Git - Working with Remote Repositories

Handout

New Concepts

Working with remote Git repositories including setting up remote repositories, cloning remote repositories, and keeping local repositories in-sync with remote repositories.

Materials

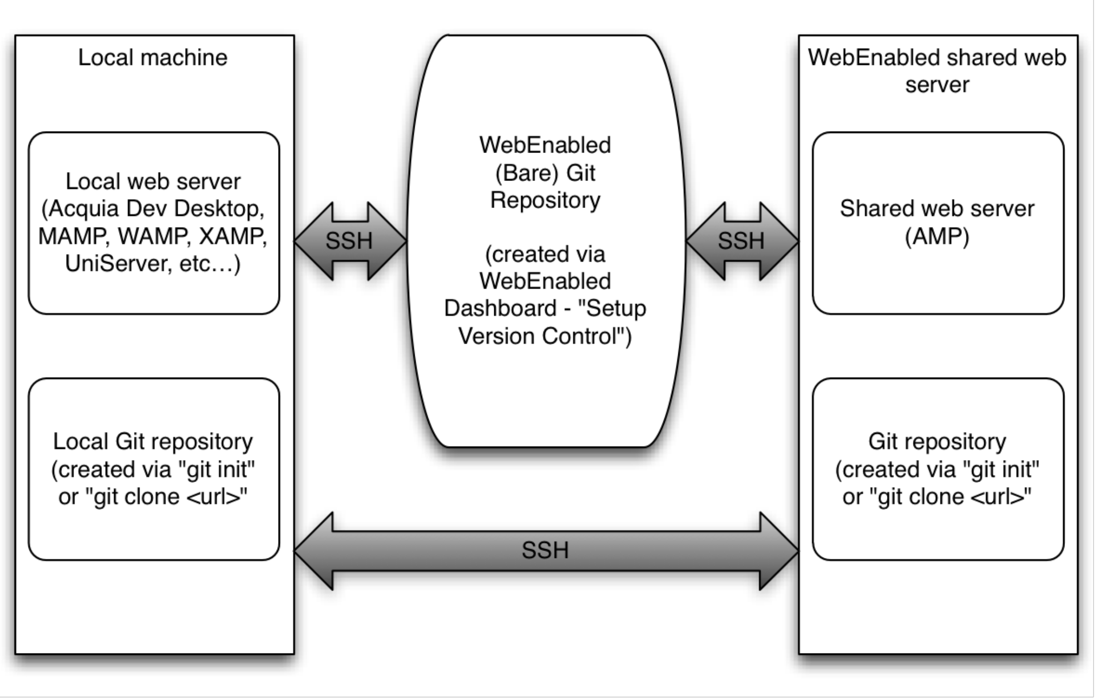
Each student should have a working instance of Drupal 7 on your computer and Git. A (free) user account with WebEnabled is required for this lesson as well. Each student should also have an SSH key-pair on their local machine, and have the public key uploaded to their WebEnabled user account. *NOTE: ensure that your SSH keys are working properly before proceeding with the steps in this document. See the “Additional Resources” section at the end of this document for generating and testing SSH keys.*

Instructor-Led Site Building

1. A remote Git repository is simply a repository that exists “somewhere else”. It may live in a different directory on your machine, a different machine in your office, or a different county on the planet.
    a. Remote repositories are useful for remote code backups as well as for when multiple people are working on the same code base.
    b. There are different protocols that can be used to communicate with remote repositories (including http://, ssh://, git://, and others), but this lesson will focus on one of the more common methods: ssh://.
2. Git is a “distributed” version control system, meaning that there doesn’t have to be just one “master” repository. In practice, each team member has a full copy of the repository,

and pushes and fetches commits to/from one or more remote repository in order to stay in-sync.

3. For this lesson, we'll focus on working with WebEnabled.com remote repositories. We'll communicate with the repositories via SSH - this method is very common and widely used. Organizations may use different locations for their remote repositories, and the vast majority of the methods in this lesson are not WebEnabled-specific.
4. It's important to note that a remote repository almost always exists on a machine that is not the web server. It is also important to realize that the web server is just another repository clone as far as the remote repository is concerned - exactly like each developer has a clone of the repository. So, when fetching from or pushing to the repository, we are not interacting with the web server.

