

## Start Point

1. Installed 7-Zip
2. Install WAMP Server
3. Turned on rewrite module (WAMP Server Apache/Modules menu)
4. Edit php.ini file options
  - increase max\_execution\_time = 180
  - memory\_limit = 128
  - post\_max\_size = 24M (for phpMyAdmin needs to be larger than your backup file)
  - upload\_max\_filesize = 10M (may need higher for video uploads, but you may have issues with script timeouts)
  - close, save and restart all services
5. change my.ini to myisam as default??
4. Move drupal install file to www folder (c:\wamp\www)
5. Right click "Extract Here" (extract from the gz)
6. Right click "Extract Here" extract from the tar
7. change directory name to "ournews"

## Validate database username/password

Assume root/no password, but check before installing drupal

1. Open phpmyadmin as per your server environment
2. Click "privileges" page for root with no password
3. Record your root password if you have one (I think it is root/root on MAMP)

## Before Installing Drupal

1. Create a default settings.php file using the default.settings.php file provided.
  - Change starting in 6.3(?) for security reasons I presume
2. Make sure the file is writable
3. Go back to phpmyadmin
4. Create a new database called "ournews", collation "utf\_general\_ci"

## Install Drupal

1. Go to localhost/ournews – this should go to the install page
  2. Install in English
  3. Setup database
    - ournews
    - user=root
    - password=blank
    - click advanced, don't change anything
  4. Configure Site
    - Site Name = ournews
    - Site email = noreply@ournews.org
- + Administrator
- Username = admin
  - email address = webmaster@ournews.org

- password = admin

- choose time zone

- make sure clean urls is on

5. Note email error. WAMPServer doesn't support mail so we will work around that

## First Content (About Page Placeholder)

1. Click the "Create Content" link #4 on the welcome page

2. Select "Page"

*briefly discuss the two out of the box content types*

3. New Page information

Title = About Us

Body = Placeholder for about us page.

4. Describe the basic options briefly

- Input Format

common source of errors for new users. Will demo in a moment

- Revision Information

Can make revisions so you can roll back to old content. May not want this for all content types.

- Comment Settings

Obvious?

- Authoring Information

blank for anonymous. As admin you may want ability to post as someone else.

May not want that for users

- Publishing Options

briefly explain, will see as we go through

Point out that what user sees may depend on their permissions. Since we are "user 1" right now we'll see all options when creating content

5. Make sure to promote it to Front Page.

6. Don't change any other options for now and click Preview

6. Now throw a bunch of junk text in there to make it long enough to have a different teaser.

7. Preview to demo trim version.

8. Use the split option to split it much further down and preview – note the much larger preview.

9. Remove extra text to leave it as "Placeholder for About Us page.". Then click Preview, then Save.

10. Click the Drupal Icon to go to home page – note content on home page now and no more default message.

11. Now logout. Note no menu and only a login/register box along with page content.

12. Log back in.

## Tour of Drupal Admin "out of the box"

13. Quick tour of Admin menu options with no additional modules enabled yet.

- Clean URLs

- File system (show them on their hard drive where Drupal created the files folder. Must be writable by server process)

The plan for our website is to create a simple newspaper website for a High School. Newspaper style websites are very common and show off a number of features of drupal such as story publishing, comment support, blog support, image support, categorization/organization (taxonomy), user privileges, etc. So what is our specific plan...

- Create a basic about page so we can get rid of the default start page

- Set up a simple taxonomy

+Sections: News, Media, blogs, events??

- Create some backend pages such as Staff Leadership, Mission, School Song

- Create and setup a contacts page (may not work on their machine)

- Create "About Menu" – menu basics

- Input Formats (talk about later)
- Performance (talk about how standard caching works – good for news site where most users are anonymous)
- Describe briefly what other options do (and which might need to be turned off when developing a site/theme)
- Site Maintenance
- Site Information

14. Modify Site Information as follows:

- Mission = Our Mission at Drupal High School is to develop the next generation of leaders – Drupal Leaders!
- Footer Message = "&copy; Drupal High School – All Rights Reserved"
- Note that here can be changed the default home page to any path (could be a specific page, or view or panels if using those modules)

15. Go briefly through admin/content section. Note that you can see/administer comments and content. Types and Taxonomy we'll talk about later.

