shows, gallery show, open studio, etc. The Metropolitan Area category would might include: New York, Boston, Providence, general New England, etc...

In the creation of your site, various 'categories' were created. The categories that already exist on your site can be found at 'Administer → Content Management → Categories'. You will see that they are listed on this page with their 'title' in the first column. The second column shows which content types that are relevant to this category, and then there are links to edit the vocabulary (the umbrella category like 'Form'), view the list of items under that category/vocabulary, and add items to that category/vocabulary.

If you go to view the terms on the list, and click on any one of those terms, you will be taken to a page that aggregates all existing content types that have been assigned this term.

The 'edit vocabulary' link will take you to a place where you can assign the category to specific content types, give the category a description, choose how the vocabulary/category will organize itself (hierarchy) and choose if multiple terms, etc, will be permitted. These are all 'macro' functions of the vocabulary/category itself.

When you click to add an item to a specific vocabulary, you are asked to give it a title and a weight. If you assigned a hierarchy to the vocabulary/category, you will also have an opportunity to give the term a 'parent' item. If this is the case, you will want to create all of the first-level hierarchy terms first and give them the parent called '<root>'.

> An example on the SCHR site is called 'Cases' and consists of a list of cases that SCHR has tried. It is not a conclusive list and there might be a need to be added to as certain cases become more active in the news. When you create a 'Resource' you are asked if you want to assign the resource to a specific case. Once a resource is assigned a case, it will become available for any 'view' or block or page that displays content for that category item. 'Cases' and 'Resources' are both each categories.

Views

'Views' is a module that you can install. The purpose of a 'view' is to find and show certain kinds of content displayed in a certain way. While it shares the Category's ability to aggregate certain content, it's abilities extend way beyond that.

Similar to menus and blocks, there are some views that come with the module, others that come as part of other modules, and then an infinite number of views that you can create yourself.

The 'views' created and used by your website can be found at 'Administer → Site building → Views'. Feel free to look through them. Views are complicated enough that to fully go over them is beyond the scope of this manual. Looking through them yourself is the most instructive way to learn how they are handled and what they do. As you can see, each view is listed here (usually in alphabetical order) with a short description, and information about if is a block or a page, and what the URL of that page is. There are also options to edit, export, delete, and clone each view.

When editing or creating a view, there are several different aspects of it's configuration, and they are listed (in order) as the main links/menus (some of which will be collapsed) when you are in the creation or edit screens for a view.

Basic Information: where the view is given a database name, assigned a description, and assigned permissions levels.

Page: A view can show up as a page (with it's own URL, etc) and/or can be created into a block that can populated in various areas of your page and on whichever pages you assign it to. If you want this view to be available as it's own page, you will want to assign it a URL, a title, a description and also how this view will appear on that page (view type). The options are typically: Full Nodes, Teaser List, Table View, List View. There might be others depending on what other modules you have installed on your site. Remember that part of the function of a view is to find and aggregate content from your site based on criteria that you give it. Full Nodes will show the full view of every piece of content that the 'view' finds. Teaser list will only show the first 400 or so characters of each content type that the view finds. To change that, you will need to go to: Administer → Content management → Post settings. A table view will show a table populated with just the types of fields that you specify, and list view will list all the fields that you select in the 'fields section' for each content item that comes back. You will also be able to give the page a header, a footer, text to show if the view finds no content, and assign it a menu item. The menu item, if you set one, will need to be configured and in the 'Menu' configuration interface discussed above. You can give it a title and a weight here in the view interface, and have other options with respect to this that are too broad to go into here.

***Note on Teasers**: Teaser list will only show the first 400 or so characters of each content type that the view finds. To change that, you will need to go to: Administer → Content management → Post settings.

Block: You also have the option to create a 'block' that will be populated by the content that the view brings back. Blocks have a similar set of options that 'pages' have.

Fields: Fields are where specific items of content are selected for view in the 'Table View' it is only relevant if you have selected 'Table View' as the view type for a page or a block. There are options to enable sort, give each item/column a name, sort, and move items up and down on the list. There are other options here for configuring as well, but they are dependent on the 'Fields' you select.

Arguments: This is a code snippet that give the view a little more information about how to select the content it is selecting. In most views, the argument will be empty. In the cases where it is not, DO NOT TOUCH or alter it unless you know php. It can be somewhat time consuming to rewrite these (read: expensive).

Filters: You might have a dozen different content types on your site, and there might be up to 1000 different descrete pieces of content on your site. Filters tell the view what it is looking for. You can filter based on things like content type, categories (taxonomy terms), author, date, etc. For each filtering mechanism you choose, you will have options that tell the filter what it is looking for.

Exposed Filters: This is made available so that users of your website can further filter for themselves the kinds of content they want to see. We used the example of gallery installation art from New York when we were talking about Categories above. If you chose 'Form' and 'Metropolitan Area' items as filters (Taxonomy Terms for ...) then you could 'Expose' the filters for your users to be able to further refine their search. A filter can only be exposed if it is selected in the section above.