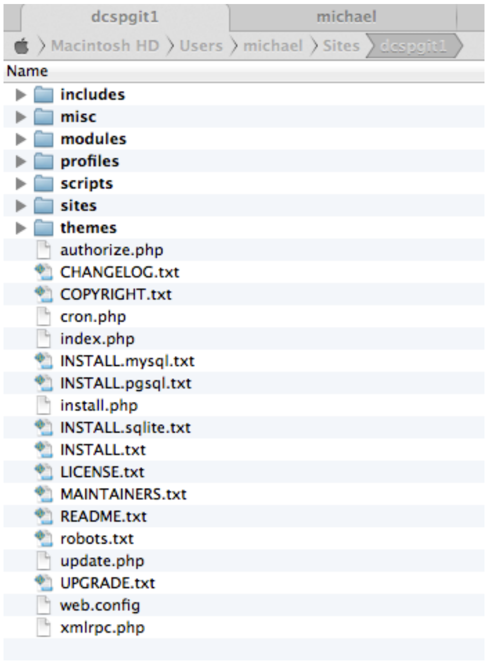

Local machine
Local web server (Acquia Dev Desktop, MAMP, WAMP, XAMP, UniServer, etc…)
Local Git repository (created via "git init" or "git clone <url>"
SSH
WebEnabled (Bare) Git Repository
(created via WebEnabled Dashboard - "Setup Version Control")
SSH
WebEnabled shared web server
Shared web server (AMP)
Git repository (created via "git init" or "git clone <url>"
SSH

The diagram above illustrates a typical setup using WebEnabled.com. Some details:

a. The WebEnabed (Bare) Git Repository is created from the WebEnabled Dashboard's "Version control" tab.
b. Whether "git init" or "git clone" is used to create the repository is dependent on where the site initially exists. For example, if the site only exists on the local machine, then "git init" is used to create the initial repository on that machine, then "git clone" is (eventually) used to create the repository on the shared web server. (Sections 6 and 7, below detail the process in both directions.) ***Note:*** *When "git init" is used, the "git remote add..." command is used to link the new repository with the remote WebEnabled (Bare) Git Repository.*
c. The SSH link between the local machine and both the WebEnabled (Bare) Git Repository and the WebEnabled shared web server is secured via the user's SSH keys. Each user must have a properly configured set of SSH keys on their local machine with a copy of the public key uploaded to their WebEnabled user account. (See Additional Resources - 1. "Authenticating with Git" below for details.)

   d. The SSH link between the WebEnabled (Bare) Git Repository and the WebEnabled shared web server is secured by a set of SSH keys generated by WebEnabled - this only occurs when the “Allow this website's unix user to have access to git without password” box is checked during the creation of the remote repository (see step 6.b.viii below).

5. Remote repositories are usually “bare” repositories. This means that they don’t actually have a copy of the repository’s files checked out. Instead, it just contains the complete commit history necessary to create the files. Local repositories usually contain both the complete commit history as well as a checked-out version of the files.
6. To push a local site to a new remote repository on WebEnabled:
   a. Create a local repository:
      i. Start with a current, local site that doesn’t have any version control associated with it. Be sure the Backup and Migrate module is installed (it will be used later to copy the DB from the local site to the WebEnabled site). For this example, consider a Drupal site named “dcspgit1”:



includes
misc
modules
profiles
scripts
sites
themes
authorize.php
CHANGELOG.txt
COPYRIGHT.txt
cron.php
index.php
INSTALL.mysql.txt
INSTALL.pgsql.txt
install.php
INSTALL.sqlite.txt
INSTALL.txt
LICENSE.txt
MAINTAINERS.txt
README.txt
robots.txt
update.php
UPGRADE.txt
web.config
xmlrpc.php

      ii. From a local command line, navigate into the dcspgit1 directory, and run “git init” to initialize the local git repository:

```
$:dcspgit1 michael$ pwd
/Users/michael/Sites/dcspgit1
$:dcspgit1 michael$ git init
Initialized empty Git repository in /Users/michael/Sites/
dcspgit1/.git/
```

iii. Use “git add .” to add all files to the git staging area. *Note: Windows users - see the note in the “Extra Stuff” section later in this document about handling line endings.*

iv. Use “git commit -m ‘Initial commit of my site” to create the initial commit. ***Note:*** *not all files in the dcspgit1 directory are automatically added to the initial commit. Drupal includes a standard “.gitignore” file that omits certain files from git repositories, including the entire /sites/default/files/ directory. (*For additional information, see:https://drupal.org/node/803746#gitignore.)

v. Use `git status` to confirm that all is well and the working directory is “clean”:

