

Putting It All Together

- Vagrant
- Drush
- Version Control

Vagrant

Most Drupal developers now work on OSX. The Vagrant provisioning scripts may not work on Windows without subtle changes.

- If supplied, read the README file for setup details.
- Check the host name + port supplied and setup your system to match.
- Make changes? **commit** them with Git.

Updating Your Git Repository

Last night Eaton made some changes to his Vagrant setup. Let's "pull" the changes to our computer.

\$ git pull

```
C:\Users\Emma Jane Hogbin\Workflow\vagrant-chef-dlamp>vagrant up
The host class is reporting that NFS is not supported by this host,
or `nfsd` may not be installed. Please verify that `nfsd` is installed
on your machine, and retry.

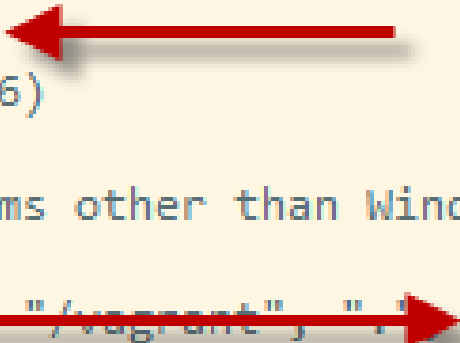
C:\Users\Emma Jane Hogbin\Workflow\vagrant-chef-dlamp>git pull
remote: Counting objects: 7, done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 4 (delta 3), reused 4 (delta 3)
Unpacking objects: 100% (4/4), done.
From https://github.com/eaton/vagrant-chef-dlamp
   3f20629..ac243bd  master    -> origin/master
Updating 3f20629..ac243bd
Fast-forward
 README.markdown | 2 +-
 Vagrantfile      | 13 ++++++-----
 2 files changed, 7 insertions(+), 8 deletions(-)

C:\Users\Emma Jane Hogbin\Workflow\vagrant-chef-dlamp>
```

Adjust the Vagrantfile Settings

- Two things are “broken” for Windows:
 - The port number.
 - The file mounting.
- Edit the Vagrantfile and change:

```
22 # Network setting for Vagrant >= 0.90
23 config.vm.network :hostonly, "33.33.33.10"
24 config.vm.forward_port(80, 8888)
25 config.vm.forward_port(3306, 3306)
26
27 # Try to use NFS only on platforms other than Windows
28 nfs = !Kernel.is_windows?
29 config.vm.share_folder("v-root", "/vagrant", ".") :nfs => FALSE)
```

Two red arrows are present in the code block. The first arrow points from the right towards the port number '8888' in the line 'config.vm.forward_port(80, 8888)'. The second arrow points from the right towards the file path and options in the line 'config.vm.share_folder("v-root", "/vagrant", ".") :nfs => FALSE)'.

Save + Commit Your Changes

- `$ git commit -m "Adjusting port + file mount for windows"`

```
C:\Users\Emma Jane Hogbin\Workflow\vagrant-chef-dlamp>git commit -m "removing the  
e win test and nfs load options"  
  
*** Please tell me who you are.  
  
Run  
  
    git config --global user.email "you@example.com"  
    git config --global user.name "Your Name"  
  
to set your account's default identity.  
Omit --global to set the identity only in this repository.  
  
fatal: unable to auto-detect email address (got 'Emma Jane Hogbin@beastie.(none)  
' )
```

To the file /etc/hosts, add your host name

Open

Computer > Local Disk (C:) > Windows > System32 > drivers > etc

Organize New folder

★ Favorites

- Desktop
- Downloads
- Dropbox

Recent Places

Name	Date modified	Type
hosts	7/24/2012 7:52 AM	File
lmhosts.sam	6/10/2009 5:00 PM	SAM File
networks	6/10/2009 5:00 PM	File
protocol	6/10/2009 5:00 PM	File