16. Go to Post Settings.

- Note the number of posts on home page. Set it to 5 only.
- Note the length of trimmed posts option (note that it is HTML not non-html characters!)

17. Briefly go through Site building menu

- What is a block
- What is a menu
- What is a module?
- What is a theme?

18. Go through User Management Menu

- Permissions (we can create roles, new modules and content types will show additional permission options so always go back here after adding modules or new content types)
- What are roles?
- Users (compare to ability to manage content – here ability to manage/edit users)

Create first real "Story" and demote first page we created above (published, not on home page)

1. Create Content Menu

2. Story Type

3. Enter as follows:

Title = Welcome Returning Students!

Body = <p><strong>Principal Dries</strong> here!</p>

<p>Welcome to a new school year at Drupal U! We hope you are looking forward to the new year just as we are. Check back soon for more news and information. We'll have information for students, teachers, and parents.</p>

<p> - <em>D.B</em></p>

4. Point out slight difference between the two content types. One shows date/time and user posted, the other not.

- Page types are meant to be static pages rather than "stories" or other transient content.

- Therefore page type defaults to not showing those (assumption = permanent/not time bound)
- You can create more than these two types and for each type you can control these options.
- 5. Go to "Site Building"/"Themes"/"Configure"/"Global Settings"
- Odd place but "Display Post Information On" allows you to select which content types display this.
- 6. Edit About Us node
- take off home page

### So Where Does Content Come From

1. We just saw pages and stories. We'll see later we can create additional content types with additional fields. But where else does "content" come from
2. Content (in the broadest sense of anything visible to the user) can be...
  - Menus
  - posted content (stories, pages, other types we'll make)
  - "Theme content" things like site title, footer, site slogan, others
  - "Blocks" provided by modules. Example is the User Login block displayed when you go to site without logging in. If you install new modules they may add additional blocks that you can use! Let's take a look.
3. Admin/Site Building/Blocks
  - Note the items already there in the default positions as placed by Drupal install. These blocks are all supplied by core modules. Note there are a few that aren't placed anywhere yet. Let's place a couple...
4. Place "Who's Online", then place "Who's New" on right sidebar.
5. Try dragging them around!

### Enable a Core Module to get new page content

1. Admin/Site Building/Blocks
  - Expand Core/Optional
  - Enable Poll Module
2. Stop a minute – explain permissions
3. Admin/User Management/Permissions
  - Note only anon and authenticated. (User1 has all permissions and doesn't show up on this list!)
4. Find Poll Module permissions
  - These permissions were defined by the module author to allow you as a site developer/designer to have more control
5. Enable vote on polls permission to authenticated users
6. Go to Admin/Site Building/Blocks
  - Note the new block for "Most Recent Poll"
7. Put "Most Recent Poll" block on top of right sidebar
8. Back on home page note there is nothing there yet

- We haven't created a poll!!

9. Choose Create content

- note new content type poll. Some modules create new content types as part of their install.

10. Create a new poll as follows:

Title = What is your favorite class this year?

Choice 1 = Drupal Intro

Choice 2 = Theming 101

Choice 3 = Ninja Development

Published, but not promoted to front page

11. Save

- Note that on the home page the poll now shows up on the most recent polls block

12. Submit a response

- Note how the poll shows a summary of the responses.

13. Click on the older polls.

- Note that the module automatically provides a way to review published polls/results.

This is OLD School! (See CCK/Views)

### Intro to Taxonomy

Taxonomy is a big subject. Basically it is a way to categorize or tag the content as you submit it. It can be used in many ways. We'll use it in two relatively simple ways for this site. Taxonomy is composed of vocabularies. If your site were about computers you might have one vocabulary for types of computer (laptop, desktop, server), another for default operating system (MAC, Win, Unix, Linux), etc. You may want some vocabularies to be single select, others you may want to allow the user to choose more than one (a desktop that can run either Linux or Windows). You may even want the user to define their own terms (free tagging like on Flickr).