Sort: This tells your view how you want the content sorted. If you specify something here, then any sorts that you enable in the 'Fields' for the 'Table View' will automatically be disregarded.

Views/Blocks/Categories: a Case Study

This is to demonstrate an instance of how each of these items works together in your site.

THE EXAMPLE ITSELF is not relevant, in that SCHR does not need to ACTUALLY do this in order to create a resource with a new case associated with it. It is just an example. In reality, what needs to be done are steps: 1, 6 and 7.

Click on the left-hand navigation menu item 'The Death Penalty' (given a 'menu' assignment when the page was created, and assigned it's location with 'blocks'). You will see a link to 'Resources' on the right-hand area of the page. The 'Resources' link was created using a 'block' that links the word 'Resources' to the resources page. There is some coding magic happening here to make it look like a button and to change the way it looks when you hover over it – but it is essentially a 'block'.

When you get to the resources page, you will see towards the bottom the text “Press releases and media coverage of select litigation:” with a bunch of cases underneath it, including the option “All Cases”. If you click on any of those options (except for 'All Cases'), you will be taken to a page that lists resources that have to do with each case. Each of these pages was created with a view. For the 'Alto' case, the view is called 'case_alto' and the 'path' for that view is 'action/resources/by_case/alto'.

In this example the 'case_alto' view is also a block (as are all the cases). If from the 'Resources' page that we are talking about above, you chose 'All Cases', you would be taken to a page that was created, but for which no content was placed in the 'Body' section – so the page only has a Title: path – 'action/resources/by_case'. Because I created a 'view' for each case, and because each view specifies a page AND a block, when you navigate to the 'Blocks' section of the site, you will find that in the 'content' section of blocks, there are a bunch of cases specified. If you click to configure any of these blocks, you will see that it is only visible on one page: the empty one that we created before. The 'weights' of these blocks tell the blocks which order to place all of the different views on this page.

As an example, lets go through all of the steps to adding a case to the list so that you can add it as a category and publish a resource:

1. Go to Administer → Content management → Categories and add it as a term (leave the weight as 0). As we have said in other areas, weights default to sort alphabetically, so if you have more vocabulary terms than weights, you can group items some under the same weight if alphabetically they are kept in the order you want. As an example, a vocabulary term for 'Cases' is 'Alto' and another one is 'Tutwiler'..
2. Go to Administer → Site Building → Views and find a view with the 'case_' prefix. Clone it.
3. When you clone it, you will need to change various aspects of the clone including (but not limited to):
    1. name
    2. page name
    3. block name
    4. url – give it the following path: action/resources/by_case/<casename> . When I write anything and put arrows (<>) on either side of a word, it means that you would substitute that item for whatever real term that is appropriate. So in this case, you would replace '<casename>' with an actual case name like 'alto'.
    5. taxonomy-case filter (select the category item you just created)
    6. remember to use a similar naming convention as the cloned item (or other similar views).
4. Go to Administer → Site building → Blocks and scroll to the bottom to find the block you just created when you cloned the view. Click to 'configure' it, and ONLY change:
    1. Page specific visibility setting (change to 'Show only on the listed pages').
    2. In the box below that write the path name: 'action/resources/by_case'.
    3. Click save/submit.
5. Find the block you created in when you were making the view (at the bottom of the list), and set it to show in the 'Content' section (dropdown). Click save. Now find the block in the 'Content' section and make sure it appears (weight) in alphabetical order. You might have to change the weight of other items.
6. Manually create a link to that view by editing the 'Resources' page (/action/resources).

7. Now create the resource/s that you were wanting to create. If there are no resources created for that case, the case block will not appear on the page.

You might want to test it. To do so, it gets tested in two places. Scroll to the bottom of the page and click on the 'All Cases' link of the case list. Does it appear in that section? Now hit back in your browser and click on the link for the case you just added. Does it go to that page? If the answer is not to any of these questions – then you will need to trace your steps back, and make sure that you did them all correctly.

Permissions/Roles/Users

Which users can see different types of content is managed by Permissions, Access control, Roles, and Users. These can all be found under 'Administer → User management'.

Users are all folks who have registered on the site. In the registration process, they are either given or select a password. There are other questions that can be asked of folks who register on the site. The ability to create these questions is enabled vis a vis the 'Profile' module. If it is not installed on the site it can be.

Once a user is registered, they move from being known as an 'anonymous' user, to being an authenticated' user. A middle step can be configured such that all new users who 'register' for the site must be approved before becoming 'authenticated'. This helps safe-guard sensitive information from getting into the hands of anyone who registers on the site. This can be configured at 'Administer → User Management → User Settings'. Existing users can be seen (and new ones can be added) at 'Administer → User Management → User Settings'.