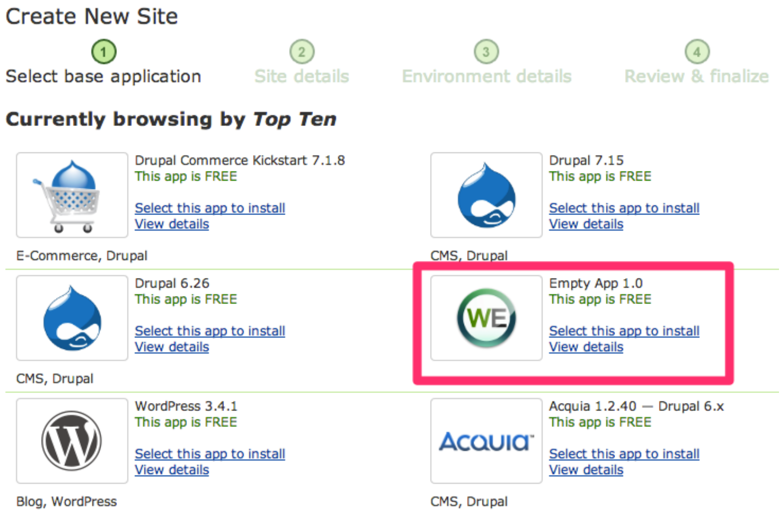
```
$:dcspgit1 michael$ git status
# On branch master
nothing to commit (working directory clean)
```

vi. The `git log` command can be used to view a list of all the commits on the current branch.

b. Create a new empty (“Empty App 1.0”) remote site and repository on WebEnabled’s servers:

i. Login to your WebEnabled account and click to “Start a site”.

ii. Select the “Empty App 1.0” to install.



Create New Site

1 Select base application — 2 Site details — 3 Environment details — 4 Review & finalize

Currently browsing by *Top Ten*

Drupal Commerce Kickstart 7.1.8
This app is FREE
Select this app to install
View details
E-Commerce, Drupal

Drupal 7.15
This app is FREE
Select this app to install
View details
CMS, Drupal

Drupal 6.26
This app is FREE
Select this app to install
View details
CMS, Drupal

Empty App 1.0
This app is FREE
Select this app to install
View details

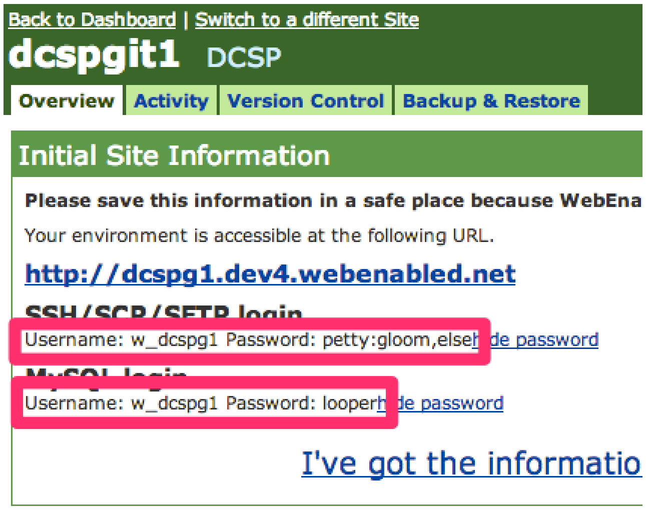
WordPress 3.4.1
This app is FREE
Select this app to install
View details
Blog, WordPress

Acquia 1.2.40 — Drupal 6.x
This app is FREE
Select this app to install
View details
CMS, Drupal

iii. For the “Site name”, it is recommended to use the same name as your local site. (For this example, “dcspgit1”). Select a “folder” if desired.

iv. Set a “shell username” and “app admin password”. ***Note***: *Remember what values you use!*

v. Once the site has been created, you’ll be directed to the “Initial Site Information” page (see image below). ***Note***: *It is vital that the usernames and passwords provided on this page are stored somewhere safe!*



dcspgit1 DCSP

Initial Site Information

Please save this information in a safe place because WebEna

Your environment is accessible at the following URL.

http://dcspg1.dev4.webenabled.net

SSH/SCP/SFTP login

Username: w_dcspg1 Password: petty:gloom,elseh de password

MySQL login

Username: w_dcspg1 Password: looperh de password

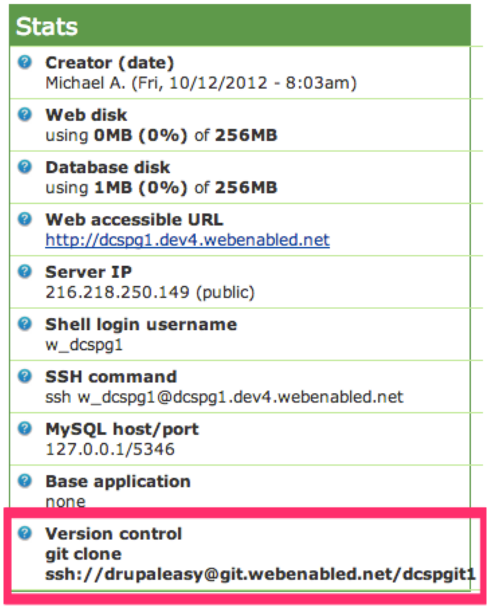
I've got the informatio

vi. At this point, you should be able to visit your new, empty site on WebEnabled's servers, but you'll just see the WebEnabled logo.

