Docker

Brian Chirgwin
Github: bchirgwin

What is Docker?

- open platform for developers and sys admins
- Build, ship, and run distributed applications.
- Run the same app, unchanged, on laptops, data center VMs, and any cloud.
- Based on Linux LXC (Linux Container) http://en.wikipedia.org/wiki/LXC

Some Docker Users

yelp

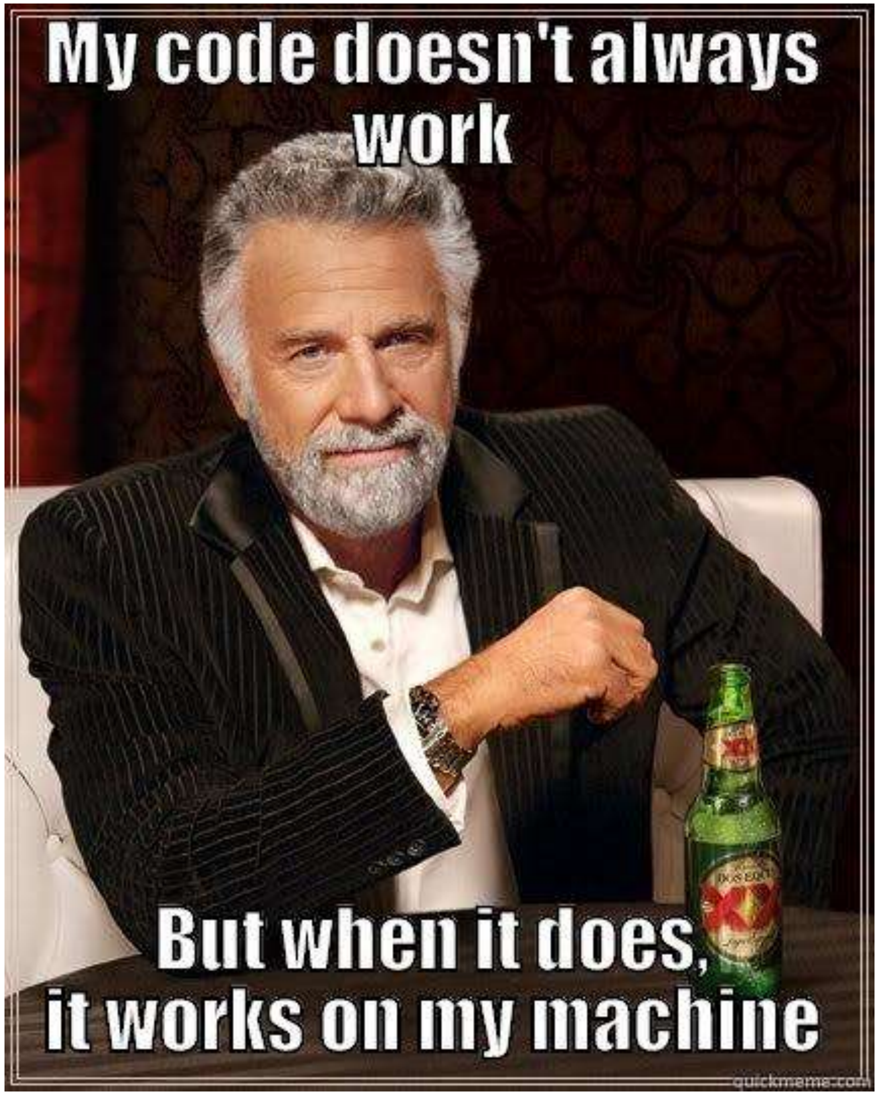
Spotify

rackspace
the #1 managed cloud company

Cambridge Healthcare

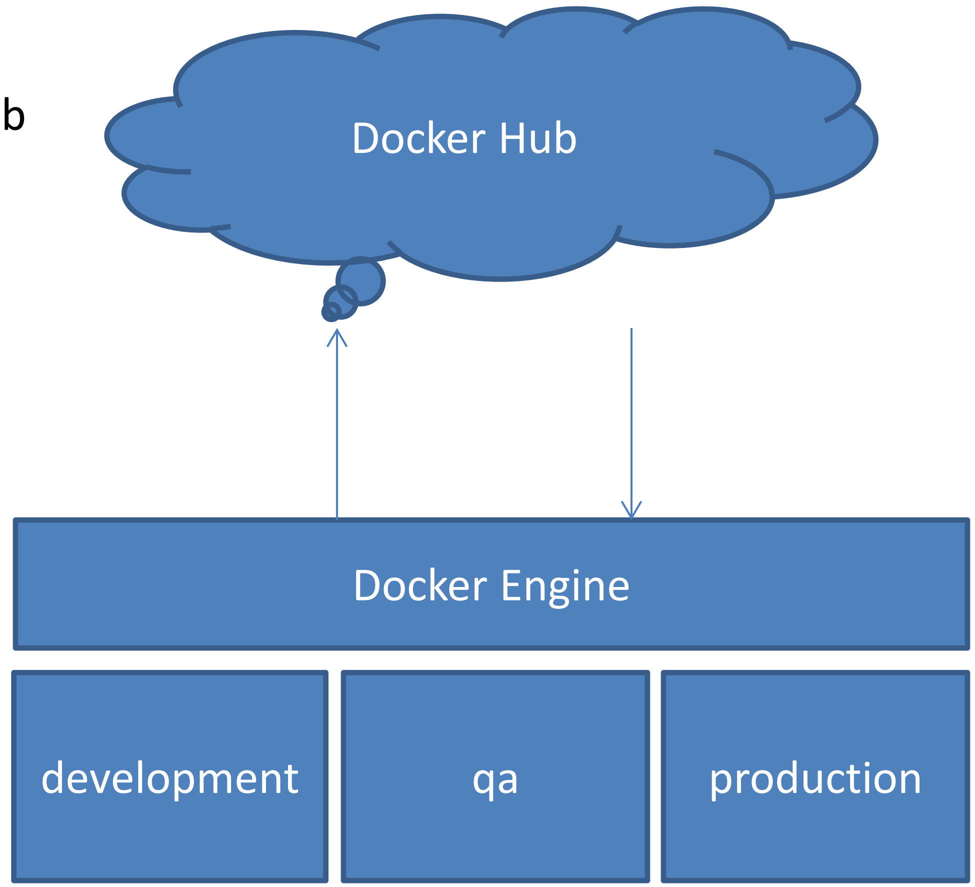
ebay

https://github.com/disney/docker-training


My code doesn't always work
But when it does, it works on my machine
quickmeme.com