### Create the section taxonomy

1. Admin/Content Management/Taxonomy

2. Select "Add Vocabulary"

3. Fill in options as follows

Vocab Name = News Section

Description = Which news section should this story be published in?

Help Text = Select a news section for this story.

Content Types = Story

Settings =Required

Weight = -10 (explain later)

4. Save

### Now add terms

1. "Add Terms"

2. Fill in as follows

Term Name = Sports

Description = Sports news as Drupal High - keep up with the "Blue Drops"!

Look at Advanced options, but don't change

3. Note that it automatically presents another term add screen. Add additional terms as follows

Term Name = Scholastic

Description = Learn by the Drop - hey it's why you're here!

Term Name = Clubs

Description = The extracurricular side of Drupal

Term Name = Arts and Music

Description = Drupal Drupal Drupal Drupal Druuuupal...

Term Name = General Announcements

Description = To all announcements

4. Go back to list terms

- Note that they can be manually dragged and dropped or reset to Alpha

5. Drag "General Announcement" so it is at the top of the list.

6. Go back to Taxonomy, note that Story is shown as the content type associated with this vocab (we can associate other content types later if we want)

Create another taxonomy as follows:

1. Details as follows

Vocabulary Name = Tags

Description = User tagging of content

Help Text = Enter a comma separated list of tags to make it easy to find similar stories later.

Content Types = Story

Tags = True

Multiple Select = True

Weight = 10

Edit the "Welcome Back" story to see the effect of the taxonomy

1. Edit the Welcome Back story

2. Note the two items in vocabularies section. Modify as follows...

3. Select "General Announcements"

4. Save story

5. Note the link to "General Announcements" on the bottom of story. Any terms selected for any vocab (or free-tagging for that matter) will show up here. If you click it shows all content (of any type) with that term on it in a teaser style list. Note the browser address and that the term name and description shows up at the top of the list. Not very impressive for one story!

Install Devel Module

Hmm. It will take quite awhile for me to have you create enough stories to have a real website that will look complete won't it? Is there some way we can create a bunch of

stories quickly? Well actually yes and that brings us to an important supporting module call devel. What is devel? It is designed as a set of utility functions to support developers – both module and theme developers. I don't pretend to understand everything it does for sure. But one function we could really use right now would be to create a bunch of stories to populate our site! Let's install Devel Module. Here are the steps to install modules

- download latest stable version from drupal.org
  - Extract it to the sites/all/modules folder
  - Go to the module list in Drupal admin
  - Enable the module
  - Set permissions provided by the module if needed for diff classes of users
  - Configure module features, turn on blocks desired (if any), etc.
1. Copy devel module to sites/all/modules
  2. Extract Here, Extract Here
  3. Enable module on modules page (Devel AND Devel generate)
  4. Go to admin/by module
  5. In Devel module settings change SMTP library to "log only"
  6. Go to Site Building/Blocks/List
    - Note the new blocks made available by devel module.
    - Add the "Switch User" block to the bottom of the right sidebar (we'll need it later)
  7. Go to Administration menu.
    - Note the "Generate Items" menu item.
  8. Click generate items/generate content
  9. Following options
    - check only story
    - Make 50 nodes
    - No comments
    - Check "add taxonomy terms to each node"
    - Do It!

Understanding what happened and what we see.

- It used mostly "junk" text (random latin) to create the story nodes
- For the required taxonomy it randomly assigned taxonomy terms
- How many of the 50 appear on the home page?
  - + Less than 50 as some were promoted to the home page, some not
- How do I get to the others?
  - + Click on a taxonomy term. You'll see a list of all articles that include that term in reverse date order INCLUDING the ones that were not promoted to the home page (all were published)

What happened to my "Welcome" story? The home page is sorted in reverse alpha. But we can force it to the top!

- Edit Welcome story and set the "Sticky" option
- Explain how the sticky option works

So what about menus?

1. Log out as Admin
2. Note that right now we have no menu at all! Just a log in block. We have some ability to navigate due to the taxonomy we created, but other than that it is pretty bare. What if we want there to be menu options to help direct the user to various things on our site such as...
  - Static pages of various sorts
  - Search
  - Contacts page
  - About us page(s)
  - offsite links, etc.
3. For now just so we can see how menus work we'll create two. An "Offsite Resources" menu and an "About Us" menu.