vii. Click the "version control" tab in your site's WebEnabled dashboard to create a new, empty repository.

viii. Click the "Setup Version Control" button, select "Git", leave the name the same as your site name, and be sure to check the "Allow this website's unix user to have access to git without password" box. Click the "Configure" button to create the empty repository. Recall from the image above that this repository lives on a different server than the empty site you just created. *NOTE: if you see a form that asks where the version control repository is located, select "WebEnabled".*

ix. Once complete, return to the main WebEnabled "overview" page for your site and notice the new "Version control" entry in the right-hand "Stats" block (image below).



Stats

Creator (date)
Michael A. (Fri, 10/12/2012 - 8:03am)

Web disk
using **0MB (0%)** of **256MB**

Database disk
using **1MB (0%)** of **256MB**

Web accessible URL
http://dcspg1.dev4.webenabled.net

Server IP
216.218.250.149 (public)

Shell login username
w_dcspg1

SSH command
ssh w_dcspg1@dcspg1.dev4.webenabled.net

MySQL host/port
127.0.0.1/5346

Base application
none

Version control
git clone
ssh://drupaleasy@git.webenabled.net/dcspgit1

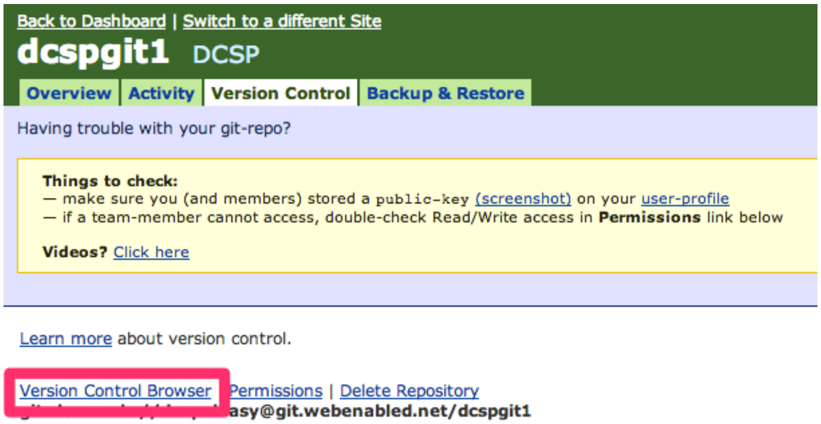
c. Push local commits to new remote repository:
   i. At this point, we have a local repository with a single commit on our local machine, and an empty remote repository on WebEnabled's servers. The next step is to link the two of them up and push our local commits to the new remote repository:
   ii. To add the empty WebEnabled repository as a remote repository for the local repository, use the "git remote add <name> <url>" command as follows:

```
git remote add origin ssh://drupaleasy@git.webenabled.net/dcspgit1
```

   Note that using a name of "origin" for the <name> is very common, but it can be set to any value. Some users name each remote as the name of the server, in this case, it could be "webenabled". The <url> is the value that WebEnabled provides in the "Stats" block as displayed above (but without the "git clone" part).
   iii. Assuming that your public/private SSH key is configured correctly, the "git remote add …" command won't provide any feedback if it works. Use "git remote -v" to confirm that all is well:

```
$:dcspgit1 michael$ git remote add origin ssh://
drupaleasy@git.webenabled.net/dcspgit1
$:dcspgit1 michael$ git remote -v
origin  ssh://drupaleasy@git.webenabled.net/dcspgit1 (fetch)
origin  ssh://drupaleasy@git.webenabled.net/dcspgit1 (push)
```

   This creates the "link" between the local repository and the remote repository.
   iv. Push all local commits to the remote repository using "git push origin master". This tells git to push all commits from the local to the remote named "origin" from the master branch (the default branch that is created for new repositories).
   v. At this point, the local repository and the remote repository are identical. You can verify that your local commits have been pushed to the WebEnabled repository by clicking on the "Version Control" tab of your site's dashboard, then clicking on the "Version Control Browser" link (see image below)



Having trouble with your git-repo?