Works Everywhere

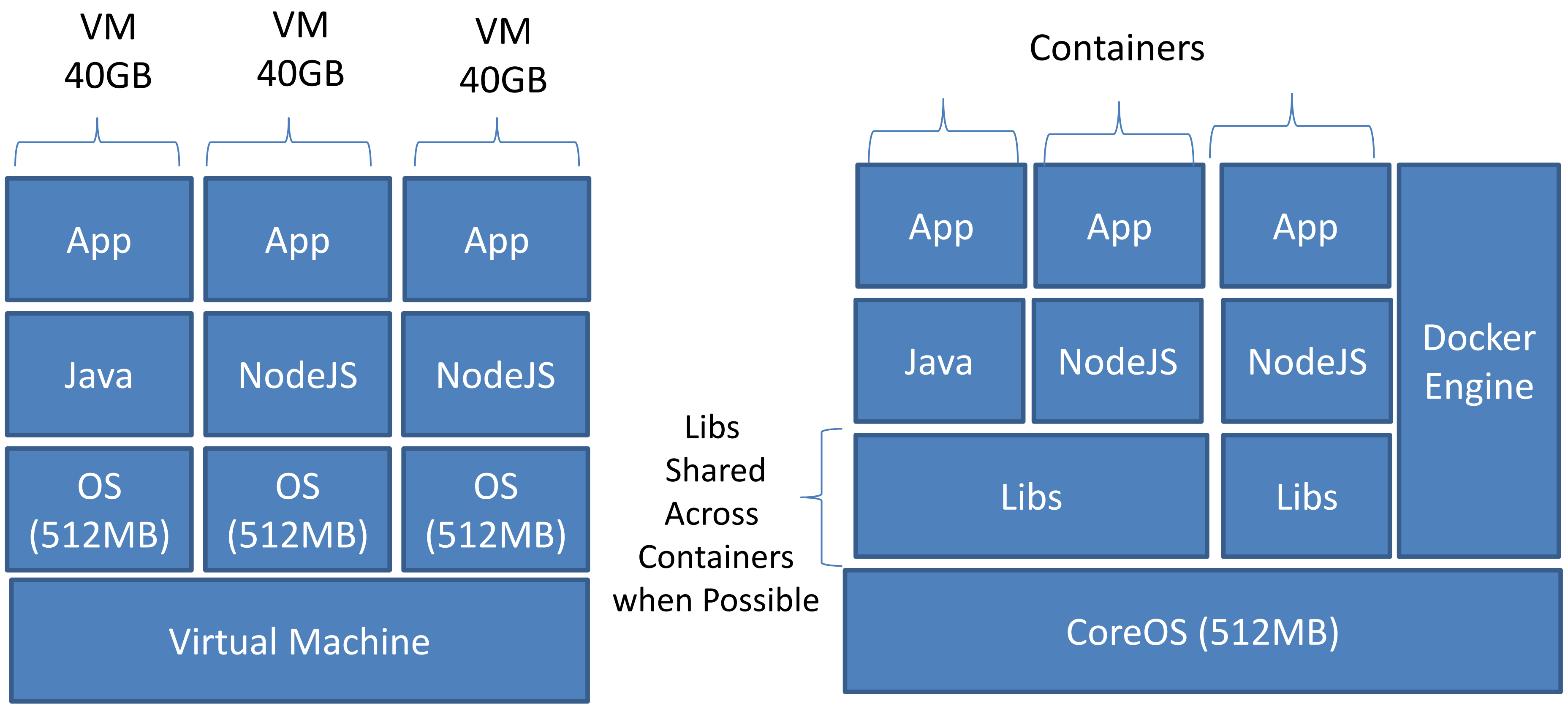
2. Push to Docker Hub



1. Build docker image on development

3. Pull image from Docker Hub for other systems

Docker VS VM



Memory: 1.5 GB
System Drive: 120 B

Memory: 512 MB GB
System Drive: ~ size of app and libs

Build Process

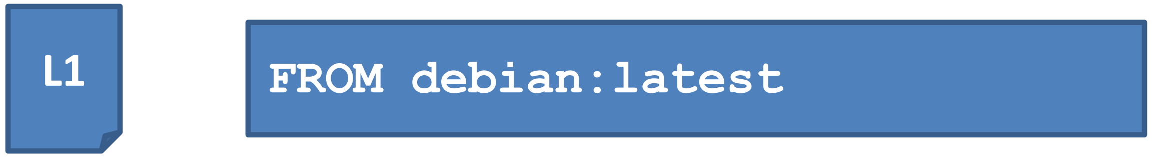
- Write a Dockerfile script
- Use Docker Engine to Build Image
- Run Image
- Add script to version control
- Push Image for others to pull

Dockerfile (line by line)

`FROM debian:latest`

```
This image will be based on debian docker image.
Debian image will be pulled from docker hub if it
doesn’t exist locally.
```

```