'Authenticated' and 'Anonymous' are all Roles that come with the drupal installation. Roles are designations that you can give users (when you edit their profile) that give or deny them access to certain parts of the site. Roles can be seen and configured at 'Administer → User Management → Roles'. Roles can be added as needed. While the rolls that you add can be deleted and configured, the default rolls 'Authenticated' and 'Anonymous' cannot.

> On the SCHR website, there are the additional roles of 'Site Editor' and 'Administrator'. 'Site Editor' is a role that the entire staff has (the ones listed as users – we might need to add some). This is so that they all have access to changing small bits of content on the site.
>
> The 'Administrator' role has full permissions on the site.

Permissions are set up for all users by going to 'Administer → User Management → Access Control'. All roles will be listed in columns, and each row represents a different type of permission. These can be tricky so be careful about how you change them. As the site Administrator, it is a good idea to establish a test user for yourself with each of the roles so that you can test them.

Access Denied and Page not Found pages

When a user of the site tries to access a part of the site that is unavailable to them (or a page that does not exist), they will be met with a generic 'Page not found (404)' or 'Access Denied' page that comes with the drupal installation. If you wish, these pages can be customized. The first step is to create a page with the text that you want, and to give it a URL name that is very simple, like 'woops'. The url will, of course, have to be different for each of the different kinds of pages.

The next step is to tell your site that it should go to these pages, instead of the default ones. This is done at 'Administer → Site configuration → Error reporting'.

> SCHR's access denied, and page not found page are customized.
>
> Their paths are '/naughty' and '/woops' respectively.

Content Creation

Once you are logged into the site, you will notice that at the top of every editable page, there are tabs that say 'View', 'Edit', etc. You might have others depending on your permissions levels and the kinds of modules you have on the site. Generally, people who are not signed in (or authenticated) on the site will not be able to view those tabs.

When you are viewing a page, you will notice the 'View' tab is highlighted. To edit a page, you will want to click on the 'Edit' tab.

If you are creating a piece of content from scratch, you will need to Navigate to the 'Create Content' menu item, and click on the content type you wish to create. The location of this page depends on the setup of your site.

> On the SCHR site, there are two places to create content. The first is in the 'Site Editor' menu, and the other is in 'Administer → Content Management'.

Every type of editable page or content item (View, Blocks and Category pages being the main exceptions) has a title that will typically appear at the top of the page, and a 'body'. It might also have lists to select a category or other additional fields. The 'body' field is also sometimes named something different.

> On the SCHR site, any place where you can input text that will accept formating will have a WYSIWYG icon list at the top. I go more into what a WYSIWYG below.

In addition to the above mentioned fields, and any other additional ones that are specific to the content type, there are other options for editing that are included on every edit page. I will go over the important ones here, but there might be others depending on your set up, and some of them on the list here you might not have depending on your site setup. The headers for each are generally collapsed, and when you click on them will open up:

Input Format: this tells drupal how to interpret the text you are entering into your 'body' section. HTML and PHP are languages that exist as code, and this setting tells drupal if there is any of this kind of code in the text you are writing, and what kind it is. Then, when opening the page, drupal will take this into consideration and will present the page accordingly to the user. The options are typically: Filtered HTML, Full HTML, and PHP code. The default is usually 'Filtered HTML' and it is also what works best with most WYSIWYGs (explained below). Others can be created as well and existing ones can be modified by going to 'Administer → Site configuration → Input Format'.

Menu settings: we have discussed menus in more detail above. This configuration area allows you to give the menu item a name, weight, and place in the existing menu hierarchy. It also allows you to giveit a description, which is generally not something that people do, but might be helpful for your site. When hovering your cursor over the menu item, if you set a description, it will pop up on the screen.

URL path settings: we have also spoken about this in more detail above. The path in this case does not need to begin with the leading forward slash (/). The url should coorespond as much as possible with your menu hierarchy so that if for example you are creating a page that is under a parent called 'action', your URL/path for the page you are creating will want to be 'action/your_page_name'.

It is NOT advisable to change the 'path' or 'url' of a page because that will break any bookmarks that other users have created to access that page. URL/paths are set in the 'Alias' group. You'll have to uncheck the check box in order to set them (more below).

> In the SCHR site, in order to maximise of Search Engine Ratings (SEO Checklist Module) there is an 'Autopath' module installed that, when creating a new page, will automatically give it a URL based on your title. When possible, you want to uncheck this and give it your own path/URL to avoid excessively long URLs. The settings for this module are at 'Administer → Site configuration → Autopath'.

Comment settings: If the comment module is installed and enabled on your site, clicking here will let you set up how and if comments are enabled. Comments on pages, used in conjunction with the 'Publishing information' can allow authenticated users to collectively edit and comment on pages before they are officially 'published' for others to see.

File attachments: We discuss this in detail under the 'Linking' section of this manual. This area enables you to upload files from your computer so that they are available for this page.

Authoring information: this automatically gets filled in with the username of the person who originally created the page. It can be changed to assign other authors to the content item you are editing.