Create a menu directly from editing an item

One way to create a menu is to assign an article (node) that you create to a menu at the time you create or edit the node. You can put this on a specific menu as a top level menu item or a submenu item. Let's first go see what menus may already exist on our site "out of the box"

1. Log in as admin/admin
2. Note right away that you now have a menu on the left side (My Account, Create Content, Administer, Log Out). This is called the "Navigation" menu (personally I find that odd). A good way to think about the Navigation menu is as a menu that most won't see unless they create an account, log in, and have permissions to create some kind of content or do some kind of admin. Because you are logged in as User 1 you see all admin menu commands.
3. Now go to Administer/Site Building/Menus
  - Notice that we have other menus as well (Primary, Secondary, Devel).
  - Primary and Secondary are automatically there for any Drupal install. If you click on them you'll see that they are empty by default.
  - Devel on the other hand was created by you installing the Devel module earlier. This raises an important point. Modules you install can affect menus, either by creating whole new menus or by adding commands to existing menus. This is good thing to check for after you install a module!
4. Click on "Settings"

Note the three options. The first option won't make sense till we use it so I'll hold off on that. The other two options allow us to define which menu is the "primary links" menu and which is the "secondary". You don't have to stick with these choices. You can create your own menus and put them here. Or you can just add commands to the existing menus. What is the effect of these choices? Well many (most) themes will be designed to put the primary and secondary menus in specific places on the screen. So by assigning them here you control which menu goes in these predefined places. We'll see where Garland (the default theme) puts them in a moment.

5. For now, leave them as is.

Now we'll add an offsite resources menu to by creating a new page.

1. Click Create Content, then Page

2. Title = "Offsite Resources"

Body = "<ul><li><a href='\"http://www.drupal.org\">Drupal Home Page</a></li><li><a href='\"http://groups.drupal.org\">Drupal Groups</a></li></ul>"

3. NOW, click "Menu Settings" and set as follows

- Menu Link Title = "Resources"

- Parent Item = expand to see, but pick "Primary Links"

- Weight = 0 (Explain weight)

4. Under publishing options confirm that it is published, but not promoted to the home page.

5. Go to the home page

- Note the new menu item on the top right

6. Click on it to go back to that page.

7. So to summarize, as you are creating some of those "backbone" static pages you may need you can assign them a menu command right as you are creating or editing.

Let's go fish out our about page and do the same. But wait, how do we find it? It isn't published and we don't see it on the home page anywhere...

1. Go to Admin/Content Management/Content

2. If there are any filter options click reset

3. We know that "About us" was a page so filter content type for only page content

4. Click Edit

5. Follow the procedure above to assign the menu to the primary links with a menu title of "About Us"

6. Hey wait (weight), we want About Us to always be last on the menu. What do we do?

7. Go back to edit it. Change the weight to 50 (heaviest). That ensures it will "sink" to the bottom of the list. Items with the same weight will be alphabetically sorted.

Now let's add a second level menu. About Us would often have a number of things on most sites. Lets add some submenus here. We'll do it by adding two pages.

1. Create a new page

2. Set as follows

Title: Staff

Body: Dries, Chx, merlinofchaos, etc.

Menu Settings:

- Menu Link Title: Staff

- Parent Item: About Us (below primary links)

3. Save

4. Go to the home page.

5. If you click on About Us it doesn't "drop down" a submenu or anything. What's up with that?

Well you could install the nicemenu module which converts a multilevel menu into a nice dhtml menu, but that is out of scope for us today. We have two choices. One is to simply display our menu in a block instead of using the Primary Menus setting. But let's try one more thing here...

1. Go back to Admin/Menus/Settings
2. Set the source for both primary AND secondary links to "Primary Links" and save
3. Now click on About us on top. Note how the About us page appears AND a staff menu item below it appears? Click on that and you get the staff page

Not very user friendly huh?

