



CODING

DevOps Strategy

ChoTot.vn

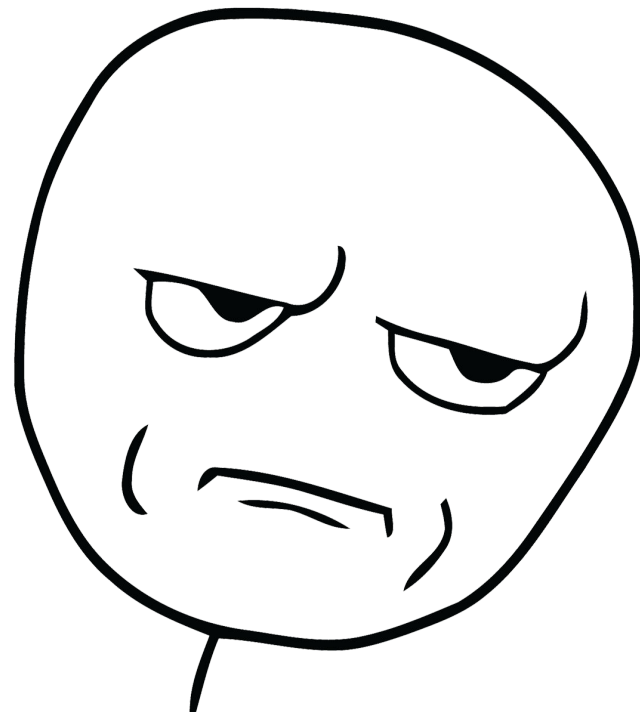
Who am I?

- Software Engineer at Chotot.vn Core Team
- Rogue hacker
- Otaku
- Gothicist

Thinh “Adolf Weismann” Tran
duythinh@gmail.com
@duythinh

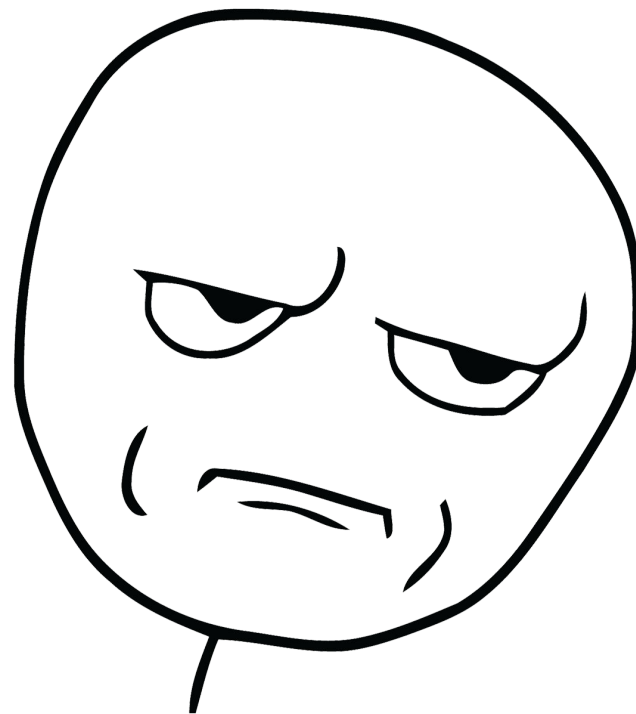


“It's works on my machine”

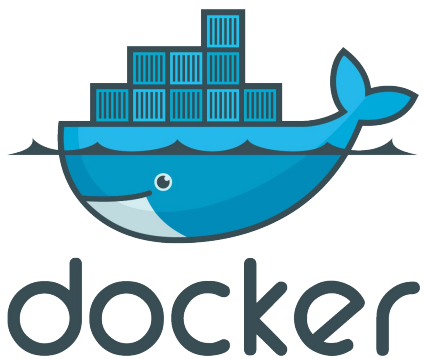


REALLY..????

“It’s all in setup.md”



REALLY..????



What is Vagrant?