Publishing information: This enables you to set up whether this content item is published (visible to non-authenticated users) and also if it 'floats' to the top of lists (used for automatic feeds) and if this is a new revision (used when editing something with other folks).

TinyMCE WYSIWYG: Formatting without HTML/CSS

> SCHR has a WYSIWYG installed on it called TinyMCE. WYSIWYG is an acronym that stands for 'What You See Is What You Get'. What this means is that instead of having to use HTML tags to format text in the body of your content, you can use an interface that is similar to the kind of interface that you are used to seeing in a word processing program like Open Office or Microsoft Word.
>
> Some of the icons you will be used to, and others will need explaining. To see what any of them do, you can hover over them. Before going into TinyMCE, it's important to know that there are places to configure how it is set up, which icons you see, and which users can use it by default. It's important not to mess too much with these settings unless you know what they mean. It can be hard to see if something you configured has messed things up until it's too late. In addition, some of the settings

are retroactive, and can change content that you have already created site-wide.

There are various items that are presently configured in the WYSIWYG that enable you to add links, anchors, images, cut/paste, and change the formatting of the text. I will go over each option here separately.

Links

Links, in the language of the internet are known as Hyperlinks. Whenever you roll your cursor over something and it turns into a hand, you know that you can click on it and will be sent either to a page inside the website (internal link), a page on another website (external link) or a specific place on the same page (anchor).

There are two icons at the top of the body section of the page you are editing. One looks like a chain link, and the other is right next to it and looks like a broken chain link. As you might guess, the icon that looks like a link is what you click to make a link, and the icon that looks like a broken link is what you would click to get rid of a link.

To make a link on a word (or image) you first need to select/highlight it. Once you do that, you will click on the link icon, and another little window will popup with options.

The first option is the 'Link URL'. In this case, that name is deceiving because the 'URL' that you put here might be a 'path' if it is an internal link instead a link outside of the site.

With the WYSIWYG tool (TinyMCE) that we have installed, the way it knows whether something is an internal link or an external link is if you give it a 'path' or a 'url'. As we discussed earlier, a path is the information that comes after the root of your site. If your site is mysite.com, then a path for that site might be '/level_one/level_2/webpage'. In this case, you will need to include the leading forward slash (/) for simplicity. If you want to link to an external webpage, you will need to give it the full URL (including the leading 'http://'). If you do not insert that leading 'http://', the tool will think that you are linking to an internal page, and will return an error.

The second option is called the Target. It simply designates if you want for the link to open up on the same page or if you want it to open up a new page/window in the user's browser. The rule of thumb is that whenever you are steering someone away from your site (to an external site) you want to have that page open in a different tab/window in order to keep your site open and encourage further viewing.

The third option is something that is usually ignored. It is the word that will be shown to the user if they hover over the Hyperlink for any length of time. It's not something that is generally used except for specific applications.

Once you click submit on the window, the item you just linked will turn a color or get underlined or will somehow look different. What it looks like depends on the theming settings of your site, and the way that TinyMCE is configured (to use the site's theme or css file, or it's own).

When you click submit on the edit page, you will see that the item you just linked is Hyperlinked. ALWAYS test the links that you make.

To unlink something, select it from the edit screen and click the 'broken link' icon.

Linking Attached Documents

You can also link to attached documents. The first step is to attach the documents to the page that you are editing. You do this by scrolling down and finding the heading 'File attachments' and clicking on it. You will see a 'Browse' button in which to search on your computer for a file. There are a few things you want to keep in mind when you are attaching files:

1. You want to make sure the name of the file is brief but descriptive.
2. You do NOTt want the name of the file to have capital letters, spaces, single or double quotation marks or any other characters (!@#$%^&*[]{}|\?/~`+=).
3. You CAN have dashes (-) or underscores (_).
4. Make sure the file size is UNDER 5MB. Anything larger has no business on the website as it could crash other folks' computers who might have limited or slow access.

Once you have selected your file, click on the upload button. You will see the computer think a minute and then list the file. There are two check boxes: 'Delete' and 'List'.

The 'Delete' check box is obvious. It will not delete the file until you save out of the edit screen. Make sure to be careful when deleting a file. If there are other pages that use the same file, it will delete it for them as well, and you will have broken links on your site.

The 'List' check box is checked by default. If checked, it will list the file at the bottom of the page.

> Generally, on the SCHR site, the convention is that we do not list any of those files, and instead are linking to them from the text. We are doing this because the file names aren't ever very good at describing what the file is.

You can change the way that name reads by clicking in the box with the current file name and changing it.

Also pay special note to the small print below the file you just linked. That small print is the actual URL of the file. Because it is an internal link, you do not need to include the root of the site (http://mysite.com), but just the part that comes after that. Don't forget the leading slash.

> At this point, you would highlight the text that you want to link to the file, and click the chain link icon.