This is really a factor of the theme. Garland is not really designed with using the primary menu for the secondary menu option so the display of second level menu is not too good. Other themes are better for this. But you can always use blocks to display a full menu and there are many other ways to work with menus beyond the scope of this presentation. One more thing...

1. Go to block admin
2. At the bottom notice there is a block for each defined menu?
3. Put the primary links menu in the left sidebar above the navigation menu.
4. Save. Note the new block.
5. Log out. Notice that this will now always be visible even to anon users. And the menu looks a little more natural and hierarchical.

We'll leave it in both places for now.

Let's go back to Create Content and Create a New Story

1. Log in if needed admin/admin
2. Create Content Story
3. Click in the body. How do you make something bold or italic. Indent? Bullets? HTML?? Will your users know how to do that?

One of the most common things folks may want from a CMS is a way to do at least some formatting without knowing html. WYSIWYG. It is definitely a double edged sword. Used too freely and you could end up with a lot of junk! But let's try one anyway...

Add fckeditor

1. the fckeditor uses the open source WYSIWYG editor called fckeditor. The editor is not GPL so it cannot be included in the module download. It has to be added separately.
2. Go out to your hard drive/finder and copy the fckeditor tar file from the resources folder
3. Put it under sites/all/modules/fckeditor
4. Then extract it (in 7zip extract here, then extract here again)
5. You should then have another fckeditor folder under the module fckeditor folder. It has to be right there for the fckeditor module to find and use it.
6. Go to admin/site building/modules
7. Under other find and enable fckeditor

8. Try creating a new story. Note the editor that appears.  
Lots of options. Actually too many!

Learn about filters

Let's try a quick experiment

1. Create a new story
2. Put Test for title
3. Pick "General Announcements" for the New section.
3. For body
  - Put "Now is the time for all good men to..." on the first line
  - Put "...learn Drupal!!" on the second line

Make Drupal bold

Now highlight the whole second line and make it an h1

4. Promote to home page and sticky
5. Save
6. Hey, what happened to my WYSIWIG!!

You've just been caught by perhaps the #1 support question for brand new drupaller – don't feel bad happens to the best of us.

So what did happen.

1. Let's go back in to edit this node.
2. Expand the Input filters section.
3. See how "Filtered HTML" is the default and how few tags it allows.
4. If you switch to plain text view for the body you'll see that fckeditor used p and h1 tags for our formatting. It got the strong right – be wouldn't have worked.

Why is this important? Because you WANT filters so your end users don't clutter your site up with crappy html. But still you may want more than what is allowed there. Two options.

1. just make full html the default (try it save this as full html – works).
2. Or you can edit or even make your own filters.

Admin/Site Configuration/Input Filters

Show where you can modify and create filters and some of the default options.

But rather than do this I'd rather just make the fckeditor not show so many options. I won't get into all the ways you can do this. The fckeditor is very configurable and a full discussion is out of scope. But there is a simple way to force fckeditor to use very simple toolbars that should solve our problem

1. Go back to admin/site configuration/fckeditor
  2. edit the fckeditor advanced profile
  3. Expand "visibility settings"
  4. Put the following into "Force simplified toolbar on the following paths"
- /node/story/add  
/node/page/add  
/node/\*/edit

5. This is a quick and dirty solution. Read the fckeditor documentation for much more information on controlling the fckeditor toolbars or functionality.
6. Try creating a new story and note simple toolbar.

Ok, we've learned a lot so far. Stories, Pages, Modules, blocks, menus, taxonomy, WYSIWYG options, etc. But so far all our stories are just simple titles and text. Don't we want more dynamic content for our website. Pictures, vid, etc.

### Core Upload Module

1. Do a quick demo of using core upload module to insert a picture into a node.
  - Note UI problems for non-tech users.
  - What if you want a small image on home page (thumb), but when you go into full story you want large image and if you click on image one more time you want to see image full size? You'd have to upload three separate images!
2. Disable upload module

### What we really want

- Easy uploading of files
- Easily structured content. We don't want users to have to fight with HTML any more than necessary
- Automatic deriving of images (we upload an image, Drupal makes a thumbnail (to our specs), a mid size (ditto), and a large.
- And make it at least relatively easy to make it look the way we want it to.

