DevShop
Drupal Infrastructure in a Box

Jon Pugh
CEO, Founder
ThinkDrop Consulting
Brooklyn NY
Who?

Jon Pugh
Building the web since 1997.
Building web apps since 2000.
Building Drupal since 2004.

ThinkDrop Consulting
Founded in 2009 in Brooklyn NY.
100% Drupal and related.
Focus on hard problems: Migrations, Integrations, Security, Permissions, and Infrastructure.
What?

"Drupal Infrastructure"

1. "Hardware" infrastructure: The computers physically storing and processing your sites and data.
2. Software Infrastructure: The software tools that make Drupal possible: Linux, Apache, MySQL, PHP.
3. Drupal Infrastructure: All of the above, plus scripts and tools to manage Drupal sites and data.
What?

DevShop

Open Source

Drupal

Infrastructure

Management

System

IMS
What?

IMS

A Drupal Infrastructure Management System:

1. Provides a web and API interface for managing multiple environments and servers for Drupal sites.
2. Automates Drupal site provisioning.
3. Integrates with version control systems.
Most Drupal IMS are:

1. Only available as a service.
2. Locked into a single hardware provider.
3. Expensive.
4. Servers are automatically provisioned.
What?

IMS

DevShop

1. Is free and open source software.
2. Can be installed on any Linux server.
3. Does not automatically provision servers.
4. Does not tell you how to configure your server.
DevShop is built on Aegir.

*Automating Drupal Hosting since circa 2004.*

Aegir handles:

- Provisioning database, db users, and grants.
- Creates VirtualHost config files.
- Creates settings.php, files folders, sets permissions.
- Allows deployment to multiple servers.
DevShop is built on Aegir.

Enhancing Aegir since February 2012

DevShop adds to Aegir:

- Uses Git to deploy Drupal sites.
- Provides extra tools to Pull Code, Sync Data, etc.
- Friendly add-on features for efficient development.
Where?

http://drupal.org/project/devshop
DevShop
Installation
How?

1. Get a (preferably short) domain name.
2. Get a Server and set the hostname to be the domain name. Use Ubuntu 12.04 LTS.
3. Add a DNS record for your domain and IP, plus a wildcard record:
   - `ourdevshop.com. IN A 1.2.3.4`
   - `*.ourdevshop.com. IN A 1.2.3.4`
4. `wget http://getdevshop.com/install.ubuntu`
   `sudo sh install.ubuntu`
Congratulations, DEVSHOP has now been installed.

You should now log in to the DEVSHOP frontend by opening the following link in your web browser:

http://devshop/user/reset/1/1373398281/8a5228302d8d2b8bcede294f76b822d4a

Stopping supervisor: supervisord.
Starting supervisor: supervisord.
devshop_public_key was set to ssh-rsa
AAAAAB3NzaC1yc2EAAAADAQABAAABABQCaGabywvUii7ADm2Koc2GSRivEUGlegGEDeNMLi9EzsNdTjyHtUPyNG+3K+BK+1BPsw/9X0cEka4lbamD0f+4F6VLFPjf+xSYDbmQDodoqNk+wALblBgR8oGQao+AeQfEvGHi5i/W04H22id8zOnjuTQ4L7hizLLmS02gOGm6EDGr1rtQVJQ1TkjMzgUwvZ4VcnMiiqtkyNWHwfb2BdJwPGI3qMS8UK/A21g
TBAGfEGz9zZhA3RaWd0iZd
eaegir@devshop.

Your MySQL root password was set as nI7aTTRq7drDmQlqa9Yp7G5yNCWbsR0
This password was saved to /tmp/mysql_root_password
You might want to delete it or reboot so that it will be removed.

An SSH keypair has been created in /var/aegir/.ssh

Supervisor is running Hosting Queue Runner.

Welcome to DevShop!
Use the link above to login to your DevShop
You should now reboot your server.
Reset password

This is a one-time login for admin and will expire on Wed, 07/10/2013 - 19:31.
Click on this button to login to the site and change your password.
This login can be used only once.

Log In
You have just used your one-time login link. It is no longer necessary to use this link to login. Please change your password.

Username: admin

Spaces are allowed; punctuation is not allowed except for periods, hyphens, and underscores.

E-mail address: webmaster@devshop

A valid e-mail address. All e-mails from the system will be sent to this address. The e-mail address is not made public and will only be used if you wish to receive a new password or wish to receive certain news or notifications by e-mail.

Password:

Confirm password:
DevShop

Create your first project.
You have no projects.