- Command line tool
- Automates VMs creation with
 - VMWare
 - VirtualBox
 - Hyper-V
- Integrates well with configuration management tools
 - shell
 - chef
 - puppet
 - ansible



VAGRANT

Why use Vagrant?

- Easy to create VM with one command

`$ vagrant up`

- Reproducibility
- Identical environment of dev and prod
- Portability
 - `git clone`
 - `vagrant up`
- Only one file to config



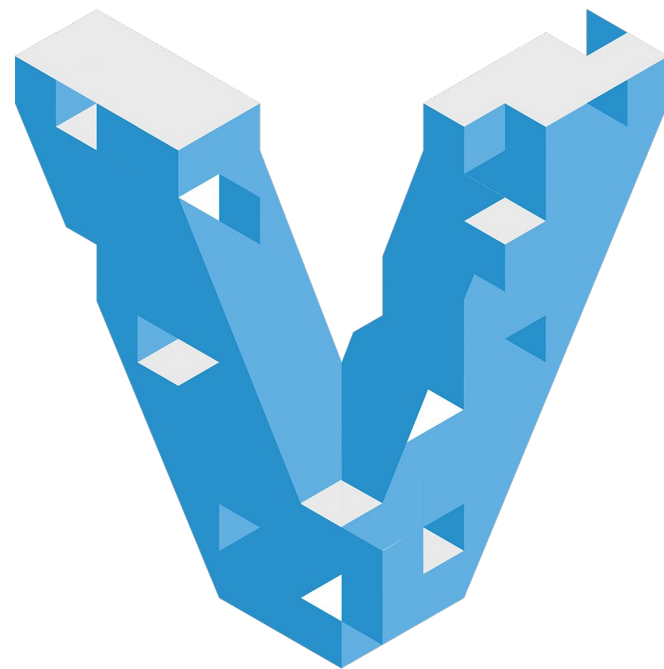
VAGRANT

Getting started

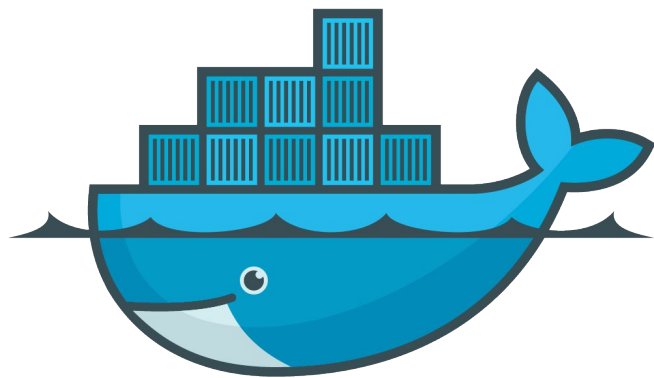
- Getting up <https://www.vagrantup.com/>

```
$ vagrant init ubuntu/trusty64
```

```
$ vagrant up --provision
```