Things to check:
— make sure you (and members) stored a public-key (screenshot) on your user-profile
— if a team-member cannot access, double-check Read/Write access in **Permissions** link below

Videos? Click here

Learn more about version control.

asy@git.webenabled.net/dcspgit1

vi. The "Version Control Browser" will display all commits that it contains, including the initial commit made in an earlier step on our local machine.

d. Clone the remote repository to a WebEnabled web server:

i. At this point, we have a local repository and a remote (bare) repository on WebEnabled's Git server. The next step is to clone the repository to WebEnabled's web server.

ii. From the "Stats" block on your site's WebEnabled Dashboard, use the "SSH Command" to login to your WebEnabled site via the command line.

iii. Navigate to your site's home directory, usually via "cd public_html/<site name>". In this example, it is, "cd public_html/dcspg1".

iv. This is the home directory of your empty site. Note that it contains two files - a simple HTML file and the WebEnabled logo file. Use the `rm` command to remove these files so that the home directory is completely empty. For example:

```
rm index.html
rm logo.png
ls
```

(verify that the directory is now empty).

v. Use the "git clone" command as listed in your site's "Stats" block to clone the remote repository to the home directory. Be sure to append a " ." to the "git clone" command to clone the repo into the current directory. For example:

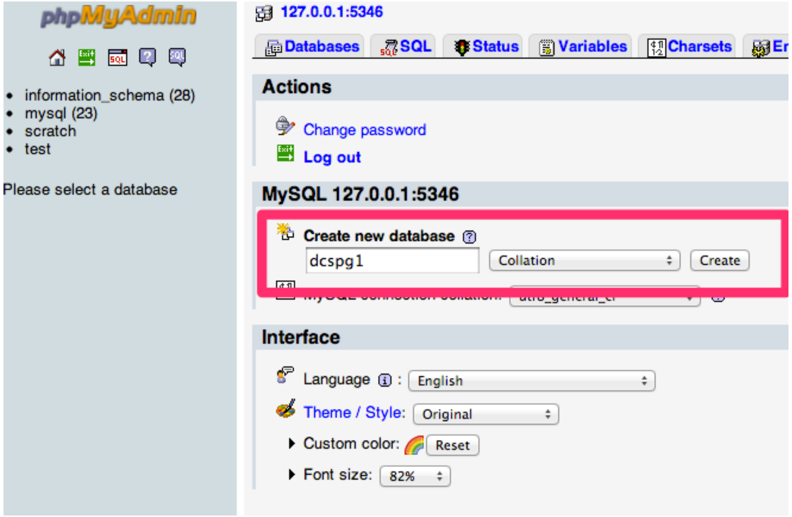
```
git clone ssh://drupaleasy@git.webenabled.net/dcspgit1 .
```

Once the clone is complete, use "ls" to confirm the presence of your code.

e. Get the WebEnabled site up-and-running:

i. Before we can get our WebEnabled Drupal site running, we need to create an empty database. From the WebEnabled "Overview" page, click the "phpmyadmin" link to launch it. The phpMyAdmin username (it will start with "w_") is displayed in the "Stats" block on your WebEnabled Dashboard. Utilize the "MySql password" you saved (from step 6.b.5 above).

ii. Once logged in to phpMyAdmin, create a new database, It is recommended to use the same name as your site. In this example, we'll set it to "dcspg1".



Create new database: dcspg1 | Collation

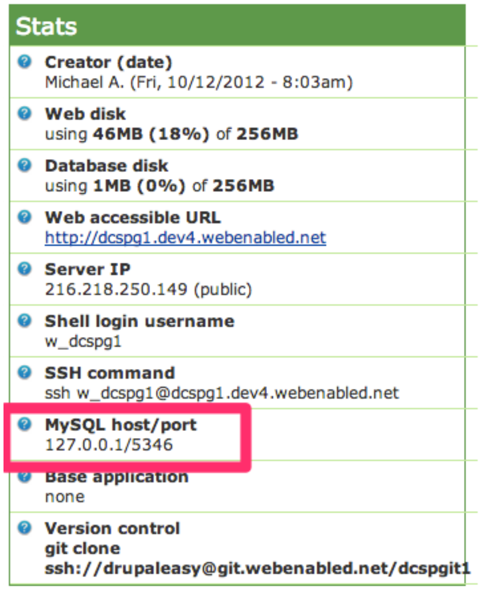
Language: English

Theme / Style: Original

Custom color: Reset

Font size: 82%

iii. Once created, the phpMyAdmin window isn't needed anymore.



