Back to 2006...

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
PHP Development Environments

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
Sharing the Project

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
Deployment

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
Hosting

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
Now back to 2015...
What a time to be alive!

DigitalOcean

github

php7

composer

debian

ubuntu

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
But, still...

“It works on my machine”
What if I told you...
What if I told you...

There's a way to make your development process more consistent, while also smoothing the transition from dev to prod?
Presenting: D.R.I.V.E

- Disposable
- Replicable
- Isolated
- Versioned
- Environment
Presenting: D.R.I.V.E

- Disposable
- Replicable
- Isolated
- Versioned
- Environment
D.R.I.V.E

Disposable & Isolated

Replicable & Versioned

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
D.R.I.V.E

Disposable & Isolated

- Virtual Machines, Containers

Replicable & Versioned

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
D.R.I.V.E

Disposable & Isolated

- Virtual Machines, Containers
  - VirtualBox, VMWare, KVM, etc
  - Docker

Replicable & Versioned

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
D.R.I.V.E

Disposable & Isolated
- Virtual Machines, Containers
  - VirtualBox, VMWare, KVM, etc
  - Docker

Replicable & Versioned
- Configuration Management

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
<table>
<thead>
<tr>
<th>D.R.I.V.E</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Disposable &amp; Isolated</strong></td>
</tr>
<tr>
<td>• Virtual Machines, Containers</td>
</tr>
<tr>
<td>− VirtualBox, VMWare, KVM, etc</td>
</tr>
<tr>
<td>− Docker</td>
</tr>
<tr>
<td><strong>Replicable &amp; Versioned</strong></td>
</tr>
<tr>
<td>• Configuration Management</td>
</tr>
<tr>
<td>− Ansible, Puppet, Chef, Salt, etc</td>
</tr>
</tbody>
</table>

**DRIVE with Vagrant and Ansible**
@erikaheidi / PHP[world] 2015
Vagrant

• Virtual Machines + Configuration Management
• Easy DRIVE for DEV
• Made Conf. Management popular with developers
  – And we nailed it!

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
Otto?

$otto
dev

$otto
deploy

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
"Otto is very young and while we tout it as a successor to Vagrant, it'll take time to reach the maturity level that Vagrant is already at. We don't recommend dropping Vagrant today, but look to Otto for the future."
The Vagrantfile

Vagrant.configure("2") do |config|

config.vm.box = "ubuntu/trusty64"
config.vm.network "forwarded_port", guest: 80, host: 8080
config.vm.synced_folder "/vagrant", "."
config.vm.provision "shell", inline: "echo hello"
end
This whole provisioning thing...
Ansible

- Simple and straightforward language (YAML)
- Agentless Architecture
- Huge collection of built-in modules
- Great community, very popular on Github - 13k+ stars and almost 4k forks
Playbook Example

---

- hosts: all
  sudo: true
  tasks:
    - name: Update apt-cache
      apt: update_cache=yes
    - name: Install Nginx
      apt: pkg=nginx state=latest
Playbook Resources

- Variables
- Loops
- Conditionals
- Templates
- Ansible Vault
LET'S DRIVE?

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
Vagrantfile

#Vagrantfile
Vagrant.configure("2") do |config|

  config.vm.box = "ubuntu/trusty64"
  config.vm.network "forwarded_port", guest: 80, host: 8080

  config.vm.provision "ansible" do |ansible|
    ansible.playbook = "demo.yml"
  end

end
---

- hosts: all
  sudo: true
  vars:
    packages: ["nginx", "php5-fpm"]
  tasks:
    - name: Update apt-cache
      apt: update_cache=yes
    - name: Install Packages
      apt: pkg={{ item }} state=latest
        with_items: packages
    - name: Change Nginx Vhost File
      template: src=default.tpl dest=/etc/nginx/sites-available/default
      notify: restart nginx
  handlers:
    - name: restart nginx
      service: name=nginx enabled=yes state=restarted
DEMO

When this baby hits 88 miles per hour... You're gonna see some serious shit

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
when this baby hits 88 mph...
1. Roles

```
#playbook.yml
---
- hosts: all
  sudo: true
  vars:
    doc_root: /vagrant/web
  roles:
    - init
    - webserver
```
2. Phansible

Phansible provides an easy-to-use interface that helps you generate Ansible provisionings for PHP-based projects.

Vagrant Settings

Choose the options for your virtual machine.

Vagrant / VirtualBox Options

Basic Options

Base Box:

Ubuntu Trusty Tahr (14.04) (LT5) 64

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
2. Phansible

```
. ├── ansible
│   ├── inventories
│   │   └── dev
│   └── playbook.yml
├── roles
│   ├── app
│   │   └── mysql
│   │   └── nginx
│   │   └── php
│   │   └── server
│   │       └── vagrant_local
│   └── vars
│       └── all.yml
└── Vagrantfile
```
2. Phansible

```
. ├── ansible
|   ├── inventories
|   |   └── dev
|   └── playbook.yml
|   └── roles
|       ├── app
|       ├── mysql
|       ├── nginx
|       └── php
|           ├── server
|           └── vagrant_local
|   └── vars
|       └── all.yml
└── Vagrantfile
```

```bash
#inventories/dev
[phansible-web]
192.168.33.10
```
3. group_vars

```
.ansible
  └── inventories
      ├── dev
      │    └── test
      │         └── prod
      └── group_vars
          ├── all.yml
          │    └── web-dev.yml
          │         └── web-prod.yml
          │                 └── web-test.yml
          └── playbook.yml
          └── roles
              └── windows.sh
  Vagrantfile
```

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
3. group_vars

```
# inventories/prod
[web-prod]
myapp.io
```
3. group_vars

#inventories/prod
[web-prod]
myapp.io

#group_vars/web-prod.yml
project_root: /var/www
doc_root: /var/www/web
sys_packages: ["vim","fail2ban"]
3. group_vars

```
#inventories/prod
[web-prod]
myapp.io
```

```
#group_vars/web-prod.yml
project_root: /var/www
doc_root: /var/www/web
sys_packages: ["vim","fail2ban"]
```

```
DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
```
4. ansible-vault

$ ansible-vault encrypt group_vars/web-prod.yml

Before

<table>
<thead>
<tr>
<th>twitter:</th>
<th>otherthing:</th>
</tr>
</thead>
<tbody>
<tr>
<td>app_token: MYSUPERTOKEN</td>
<td>secret_thing: SECRET</td>
</tr>
<tr>
<td>app_secret: MYSUPERSECRET</td>
<td>secret_other: TOPSECRET</td>
</tr>
</tbody>
</table>

After

$ANSIBLE_VAULT;1.1;AES256
39356166303165393330613634373636
61343834313564386262323234303063
3539656138633383
32353631303265623232306338303266
65306531633835630a36306133306539
3835356331343862
32346132653432623766366161333334
66393964396261303637313335646463

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
4. ansible-vault

$ ansible-vault view path/to/varfile.yml

$ ansible-vault edit path/to/varfile.yml

$ ansible-playbook (...) --ask-vault-pass
DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
QUESTIONS?
Please rate the talks you attend!

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015
THANKS!

See you in the future.

DRIVE with Vagrant and Ansible
@erikaheidi / PHP[world] 2015