Vagrantfile

```
19 # Network setting for Vagrant < 0.90
20 # config.vm.network("33.33.33.10")
21
22 # Network setting for Vagrant >= 0.90
23 config.vm.network :hostonly, "33.33.33.10"
24 config.vm.forward_port(80, 8888)
25 config.vm.forward_port(3306, 3306)
26
27 # Try to use NFS only on platforms other than Windows
28 nfs = !Kernel.is_windows?
29 config.vm.share_folder("v-root", "/vagrant", ".", :nfs => FALSE)
30
31 config.vm.customize ["modifyvm", :id, "--natdnshostresolver1", "on"]
32
33 config.vm.provision :chef_solo do |chef|
34   # This path will be expanded relative to the project directory
35   chef.cookbooks_path = ["cookbooks/site-cookbooks", "cookbooks/
36   chef.roles_path = "roles"
37
38   # This role represents our default Drupal development stack.
39   chef.add_role("drupal_lamp_dev")
40
41   chef.json.merge!({
42     :www_root => '/vagrant/public',
43     :mysql => {
44       :server_root_password => "root", # TODO: Hardcoded MySQL
45       :allow_remote_root => true,
46       :bind_address => "0.0.0.0"
47     },
48     :hosts => {
49       :localhost_aliases => ["dev-site.vm"]
50   },
```

/etc/hosts

```
# localhost name resolution is handled within DNS itself.
#       127.0.0.1       localhost
#       ::1            localhost
127.0.0.1       videos.localhost
127.0.0.1       drupalcommerce.localhost
127.0.0.1       commercekickstart.localhost
127.0.0.1       og.localhost
127.0.0.1       teaching.localhost
127.0.0.1       drupalfundamentals.localhost
127.0.0.1       projectmanager.localhost
33.33.33.10    drupal.vbox.local dev-site.vbox.local
```

Start Your Vagrant Machine

- From the host machine (windows or OSX):
 - `$ vagrant up`
 - `$ vagrant ssh` (or putty -> `ssh 127.0.0.1:2222`)

Prepare To Install Drupal

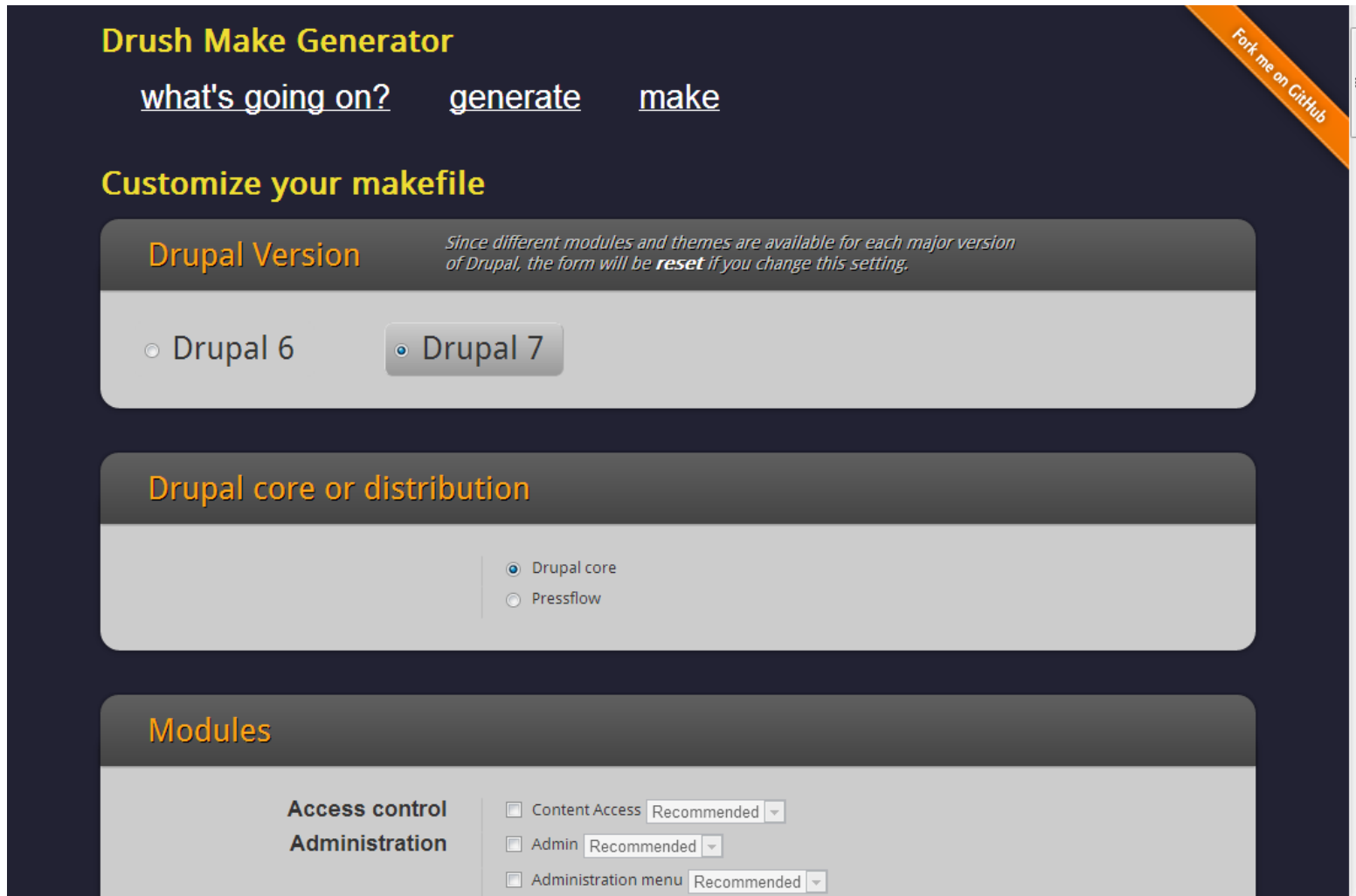
- From inside vagrant
 - `$ cd /vagrant`
 - `$ mkdir public`
 - `$ cd public`
- This is a SHARED folder, so you can do this step from your host machine as well.