Creator (date)
Michael A. (Fri, 10/12/2012 - 8:03am)

Web disk
using 46MB (18%) of 256MB

Database disk
using 1MB (0%) of 256MB

Web accessible URL
http://dcspg1.dev4.webenabled.net

Server IP
216.218.250.149 (public)

Shell login username
w_dcspg1

SSH command
ssh w_dcspg1@dcspg1.dev4.webenabled.net

MySQL host/port
127.0.0.1/5346

Base application
none

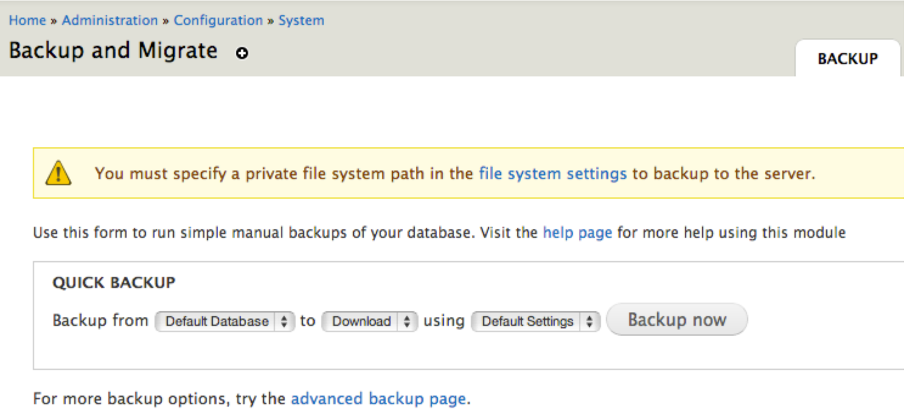
Version control
git clone
ssh://drupaleasy@git.webenabled.net/dcspgit1

iv. Now, visit your WebEnabled site (via the link provided in the "Stats" block on your WebEnabled Dashboard) and perform a normal Drupal installation of your site. We'll migrate the local database in a bit. For the database settings, utilize the information from the previous step as well as what is provided in the "Stats" block (image right). Be sure to set the "Database host", "Database port" in the "Advanced Options" area to the values provided in the "Stats" block.

v. Click through to complete the Drupal installation as desired.

vi. At this point, our code is sync'd between the local and WebEnabled sites, but our database only exists on our local machine. Use the Backup and Migrate module to copy the local DB to WebEnabled.

vii. From your local site, make sure the Backup and Migrate module is enabled and navigate to Configuration | System | Backup and Migrate (admin/config/system/backup_migrate), and click to perform a "Quick backup" using the default settings.



Home » Administration » Configuration » System

Backup and Migrate

You must specify a private file system path in the file system settings to backup to the server.

Use this form to run simple manual backups of your database. Visit the help page for more help using this module

QUICK BACKUP

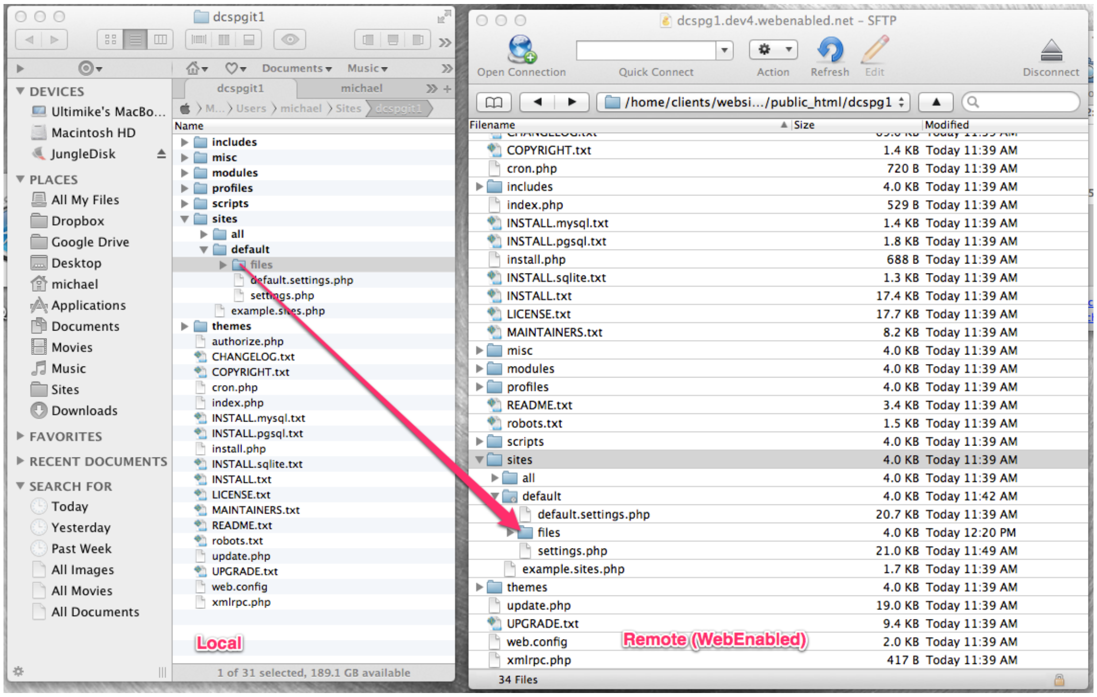
Backup from Default Database to Download using Default Settings

For more backup options, try the advanced backup page.