Creating Anchors and linking to them

An anchor, as we described earlier, is a link to a specific place on the page that you are currently on. It is the same as a link, except that instead of taking you to another page, it takes the viewer to a different section on the same page. This is done when there is a lot of information on one page, and you want to create a sort of 'Index' at the top that people can link to. It's also sometimes used to direct people back to the top of the page.

> In order to link to an anchor, you need to create one first. Navigate to the section of the page that you want to anchor to, and place your icon there. Remember that the anchor will make this the very top of the page, and so sometimes people place the anchor a few lines above the part they are linking to so as to give it a little breathing room. Once you have put your cursor on that line, click on

the anchor tool. It will ask you to give the anchor a name. Be sure to write that name down so that you can remember it – especially if you have several anchors on one page.

Once you have created your anchor, you will go to the actual word that you want to make a hyperlink for, highlight it, and click on the 'Link' icon. At this point, instead of a 'path' or 'URL' you are going to give the link and 'anchor'. All anchor start with a leading '#' sign. So if the anchor name that you gave something is 'dizzy', you will give it a 'Link URL' of '#dizzy'.

Formatting Styles

It is important, in websites, to keep a consistent look or brand throughout the site. The 'theme' for this site takes this into account, and has pre-selected different font, font size, weight and color combinations which help tremendously in keeping the consistency of the site.

To use these formats (and you should definitely decide to) you can select your text and then click on the box that says '--format--' and choose one. The way it gets formatted in your edit screen might be different than what you will see in on the actual saved page, so you'll want to take that into account if that is the case.

Cutting and Pasting

Microsoft Word and other word processing programs tend to embed the formatting in code on the document itself, which often has negative effects on the look and consistency of the formatting styles. Before you paste it into drupal, generally you first need to paste it into a program that will automatically strip all formatting like Text Editor (Linux), or Notepad (Microsoft), etc.

There are the normal cutting and pasting icons in TinyMCE and also icons for cutting and pasting from word, but there is also a pasting option with a Microsoft emblem on it that strips that formatting for you, and then pastes it into the field.

Images

Images have to be created before they can be posted to a page. Images are, like pages, content types. To create or upload an image to your site, you navigate to the 'Create Content' page of your site, and select 'Image'. You will upload the image from your computer, give it all the necessary names, descriptions (if you wish), etc, and then click submit.

Images should all be in a .jpg or .png format (which are web friendly), and they should also be close to the size that you will want them to be when you post them on a page or other content type. THEY SHOULD ALSO BE LESS THAN 1MB in size, and preferably less than 500KB. The reason is that pages with larger pictures take FOREVER to load. Web surfers are pretty impatient, anyhow, and you do not want them to leave your site before a page even loads...

Once you have create your image as a piece of content on the site, you can post it on a page.

> With TinyMCE, you will see the blue/black image of a camera as one of the icons on the menu. First place the cursor in the spot where you want your image to go, and then you will want to click on that icon.
>
> Learning this tool will take some practice. When you click on the camera, you will get a popup window that lets you select the image. If the image you want is not there, it means that it did not get created. Once you select the image, the popup window's content will change. This is where you really need to start paying attention.
> In the 'Title' you will see the title of the image. You will want to erase it. If there is any text in the description, you will want to erase that as well.
>
> Where it says 'size' it defaults to 'thumbnail'. You will want to choose 'original' instead. If you were to choose 'Other' , the original size (in pixels) of the original image will appear. You can change this, but remember that there are two dimensions, and that you will need to do the math. It's not a matter of decreasing both dimensions by 50 pixels. You will need to change the dimensions proportionately, as they will not change automatically in a proportioned way.
>
> Next you want to set the 'Alignment' of the image. This is not always going to seem intuitive – so you'll want to experiment with cursor placement if things do not happen as you expect them to.
>
> Configuring the link is somewhat intuitive. Generally, you should go with 'no link' unless there is a clear reason not to do otherwise. The option to 'link to image page', unless you are using the 'Images' content type as a gallery or some sort of silent auction or display catalog or educational manual, is a little cheesy.
>
> For insert mode, you always want it to be HTML. Click 'insert' and you are good to go.
>
> Again, the thing to keep in mind with images is that they might take a little practice and experimentation before you really get a sense of the tool.

Searching for content

When you create a content item and you do not give it a menu setting (or a URL), once you save it and go to another place on the site, it can seem like the item you just created is gone.

It's not.

Remember that drupal is built on a database, and that database records everything that happens. There is a way to search for items on the site that you swear are there but can't seem to locate now. Go to 'Administer → Content management → Content'. There are 3 main things you want to pay attention to and understand on this page: a filter to help you refine your search, a full list of all content (automatically sorted by time updated), and in between it an action drop down so that you can bulk change content in different ways.

The Filter has 3 categories: Status, Type and Category. The 'Status' filter refers to it's publication status, and variations of how drupal chooses to display certain types of content and to whom. If something is unpublished, it will not be visible to unauthenticated users, if it is promoted in a list, then when there is an rss feed or a view that is created without any other automatic sorts, it will stick or rise to the top of that list...etc. Type refers to content type. Category refers to the different categories/taxonomy terms that you have on your site.