Create a Database

- From inside Vagrant, at the command line issue the following command:
`$ mysql -u root -p`
- When prompted for a password, use: root
- At the new command prompt, type:
> create database drupal7;
> exit

Creating a Drush Make File

- <http://drushmake.me/>



The screenshot shows the Drush Make Generator website. At the top, there's a dark blue header with the title "Drush Make Generator" in yellow. Below it are three links: "what's going on?", "generate", and "make". A yellow banner in the top right corner says "Fork me on GitHub". The main content area is titled "Customize your makefile" in yellow. It contains three sections: "Drupal Version" with radio buttons for "Drupal 6" and "Drupal 7" (selected), "Drupal core or distribution" with radio buttons for "Drupal core" (selected) and "Pressflow", and "Modules" with a list of modules and their status. The "Access control" and "Administration" sections are highlighted in dark blue. The "Access control" section has a checkbox for "Content Access" (checked) and a dropdown menu set to "Recommended". The "Administration" section has checkboxes for "Admin" (checked) and "Administration menu" (checked), both with dropdown menus set to "Recommended".

Drush Make Generator

[what's going on?](#) [generate](#) [make](#)

Customize your makefile

Drupal Version *Since different modules and themes are available for each major version of Drupal, the form will be **reset** if you change this setting.*

☐ Drupal 6 ☒ Drupal 7

Drupal core or distribution

☒ Drupal core ☐ Pressflow

Modules

Access control

☒ Content Access Recommended

Administration

☒ Admin Recommended

☒ Administration menu Recommended

Download [Custom] Drupal With Drush

- Copy the Drush Make file contents.
- **Inside** your Vagrant host at `/vagrant/public`, create a new file named `simple-seven.make`. [hint: this folder is shared, you can do it from your host machine.]
- Paste the contents of your Drush Make generator into this file and save the file.
- From the command line inside Vagrant, run the command: `drush make simple-seven.make`