Create your first Project
You have your Drupal site in a git repo, don't you?
This server's Public SSH Key. Add to your Git repository account so the server can access your repo.
git ls-remote git@github.com:thinkdrop/drupal-flat.git
Step 1: Source » Step 2: Environments » Step 3: Install Profile

<table>
<thead>
<tr>
<th>Environment Name</th>
<th>Visible Error Logs</th>
<th>Git Branch</th>
<th>Web server</th>
<th>Database server</th>
<th>Pull on Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name this environment...</td>
<td></td>
<td>dev</td>
<td>devshop</td>
<td>localhost</td>
<td></td>
</tr>
</tbody>
</table>

Create as many new environments as you would like. For example: "dev", "test", and "live"
## Name your environments

- dev
- test
- live

## Pick your Git Branches

- dev
- test
- live

## Choose Servers

- Web server: devshop
- Database server: localhost
- Pull on Commit: 

## Extensible per-environment Settings

- Visible Error Logs
- Git Branch
- Web server
- Database server
- Pull on Commit
Cloning your repo to each of your environments.

git clone git@github.com:thinkdrop/drupal-flat.git --branch dev
Once cloned, DevShop analyzes your Drupal code.
Your project has been created. Once installed, your sites will be available.

DevShop Project nycdevshop

Dashboard

Submitted by admin on Tue, 07/09/2013 - 19:37

Git URL: git@github.com:thinkdrop

Remote Branches: dev dev-jon live
- master
test

Install profile: standard

Environments

<table>
<thead>
<tr>
<th>Environment</th>
<th>URL</th>
<th>Dev</th>
<th>Commits</th>
<th>Errors</th>
<th>Files</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>dev</td>
<td>dev.nycdevshop.devshop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>live</td>
<td>live.nycdevshop.devshop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Task

- Verify Project: View Run
- Pull Code: View Run
- Sync Data: View Run

Actions
BACKEND MAGIC

MYSQL:
CREATE USER 'devnycdevshop'; CREATE DATABASE 'devnycdevshop';
GRANT PERMISSIONS;

APACHE:
Create <virtualhost> entry.

DRUPAL:
Create settings.php and files folders.

DRUSH:
Create drush site alias.
Install Drupal.

FOR ALL ENVIRONMENTS
Welcome to dev.nyccamp.devshop

No front page content has been created yet.
Project Settings

Easy-to-copy git URL

Available git branches

Last automatic Git Pull

Git URL: git@github.com:thinkdrop
Remote Branches: dev dev-jon live master nyctmp test
Install profile: standard
Last Commit: 3 hours ago
MORE ON THIS LATER...
DevShop
Connect your Git Repo

"Auto-pull!"
### Environment Settings

<table>
<thead>
<tr>
<th>Environment</th>
<th>Visible Error Logs</th>
<th>Git Branch</th>
<th>Web server</th>
<th>Database server</th>
<th>Pull on Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>dev</td>
<td>✓</td>
<td>dev</td>
<td>devshop</td>
<td>localhost</td>
<td>✓</td>
</tr>
<tr>
<td>test</td>
<td>✓</td>
<td>test</td>
<td>devshop</td>
<td>localhost</td>
<td></td>
</tr>
<tr>
<td>live</td>
<td></td>
<td>live</td>
<td>devshop</td>
<td>localhost</td>
<td></td>
</tr>
</tbody>
</table>

### Git Settings

**Automatic Git Pull Method:**
- Pull disabled.
- Pull on queue (every minute).
- Pull on URL Callback (ie. GitHub Webhook)

Choose the method of regularly calling “Pull Code”. See Hosting > Queues to configure the queue. See Hosting > DevShop Pull Settings to configure URL Callback.

**Pull Trigger URL:**

http://devshop/devshop/pull/project_nycdevshop/3fe36766dabd0d8a04d8a238092af1b6

Configure your repo to hit this URL when it receives a commit.
Available Service Hooks

WebHook URLs (1)

- ActiveCollab
- Acunote
- AgileBench
- AgileZen
- AmazonSNS
- Apiary
- Apolo
- AppHarbor
- Apropos
- Asana
- Backlog
- Bamboo
- BasecampClassic
- Basecamp
- Blimp
- Boxcar
- buddycylabel (GitHub plugin)

WebHook URLs

URL (remove)

http://devshop/devshop/pull/project_nycdevshop/3fe36766dabd0d

URL (remove)

Add another webhook URL

Test Hook  Update settings

We'll hit these URLs with POST requests when you push to us, passing along information about the push. More information can be found in the Post-Receive Guide.

The Public IP addresses for these hooks are: 204.232.175.64/27, 192.30.252.0/22.
1. Git host receives a commit.
2. Git host pings your DevShop.
3. DevShop runs "Pull Code".
4. Drupal caches are cleared.

Doesn't run update.php or revert all features. Don't automate those things.
DevShop

Tasks, not Scripts
"Tasks"

- Front-end interface to backend commands.
- Drupal module + a drush command working together.

```sh
drush @project_name provision-devshop-commit

drush @project_name provision-devshop-pull

drush @project_name provision-devshop-sync

drush @project_name provision-devshop-test
```
Tasks

- Drupal forms become drush options.

```bash
drush @project_name provision-devshop-commit-dev --message="Features Export!" --push --revert
```
"Tasks"

- Tracks who started it, when, and how long it took.
- Error logs are stored for easy viewing.
Tasks

- Every "script" you use to manage your server should be a task.
- A task is nothing more than a drush command and a Drupal module.
DevShop
Development Workflow Tasks
Pull & verify platform code and (optionally) run update.php, clear cache, and revert features.