Update options allow you to bulk modify status, url aliases and mass delete content items. The list of content that the filter has found is below the Update options, and there is a checkbox for each item so that you can choose which items you want to update.

SCHR Content Types

There are three types of content on your site: core content types that come with drupal, module content types that are created when certain modules are installed, and custom CCK content types that were created specifically for your organization. If you have the CCK module, any of the content types, regardless of type, can have additional fields added to them.

Core Content Types.

Page: A page is the simplest content type. It is the one that most people think of when they think of a web-page. The lions share of a typical website are pages.

Story: Story is like for a news story or article. It has a 'description' in addition to a 'body' field. The description field is modeled after the first paragraph of a news article that acts as an abstract summarizing the article. It is automatically bolded.

Blog: A blog is like an editorial or prose written by staff or stakeholders on the site. The mechanism in drupal for managing blogs organizes them automatically by author (user). A blog is not intended to create a discussion among various users (like forums), it's content is typically more personal, and about the personal perspective of the user/author. Generally, if blogs are being used, the exist within their own menu structure in which users can read their own blogs, create a new blog, or see a list of blogs.

Forum: a forum is intended and geared for discussions among a broad population of users. Forums can have permissions levels associated with them. They organize themselves hierarchically, and that hierarchy can be set up in the 'Administer → Content management section of your site if Forums are installed. The idea of a forum is to create a hierarchical list of subjects that people on the site can create entries into. Each entry can be responded to, creating a 'thread' so that discussions can be followed easily. As people respond to different initial responses, sub-threads are created. Forums are typically used for sites where networking and/or organizing/work needs to happen amoung a group of people who might not be geographically linked. The forum requires a very active user base, and should not be implemented unless that is the case. If you still want to enable responses to certain content, you can install the 'Comment' module (more below) which will enable the creation and viewing of comments and responses. Comments are enabled for specific content types, and also for specific users. Users are given permissions in 'Administer → User Management → Permissions'. Comments are configured in 'Administer → Content Management → Comments → Settings' and associated with certain content types in the content type itself which can be edited by going to 'Administer → Content Management → Content Types' and clicking 'edit' next to the specific content type in question.

Poll: There are sometimes instances where you will want to get feedback from certain types of users. A

Poll is the easiest way to do that. Polls are written by asking an initial questions, and then giving multiple answer choices for people to select. They come with a mechanism that, for folks with permissions to administer them, calculates statistics about how people are voting. Polls can be given menu space, a URL/path, and can also be linked to from the text of another content type...

Comment: A comment is a core content type that can be associate with other content types by editing the content type itself from 'Administer → Content Management → Content Types'. Enabling comments for a content type (and giving certain user roles permission to comment) can create a great forum for folks to either comment on content before it is published, or to respond to certain posts.

Module Based Content Types

Event: An event is anything with a start and end time. The event module comes with a lot of great features including a whole calendaring function where different calendar views can be created. It is a great module to keep track of organizational events, as well as internal meetings etc..

> In the case of SCHR, events populate a section of the site under '/action/attend'. As with the cases and resources (spoken about above) there are various blocks that have been created for specific events that enable the purchase of tickets/registrations/etc for that event.
>
> When you create an event, you want to give it a URL/path of '/action/attend/<event_name>'. If when you create the event, you choose for it to be 'published' it will automatically populate a list (created by a view) at the URL/path '/action/attend'.
>
> SCHR does not have enough events that are published on the site to make the use of any of the event calendar functions necessary, but if in the future that changes, those features of the events module can be configured.

Image: The image module is what allows for the uploading and treatment of images on the site. There are sometimes other modules (Image Assist, Image Galleries, etc) that can augment and change the way that images can be used and managed on the site. These can be configured under 'Adminsiter → Site configuration'.

Flash: Flash is a content type created by the 'Media' module, and is used to house flash files for use later in different parts of your site. This is a useful content type if you are selling advertising on your site, and some of the advertisements are moving flash movies.

> On the SCHR site, there are flash movies incorporated into all of the headers. They were placed in the header by creating a block for the 4 movies that are uploaded. Each of these blocks are configured to show on (different) certain pages and also to appear in the header section.
>
> This placement of these blocks is heavily wired, and formatted in the CSS (theme) code for the site, and so you want not to mess with this very much in terms of moving it away from the header area, deleting it, or creating a new one. This will require a developer with CSS knowledge.

Webform: Webforms are perfect for filling out applications or registrations. In a webform, you create a variety of specific questions that can then be answered by different users. The built in functions of webforms allow you to see responses in a list, export them into a .csv (that can be opened with spreadsheet software like Excel), and delete them. The workflow is usually to export them, and then delete them, so that the next time you export them, you do not have duplicates.