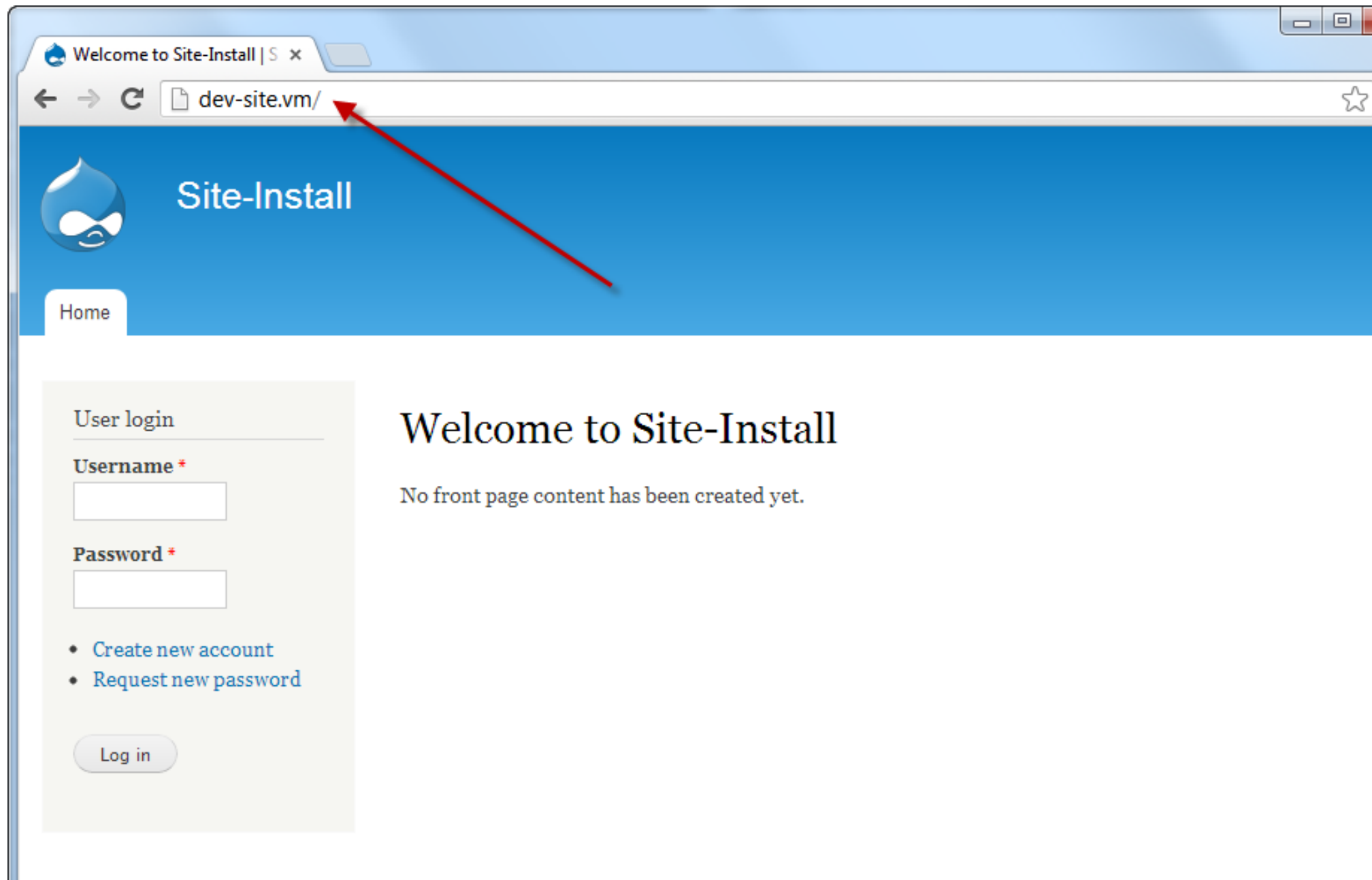
Install Drupal with Drush

- `$ drush site-install help`
- `$ drush site-install --db-url=mysql://root:root@localhost/drupal7`

```
vagrant@precise32:/vagrant/public$ drush site-install --db-url=mysql://root:root@localhost/drupal7
You are about to create a sites/default/files directory and create a sites/default/settings.php file and DROP all tables in your 'drupal7' database. Do you want to continue? (y/n): y
No tables to drop.
Starting Drupal installation. This takes a few seconds ...
Installation complete.  User name: admin  User password: fej7G7kM74
```

- Note the password after you have installed Drupal!