The answer is CCK + Imagefield + ImageCache. Let's get it set up and see what we can do.

1. Lead them through installing CCK, Imagefield, ImageCache, Filefield, ImageAPI,
  1. Admin/Settings/Modules
  2. Expand CCK Section
  3. Enable content module and Save (I'm concerned about php timeouts if we try to do too many at once)
  4. Now Expand the image cache section. Enable ImageAPI, ImageAPI GD2, ImageCache, ImagecacheUI and Save.
  5. Finally back to CCK and enable the following: filefield, imagefield, nodereference, number, option widgets, text, user reference and save

OK, so let's talk about CCK. Use course example...

Now what we're going to do is simpler. We're just going to add a lead picture field to our story type to make it easy for users to add a lead picture in a user friendly and powerful way. Here is how we want it to work from the users perspective.

- They go to create a story
- There will now be a lead picture field

- They upload their image
- Drupal AUTOMATICALLY creates a perfect square thumbnail for the teaser view, a midsize scaled picture for the node display in addition to the full image.

First set up Image Cache so it can generate the images you'll need. In Imagecache this is called presets. We'll need two. One to generate small square thumbs for the home page teasers, another to create mid size scaled images for the node view page.

1. Admin/Site Building/imagecache
2. Add new preset
3. Present namespace = thumb
4. Add action "Scale and Crop"
5. Set it to 100 x 100 (not percent!)
6. Go back to image cache and make another as follows  
present namespace = node  
Scale action  
Width=250  
Height=blank (so it won't distort images)  
Check off allow upscaling (in case someone posts a really small pic)
7. OK, done with that.

Now let's modify the Story content type to have a new field for images...

1. Administer/Content Types/Story Edit
2. Add field
  - label = Lead Picture
  - name = nodeimage
  - field type = image
4. On next screen edit as follows:
  - Help Text = Pick a picture as the lead for this story
  - Max resolution = 0 (unlimited)
  - Min resolution = 100
5. Now drag "Lead Image" above body. While we're at it drag menu settings to the bottom.
6. Go to "Display Fields"
7. Set as follows
  - Label = <hidden>
  - Teaser = thumb image linked to node
  - Full Node = node image linked to image
8. Save

Demonstrate functionality with a new story

1. Create Content/Story
2. Fill in as follows  
Title: Dries Speaks!

News Section: Scholastic

Lead Picture:

- Browse and select from resources folder "driesszeged.jpg"
- Upload
- Turn off "list" option

Body:

Dries of "Drupal School" speaks to the global community.

3. Select promoted to home page and sticky
4. Save
9. Demonstrate functionality
  - Home page you get a square 100x100 thumb
  - Click on the thumb you go to the node and see a larger pic
  - Click on the pic in node and you get a full size

We'll talk theming in a minute, but how about me add our own "mini-gallery" for our story functionality...

1. Administer/Content Types/Story Edit
2. Add field
  - label = Story Picture Gallery
  - name = nodegallery
  - field type = image
3. Drag it under Body field then click Save
4. On next screen edit as follows:
  - Help Text = Select Up to Four pictures to display with this story
  - Max resolution = 0 (unlimited)
  - Min resolution = 100
  - Number of Values = 4
5. Save!
6. Go to "Display Fields"
7. Set as follows
  - Label = <hidden>
  - Teaser = <hidden>
  - Full Node = thumb image linked to image
8. Save

Edit the Dries Speaks Story as follows:

1. Add 4 pictures to the story gallery as follows
  - Picture 1
    - Select Picture: Driestv.jpg (then click Upload)
    - Description: Dries TV
  - Picture 2

Select Picture: drupalcry.jpg

Description: Drupal Cry

Picture 3

Select Picture: drupaloldschool.png

Description: Old School

Picture 4

Select Picture: Drupalmascot.jpg

Description: Drupal Mascot

Theming

Outside of Scope, but let's do a quick demo.

1. Copy Marinelli
2. Install It (Go to Admin, Themes and make Marinelly default and loaded)
3. Note radical new appearance of website, new locations of menus, etc.