Environments: *
- [x] dev
- [ ] test
- [ ] live

Choose the environments to pull code to.

- Run update.php after code pull?
- Revert all features after code pull?
- Clear cache after code pull?

Pull Code
Sync Data

drush @project_name provision-devshop-sync dev test --files --database --update --revert --cache --pull

git pull
drush sql-sync @source @destination
drush rsync @source:%files @destination:%files
drush update-db
drush features-revert-all
drush cc all;
drush cc all;
drush cc all;
Run Tests

drush @project_name provision-devshop-test dev SystemIndexPhpTest

drush en simpletest
drush test-run SystemIndexPhpTest
Commit Features

```
drush @project_name provision-devshop-commit dev --message="Commit from the Web!" --push --revert
```

```
drush features-update-all

git commit -am

git push

drush features-revert-all --force
```
Fork Environment

```
git clone --branch dev

git checkout -b newbranch

git push origin newbranch

drush sql-sync @source @new_environment
```
DevShop Servers

Must setup the server ahead of time:

- LAMP + Aegir user + SSH Keys

Once connected with SSH, Aegir can put Drupal platforms and their sites on any server.

On "Verify", Aegir rsync's the codebase to the remote servers.
Create Server

Server hostname:

The host name of the server. This is the address that will be used by Hostmaster to connect to the server to issue commands. It is to resolve to the internal network, if you have such a separation.

Be careful when changing the node title, it is used to create the SQL users if the IP address, below, is not set.

Ip addresses:

A list of IP addresses this server is publicly available under, one per line. If none is specified, a DNS lookup will be performed based on the server hostname above.

This should point to the public network, if you have such a separation.

Soir
- None
- tomcat

Database
- None
Server hostname:

The host name of the server. This is the address that will be used by Hostmaster to connect to the server to issue commands. It is to resolve to the internal network, if you have such a separation. Be careful when changing the node title, it is used to create the SQL users if the IP address, below, is not set.

Ip addresses:

127.0.1.1

A list of IP addresses this server is publicly available under, one per line. If none is specified, a DNS lookup will be performed based on the server hostname above. This should point to the public network, if you have such a separation.

Solr

- None
- tomcat

Database

- None
- mysql

Web

- None
- apache

Port:

80

The port that a service is listening on.

Restart command:

```
sudo /usr/sbin/apache2ctl graceful
```

The command to run to restart this service.
Database:
- None
- mysql

Port: 3306

Username: root

You have already set a password for this database server.
The password for the user that will be used to create new mysql users and databases for new sites.
DevShop

Bonus Features
Bonus Features

Apache Solr for every site.
Download Drush Aliases
HTTP Password Protection
Server Clusters and Master/Slave DBs
NGINX
Automatic Cron Setup
DevShop

The Future?
The Future?

Better GitHub Integration:

- Automated Git repo creation.
- Automatic Environments from Pull Requests.
- Merge Branch interface.
- Tag and Release system.
The Future?

Better User Management

- SSH Key uploading.
- Assign users to project.
- Activity streams.
The Future?

Better User Interface
- Needs a major facelift.
- Responsive/Mobile web.
The Future?

Chef Integration

- node/add/server
- Improve the open source chef recipes for Aegir/DevShop.
- Integrate with any chef knife or openstack-compatible cloud host to instantly deploy properly configured servers.
Who?

**DevShop Contributors**

Jon Pugh  
alchemist  
fastangel  
helmo  
frazras  
wodenx  
ergonlogic  
kjl  
dfried

**Aegir Contributors**

ergonlogic  
anarcat  
Deciphered  
helmo  
Steven Jones  
omega8cc  
thedavidmeister  
Jon Pugh  
fastangel  
jlscott  
mig5  
tstoeckler  
Dane Powell  
GuyPaddock  
seth.  
v Vincent  
joestewart  
sfyn  
Thomas Bosviel  
cafuego  
adrian  
drumm  
yhahn  
spiderman  
liorkesos  
lieb  
marvil07  
wamilton  
Pol  
bgm  
halcyonCorsair  
juampy  
Thomas Bosviel  
greenscientist  
mathieu
Questions?

drupal.org/project/devshop
#devshop and #aegir on irc.freenode.net

Jon Pugh
CEO, Founder
ThinkDrop Consulting
Brooklyn NY
Get Involved in Core Sprints!

1. It starts at 9am tomorrow.
2. Anyone who has site building experience and a laptop is qualified and encouraged to come.
3. There will be friendly mentors on-site to help and match you with tasks.