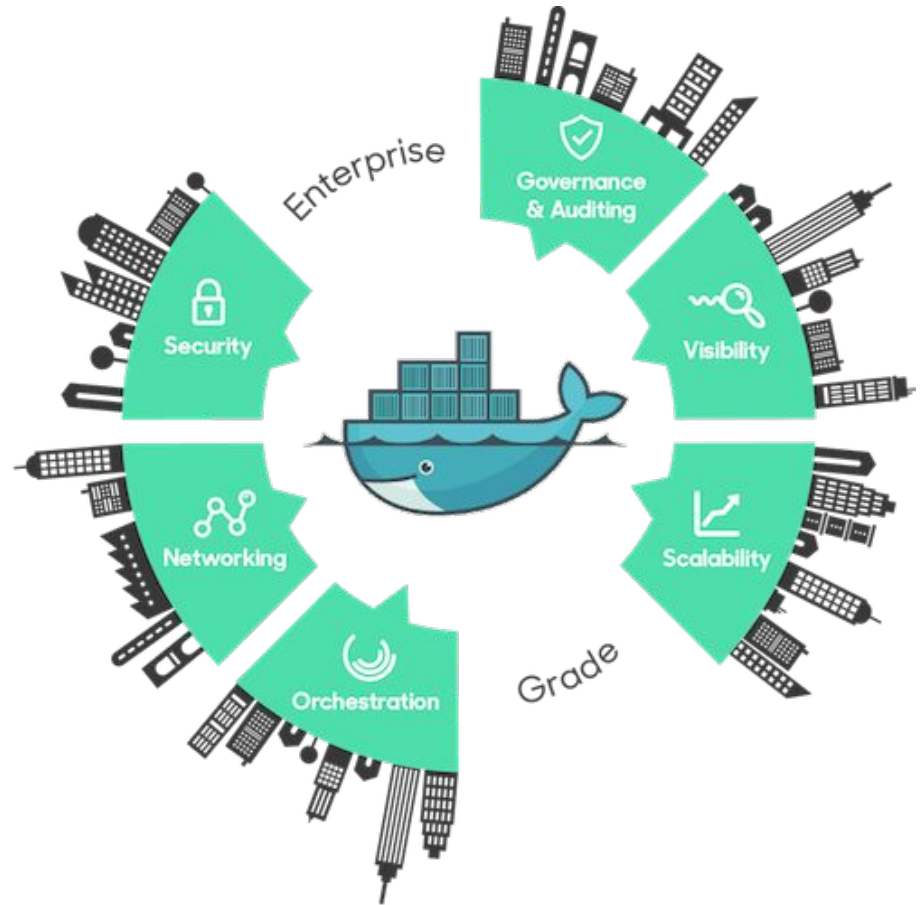
VAGRANT



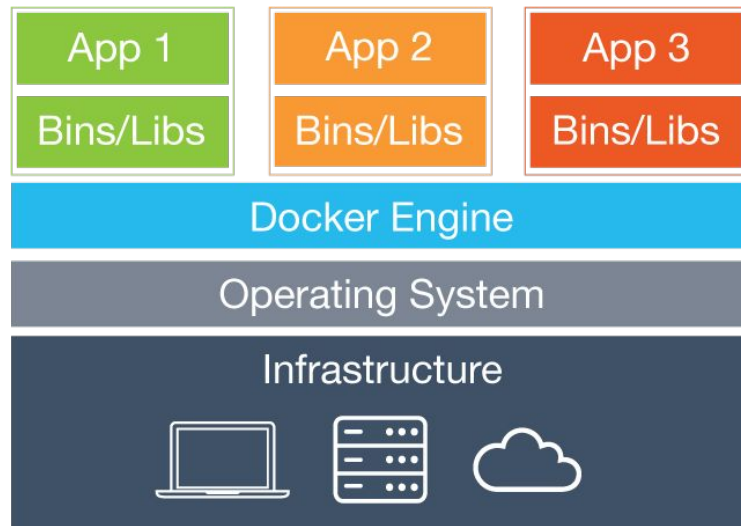
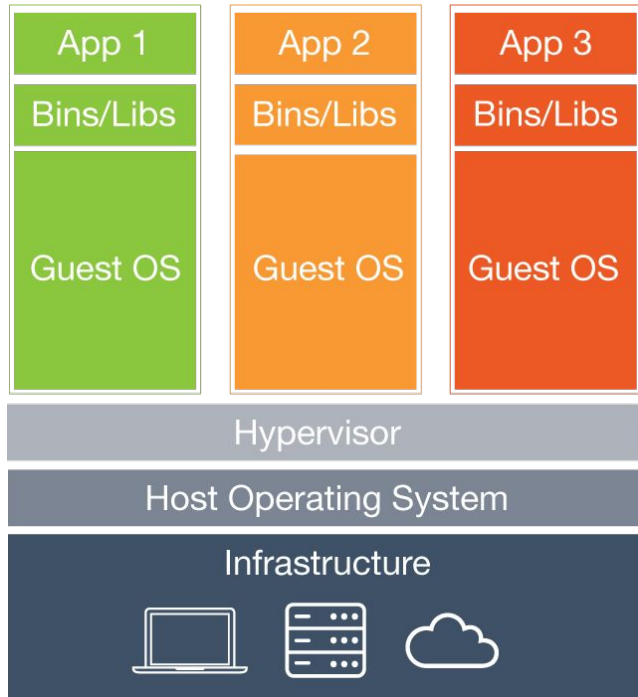
docker

What is Docker?