SCHR uses webforms to keep track of volunteer applications. You can access that webform from '/action/volunteer'.

The interface for managing the form is only visible to authenticated users. The entry point into administering a webform, is the same as creating it. To administer a webform, you would go to the webform itself as if you were going to submit an entry, and then the tabs for administering it will be visible to users with access.

Product / Product Kit: These are content types used in the Ubercart module, which is the module we use to process payments for event registrations. More will be said about them in that section of this manual.

CCK Custom Content Types

CCK is a module that is used to create and append (as in add to existing) content types. Content types can be created or edited with CCK by going to 'Administer → Content management → Content Types'. It is not within the scope of this manual to fully describe how CCK works. You should definitely take a look if you have it installed on your site. By looking around and experimenting you can usually figure it out intuitively, especially if you already have experience with databases. You should know that you can CCK modules that enable the creation and use of field types like calculation, boolean, taxonomy, date and other types of fields. CCK is a little intimidating, but also incredibly powerful. It's worth getting to know better.

Resource: the Resource content type was created to keep track of media articles, press releases, and schr papers that get published. It incorporates the 'Subject', 'Case', and 'Front Page' categories/taxonomies that are used to populate the items in different views and blocks created by those views.

FrontPage determines if you want the Resource to be posted on the front page of the site. There are only so many slots for the front page, and they are sorted in order of their respective creation/publication dates. So when you create a new 'Resource' and click the 'FrontPage' category, it will displace one of the older items, and place the one you just made at the top of the list. That front page “In the News” list is, itself, a view that generates a block.

'Subject' is also associated with views and blocks that publish specific resources at the bottom of the content of different pages of the site. If a new 'Subject' is added, and you want for it to appear in a block of it's own, or to show up in an existing block, you will need to create/modify the view and/or the block as well. 'Case' is wired in this way as well. For more on how these both come together, see above the section called 'Views/Blocks/Categories: a Case Study'.

Job: a job is that thing that you go to, and try to pack a lunch for, but never quite end up getting it together - and so you end up spending a lot of money buying lunch from the bagel shop down the street. It is also a content type created custom for SCHR that manages the posting of career opportunities on the website.

When you go to '/about/jobs' you will be taken to a page that was created by a view. This view populates a table that houses all of the 'Published' jobs. When there are published jobs, the header of the view reads differently than where there are no jobs published. When you 'un-publish' a Job, it will still exist in the database, and can be edited and/or re-published at a later time.

Donations

Although SCHR has an ecommerce module installed, a decision early on was made not to process donations using this module. Donations are processed through passing HTML forms to the USAepay transaction gateway. If you do not fully understand what I just said, then it's best to allow a developer to make any modifications needed. Editing the form itself doesn't require entry first into the HTML screen (from the WYSIWYG). However, if the page is altered using the WYSIWYD, the backend coding will likely be jeopardized as every form box is tied into the USAepay backend code in a very specific way.

SCHR takes both regular and recurring donations, which is represented on the Donation form on the website.

There are custom fields that have information needed (namely about in honor of or in memoriam donations). USAepay is doing work on having those fields show up in the merchant receipt that we get so that we do not have to look at each item from the USAepay terminal separately. Hopefully, by the time you read this, this work will have been completed.

USAepay

USAepay is the gateway (we are our own merchant) that we use to process all credit card payments. They have their own terminal where transactions can be viewed, batches can be processed, charges can be directly entered, and all sorts of settings can be modified. The website is: usaepay.com. From that page, you want to click the blue 'Merchant' box on the upper right hand side of the screen. You will be asked for your username and password. For security reasons, that password and username will occasionally have to change. It is a good idea to make sure to keep that information in a safe place, so that if it gets lost or changes or you forget you have a reference.

This manual is going to go briefly into the different sections of the Merchant screen for USAepay. They have a wiki where there are FAQs, manuals, and tutorials. They also have a support team.

Home: there's no place like it. There is a link here to a USER GUIDE, and then other information that you would sooner call support than read. It's dry and not that important.

vTerminal: This tab enables you to input transactions manually without having to go through the website. It's not sufficient for what we need, has a lot of fields that we do not use, and does not allow for recurring donations. It's easier to just go through the website.

Sale Form: This is a little more useful than the vTerminal (more fields that we would use), and has basically the same function as vTerminal, but again does not allow for recurring donations. Easier to use the website.

Customers + Billing: This is another section that is geared more towards the small business owner making transactions and isn't very useful to us.

Batches: This is a very important and useful tab for us. Batches can be set to automatically close. It will only close if there are items in it. That is setup on the 'Setting' opening page. To process a batch, you would click to close or process the batch. This area also enables you to VOID certain transactions. To void a transaction you want to click the checkbox under the 'void' column, and then click the 'Process Changes' button below the table.

> Currently, transactions coming from donations can be identified as they will not have a Flag associated with them (as in the American Flag). The transactions coming from the ecommerce function on the website will have flags.