The backup will be "downloaded" from your site to your browser's default downloads directory. ***Note:*** *Windows users: depending on your site setup, you may need to click the "advanced backup page" and change the "Compression" to "Zip".*

viii. On the WebEnabled site (destination), ensure Backup and Migrate is enabled, and navigate to Configuration | System | Backup and Migrate (admin/config/system/backup_migrate), but this time, click on the "Restore" tab. We're going to use the backup we just made on the local site and "restore" the WebEnabled site to it. On the "Restore" page, use the "Upload a Backup File" and select the local backup file you just created. Click the "Restore now" button to proceed. ***WARNING****: be sure you are restoring to the proper site, as this step will remove the entire existing DB and replace it with the one that is being uploaded. If you're not 100% sure, create a DB backup for this site as well.*

ix. Once the "restore" process is complete, the local and WebEnabled sites should have identical content and configuration, with the exception of images and files uploaded to the site as part of nodes, users, etc.

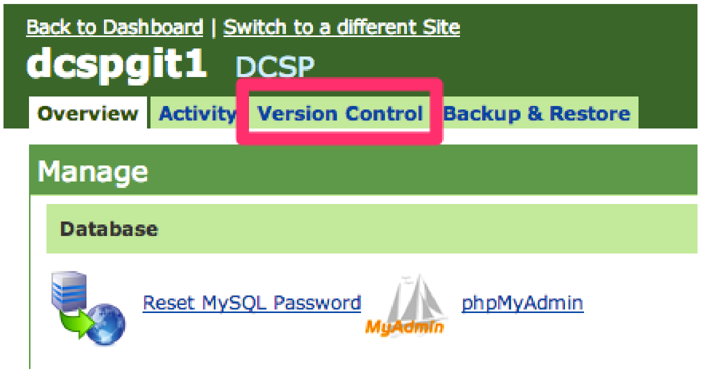


Filename	Size	Modified
COPYRIGHT.txt	1.4 KB	Today 11:39 AM
cron.php	720 B	Today 11:39 AM
includes	4.0 KB	Today 11:39 AM
index.php	529 B	Today 11:39 AM
INSTALL.mysql.txt	1.4 KB	Today 11:39 AM
INSTALL.pgsql.txt	1.8 KB	Today 11:39 AM
install.php	688 B	Today 11:39 AM
INSTALL.sqlite.txt	1.3 KB	Today 11:39 AM
INSTALL.txt	17.4 KB	Today 11:39 AM
LICENSE.txt	17.7 KB	Today 11:39 AM
MAINTAINERS.txt	8.2 KB	Today 11:39 AM
misc	4.0 KB	Today 11:39 AM
modules	4.0 KB	Today 11:39 AM
profiles	4.0 KB	Today 11:39 AM
README.txt	3.4 KB	Today 11:39 AM
robots.txt	1.5 KB	Today 11:39 AM
scripts	4.0 KB	Today 11:39 AM
sites	4.0 KB	Today 11:39 AM
all	4.0 KB	Today 11:39 AM
default	4.0 KB	Today 11:42 AM
default.settings.php	20.7 KB	Today 11:39 AM
files	4.0 KB	Today 12:20 PM
settings.php	21.0 KB	Today 11:49 AM
example.sites.php	1.7 KB	Today 11:39 AM
themes	4.0 KB	Today 11:39 AM
update.php	19.0 KB	Today 11:39 AM
UPGRADE.txt	9.4 KB	Today 11:39 AM
web.config	2.0 KB	Today 11:39 AM
xmlrpc.php	417 B	Today 11:39 AM

As a final step, if necessary, use an SFTP client, Drush rsync, or some other method to copy the local “files” directory (usually found in sites/default/files) to WebEnabled’s servers. Be sure to copy the directory to the same relative path. ***WARNING****: Do not overwrite your settings.php file!*

x. When complete, your local and remote (WebEnabled) sites should be identical!