- Operating system level Virtualization
- Process isolated as a Linux Container
- Does not require or include a separate operating system



Difference between Docker & VMs



Philosophy of docker

- Build

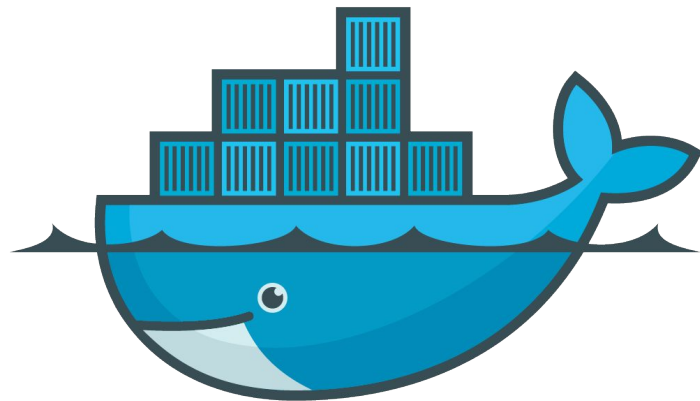
Docker allows you to compose your application from microservices, without worrying about inconsistencies between development and production environments, and without locking into any platform or language.

- Ship

Docker lets you design the entire cycle of application development, testing and distribution, and manage it with a consistent user interface.

- Run

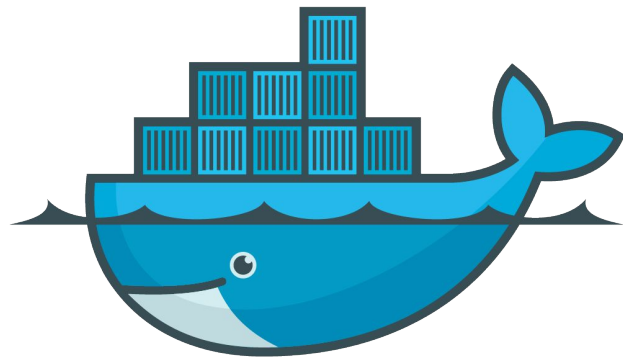
Docker offers you the ability to deploy scalable services, securely and reliably, on a wide variety of platforms.