Reports: This screen has bunches of reports you can run. Just click on the report title, and it will do it's thing (sometimes with input necessary from you). It's pretty intuitive. You can also build custom reports by clicking that button at the bottom. This tool might actually be useful as it enables reporting on some of the custom fields that we've built.

Search: this enables you to search for different transactions based on lots of criteria, including (hypothetically) sutom fields. That function, however, seems broken (it returns no results and an error). Wildcards like '*' do not work.

Settings: The opening screen for this section covers general setup functions like who reports are sent to, and how often batches close, setting regarding receipts, and also setting regarding how the system handles recurring billing. In the Users tab, people who have administrative privileges can add users and also edit existing ones. It also enables the changing of passwords for users (you do not have to know their old password to do this unless you are changing the main administrative password [for the 'southctr' user]).

> **Source Keys**: These are important because they tell you where transactions are coming from. In the case of SCHR, there are two 'sources' for transactions: Donations, and registrations. Because registrations are all managed by the Ubercart module, they all need to have the same 'Source Key'. Ubercart does not allow for multiple source keys.
>
> If you click to edit a source key, you are able to customize a lot of different kinds of settings including how the transaction page looks (relevant only to Donations, as the ecommerce module never actually shows a USAepay page). Some of the settings require knowledge of HTML. Please feel free to look around but be careful – especially with the Event Registration source key, as all of the settings for that key are managed by the Ubercart module on the site.

Fraud Center: This allows you to add (mostly security) modules for the different source keys. It's not super relevant to us at this time, but might be if we get burned in the future...

Ubercart

Ubercart is the module that handles ecommerce. It has a wide range of modules that function together to process payments. They have online documentation (very thorough) at: http://www.ubercart.org/docs/user

We use Ubercart version 1.x which is the version that works with our version of drupal (5.x). While Ubercart works with content management systems other than drupal, it works best with drupal.

Including simple ecommerce functions, Ubercart also integrates:

- Selling physical goods from various sized product catalogs.
- Selling file downloads (i.e. music, videos, software).
- Selling site access for members only websites, including automatic renewals and expiration of user

access.

SCHR does not have any of the more complex functions like selling site access and file downloads, but they can be set up if that changes in the future.

Because Ubercart's documentation does SUCH a great job of going through all the configuration settings, my goal in this section is to briefly explain the salient characteristics of your site specifically.

Documentation:

- Ubercart 1.x - http://www.ubercart.org/docs/user/7436/ubercart_1x
- Ubercart 2.x - http://www.ubercart.org/docs/user/7437/ubercart_2x

Site Specific Settings

Please go to 'Administer → Store Administration' and look around. This is the best way to familiarize yourself with this tool. Snoop around as much as you can. The follows are tips and tricks that are used on the site.

General Settings

In the 'Administer → Store administration' section, when you press on an actual Title (like 'Products' or 'Customers' you will be get sent to the list of them. When you click on the 'Show links' word below those titles, it will expand the menus to configure the items.
When you click on the actual title to see a list of those items, there are options as tabs above that list, and also to the left hand in the menu section under 'Administer → Store administration'.

Adding a New Event Registration/Product(s)

An event is a content type that is managed by the event module. To create a new one, you go to the 'Create Content' menu item and choose 'event'. Please see the section above for more information about this. Also, please follow the naming conventions used for the other events (action/attend/<eventname>) and also place the event in the same menu hierarchy as the others. The event will automatically show up on the '/action/attend' view if it is published. If you no longer want it there, unpublish it. DO NOT DELETE IT.

Once the event page has been created, you will need to create a new catalog. The catalog that a product is associated with tells the view/block that displays all the products for an event (under the event information on the event page) which products to select. So to make a new catalog go to 'Adminster → Content management → Categories' and add the new catalog to the 'Catalog' category as a term.

At this point, you can add products. To add products, go to 'Administer → Content management' and look on the left hand navigation under 'Content Management' for 'Create Content → Product'. The configurations for the products are pretty self explanatory. If you want to use a cheat sheet, look at another product first, and then copy what you did there.

Displaying Registration Information on Event Page

The site uses blocks to display certain registrations and materials under the actual pages of the event itself. These blocks need to be created for each event. The block is created by a view, and then is configured to show up in the 'content' area by activating the block from the block interface. An example is the 'mc4l_product_list' which is a view and also a block. You can look at these and replicate them by cloning and editing the view, and then activating the block.

Adding Attributes to a Product

When creating a registration product (example: Voir Dire Training) that requires additional information (like the CLE Bar and State certification information) you can add this 'attribute' to the specific registration/product. These 'attributes' are created at 'Administer → Store administration' and then clicking the 'show links' under 'Products' and clicking on 'Manage Attributes'. I have an example for the Voir Dire Registration, and you can model it after that. Once that is created, you can associate that 'Attribute' with a specific product by going to the product and then clicking on the secondary 'Attributes' tab from inside the Edit screen.