Visit Drupal in Your Web Browser



Installing New Modules With Drush

From **inside** the Vagrant machine, you can use Drush.

- \$ cd /vagrant/public/sites/default
- \$ drush dl <module_name>
- \$ drush en <module_name>

Updating Drupal Projects

This will also update Drupal core.

- `$ cd /vagrant/public/sites/default`
- `$ drush up`
- If you are running multiple sites from the same code base, you may need to run the update script for subsequent sites (and clear the cache).
- `$ drush updb`
- `$ drush cc all`

Download Data to Your Test Site

Use the module Backup and Migrate to grab a copy of your live database.

```
$ drush help
```

All commands in backup_migrate: (backup_migrate)

- `bam-backup (bb)` Backup the site's database with Backup and Migrate.
- `bam-backups` Get a list of previously created backup files.
- `bam-destinations` Get a list of available destinations.
- `bam-profiles` Get a list of available settings profiles.
- `bam-restore` Restore the site's database with Backup and Migrate.
- `bam-sources` Get a list of available sources.

Create a Makefile for Your Site

This allows others to get a copy of your code base (sort of).

```
$ drush help make-generate
```

```
$ drush generate-makefile my_site_v2.make
```

(The file is added to the root directory of Drupal's installation. e.g. /vagrant/public/my_site_v2.make)

Collaborating With Others

- Determine a location for a “pick-up” spot for your code. (AKA The Central Repository.) This could be a machine in your network, GitHub, Bitbucket, ...
- Create a new project in your central repository.
- Push code to the central repository.
- Clone/Pull code to copy down the work of others.

Create/Apply a Patch

Your Git hosting platform probably has instructions.

e.g. <http://drupal.org/project/vert/git-instructions>

- `$ git diff > what_this_does.patch`
- `$ git apply --check what_this_does.patch`
- `$ git apply -v what_this_does.patch`

What Next?

- All the things!
 - <http://git-scm.com/doc>
 - <http://vagrantup.com/v1/docs/index.html>
 - <http://drush.ws/help/docs>
 - <https://help.github.com/>
 - <https://bitbucket.org/support>

Problems Reviewed

Problem:

What version of the file is on my server?

- “diff” -- what's the difference?
- never edit directly on the server, always “pull” from another location
- install only code which has a version associated with it (e.g. module.info)

Problem:

What Was I Thinking When I Wrote This?

- add in-code documentation
- use frequent commit messages, but only share (or “push”) relatively stable code
- put radically different, or unrelated, ideas into different branches
- put related ideas, which are allowed to evolve, into the same branch, but add tags to show milestones
- for very significant milestones, you may want to have different (numbered) branches

Problem:

Untested Code (Eventually) Breaks Stuff

- create a development environment on another machine.
- copy your data down (backup and migrate)
- keep everything versioned (have a central repository that is web-accessible which you can “pull” changes from for both your local env + dev server)
- upload your tested configuration changes.

Problem:

My Client Changed Their Mind ... Again

- You need a giant undo button to be able to roll back your code to a previous state.
- build all three designs (get basics right, then modify only bits but tag between each)
- add lots of “milestones” to your code so that you can see **easily** where things changed

Problem:

I Changed **Something**, And Broke Stuff.

- Make only very small changes before making commits.
- Commits in distributed version control systems are like an “undo” button you can apply after restarting your computer.
- Make only related changes within one commit. e.g. only font changes; vs. only colour changes

Problem:

My Computer Died...Now What?

- having your configuration files saved (and perhaps versioned) means you can quickly recover an entire system

Problem:

Two Clients Want Something Similar...

- one central repo
- create branches for each client
- where it makes sense, merge common functionality back into the main branch / trunk

Problem:

My [Collaborator] Overwrote My Work.

- You're working with a very tiny team and you think you don't need a whole source control system.
- Except you keep overwriting each others' work.
- Create a centralized repository that you both check your work into and push from that centralized server to the "live" site.
- On a regular basis "pull" your partner's work from the centralized repo.

After This Class

- Videos are coming soon! You will get an email with details on how to view the videos.
- If you still can't get vagrant / git working on your primary workstation, email me and we'll do a Skype call to get it working.