7. Create a new site on WebEnabled and clone to your local machine;
   a. Create a new WebEnabled site.
      i. Login to your WebEnabled account and click to “Start a site”.
      ii. Select a base application - and proceed through the installation process until complete.
      iii. Configure the new site as desired. Add modules (WebEnabled has Drush built-in), create content types, views, etc.
   b. Create a new WebEnabled remote repository for the site.
      i. Referring to the diagram in section 4 above, we first need to create an empty remote repository to push to. Click on the “Version Control” tab on your site’s WebEnabled Dashboard to begin.



ii. Click the “Setup Version Control” button, select “Git”, leave the name the same as your site name, and be sure to check the “Allow this website's unix user to have access to git without password” box. Click the “Configure” button to create the empty repository.

iii. Once complete, return to the main WebEnabled “overview” page for your site and notice the new “Version control” entry in the right-hand “Stats” block.

c. Push the new WebEnabled site to the remote repository.

i. From the “Stats” block on your site’s WebEnabled Dashboard, use the “SSH Command” to login to your WebEnabled site via the command line.

ii. Navigate to your site’s home directory, usually via “cd public_html/<site name>”. In this example it is: “cd public_html/dcspg1”. (This is the home directory of the site you created in section 7.a.)

iii. Initialize a git repository in this directory via “git init”.

iv. Use “git add .” to add all files to the git staging area.

v. Use “git commit -m ‘Initial commit of my site’” to create the initial commit.

vi. Use “git status” to confirm that all is well and the working directory is “clean”.

vii. To be able to push these commits to the empty repository we set up in section 7.b., use the “git remote add <name> <url>” command.

For example, it is as follows:

```
git remote add origin ssh://drupaleasy@git.webenabled.net/dcspgit1
```

Note that using a name of “origin” for the <name> is very common, but you can set it to any value. Some users name each remote as the name of the server, in this case, it could be “webenabled”. The <url> is the value that WebEnabled provides in the “Stats” block as displayed above (but without the “git clone” part).

viii. Use “git remote -v” to confirm that all is well.

ix. Push your initial commit to the remote repository via “git push origin master”.

d. Clone the remote repository to your local machine.

i. Via the command line, navigate to your local “sites” directory (or to the location you store the rest of your sites).

ii. Utilize the “git clone” command specified in the “Stats” block of your WebEnabled Dashboard to clone the remote repository to your local machine. For this example, the command is:

```
git clone SSH://drupaleasy@git.webenabled.net/dcspgit1
```

Your command will be slightly different depending on your WebEnabled username and site name. This will clone the remote repository into a directory in your sites directory.

e. Getting the local site site up-and-running:

i. Create a empty database on your local machine using phpMyAdmin.

ii. Navigate to your new local site and complete the default Drupal installation. ***Note:*** *Be sure to use your local DB information, not the WebEnabled DB information.*
iii. Utilize the Backup and Migrate module to backup the DB on the WebEnabled site and restore it to your local site, similar to steps 6.e.vii-6.e.ix, only in the opposite direction (WebEnabled -> local).
iv. Sync the “files” directory using SFTP or Drush rsync, similar to step 6.e.x, only in the opposite direction (WebEnabled -> local).

Extra Stuff

1. Steps to get a cloned site up-and-running on Drupal Quickstart: (http://www.drupal.org/project/quickstart)
   a. Create a new virtual host for your site using:

      ```
      drush qc dns apache --domain=<your_new_site.dev>
      ```

      Running this command will create a new “your_new_site.dev” directory in QuickStart’s “websites” directory.

      Additionally, if you also want to create an empty database or your site, you can use:

      ```
      drush qc apache database --domain=<your_new_site.dev>
      ```

      This will also create an empty database named “your_new_site_dev”.
   b. Navigate into the new Desktop/websites/your_new_site.dev folder via the command line.
   c. Run the "git clone <URL> ." command.
   d. You can access your site via a web browser at your_new_site.dev

2. Windows users - if, when attempting a local “git add”, you see a fatal error like, “LF will be replaced by CRLF”, run the command “git config --global core.autocrlf false”. See https://rubyhood.wordpress.com/2011/01/28/git-warning-lf-will-be-replaced-by-crlf/ for additional details.

Additional Resources

1. Using Git to clone from WebEnabled to a local machine - https://www.youtube.com/watch?v=Lo50e-Q01ZE (via http://youtube.com/drupaleasy)
2. Using Git to push a local site to WebEnabled.com - https://www.youtube.com/watch?v=-5FCVZdm2Q4 (via http://youtube.com/drupaleasy)
3. Authenticating with Git (includes information about using SSH keys): https://drupal.org/node/1027094
4. Generate and upload SSH keys from Drupal to Git: https://drupal.org/node/1066762

5. WebEnabled’s help document on setting up SSH: http://webenabled.com/creating-and-uploading-your-ssh-public-key
6. Git reference guide and cheat sheets: http://git-scm.com/docs