docker

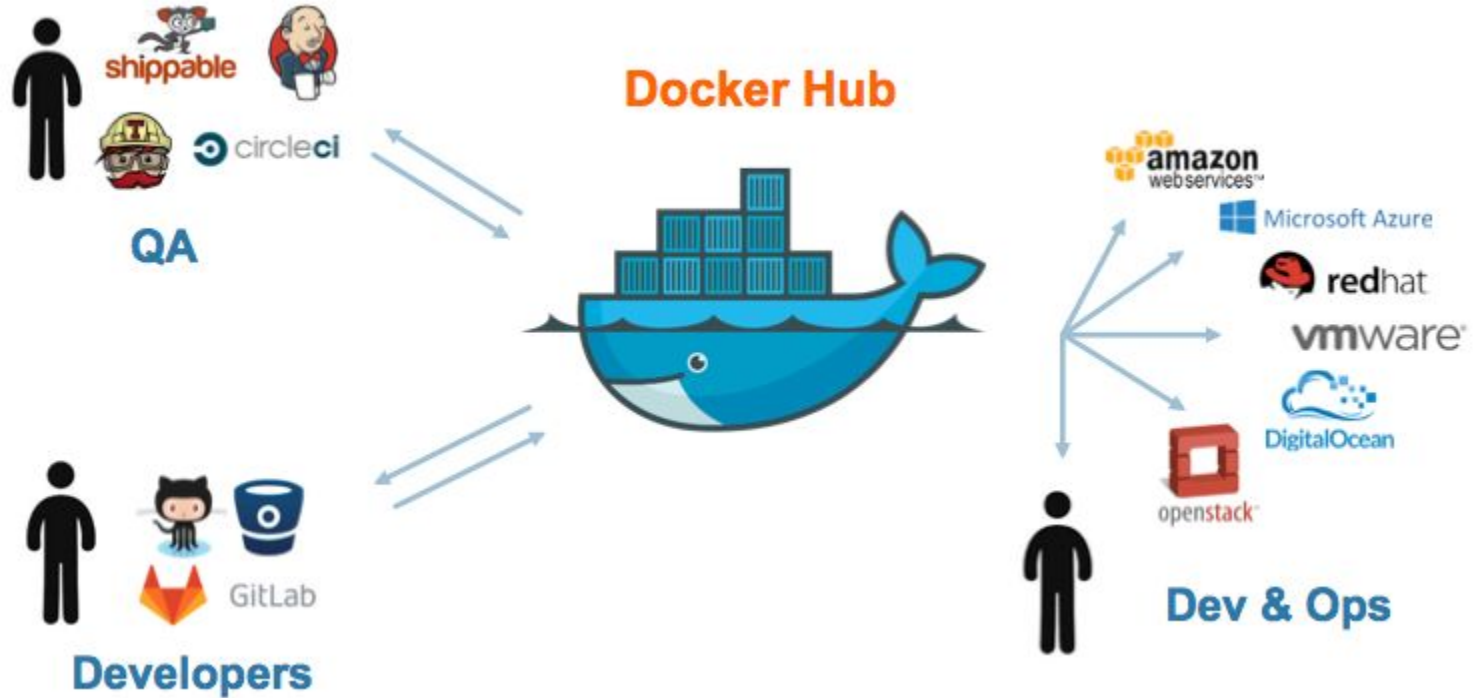
Play with docker

- Getting up docker toolbox at <https://www.docker.com/>
- `$ docker run hello-world`



docker

Docker flow





ANSIBLE

In beginning...

- Developers wrote code
- System admins deployed code



Until one day...

- I will write code to tell the computer how to set up itself.



CODING

Shell script

- Not “robust”
- Not idempotent
- Everyone’s rolling their own

```
#!/bin/sh

case "$(uname)" in
    "Darwin")
        echo "Mac OS X"
        ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
        brew install git mvim zsh tree
        ;;
    "Linux")
        sudo add-apt-repository -y ppa:numix/ppa
        sudo apt-get update
        sudo apt-get install -y git vim-gtk zsh tree
        sudo apt-get install numix-gtk-theme numix-icon-theme
        ;;
esac

# Set up runtime config...
cd ~/
[ -d dotfiles ] && rm -rf dotfiles
git clone https://github.com/duythinhht/dotfiles.git
cd ~/dotfiles

while read f; do
    echo "Copying file $f..."
    cp -rf $f ~/
done < Dotfiles
echo "Copy files done!"

mkdir -p ~/.themes
cp -rf iris-light ~/.themes
```

What is ansible?

- IT automation tool (Python)
- Free, Open Source
- Simply
 - Clear
 - Fast (learn, setup)
 - Complete
 - Efficient (run on OpenSSH)
 - Secure (without agents)



ANSIBLE

Use cases?

- Server provisioning
- Application deploy
- Rollback a version
- Continuous Integration



ANSIBLE

Getting up

- Install over pip

```
$ sudo pip install ansible
```

- Checkout the documents

```
http://docs.ansible.com/
```



ANSIBLE

Ansible structure

- Inventory
- Modules
- Ad-hoc commands
- Playbooks
 - Tasks
 - Variables
 - Templates
 - Handlers
 - Roles

Let's do a small things

Thank you!