Each Dockerfile command creates a new layer. Layer
compared with previous layer and difference stored.
```



Dockerfile (line by line)

`MAINTAINER` ***`name <email>`***

`Specify maintainer of container`

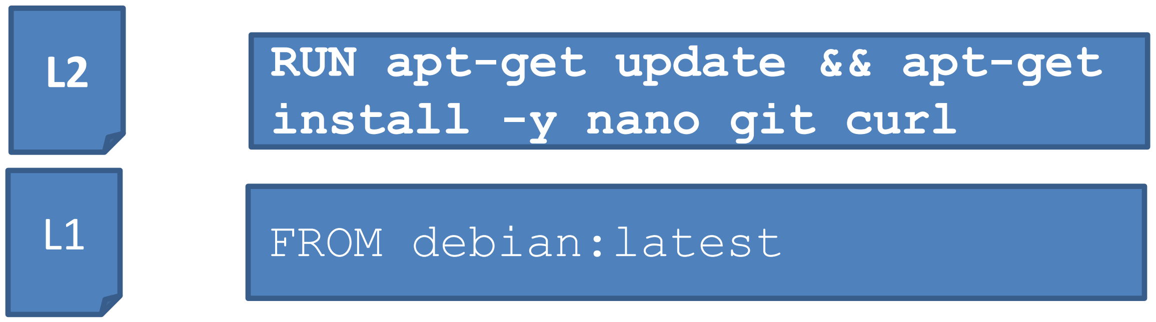
L1

```
FROM debian:latest
```

Dockerfile (line by line)

`RUN apt-get update && apt-get install -y nano git curl`

```
execute apt-get update and install
      nano
      git
      curl
```

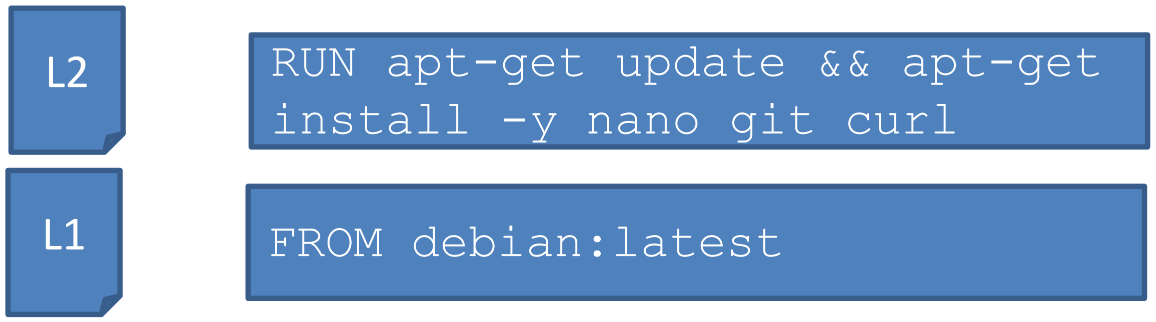


Dockerfile (line by line)

`VOLUME ["/data/db","config.js"]`

```
Define external volumes directory /data/db
file config.js

When container is executed location of /data/db and
config.js can be specified
```



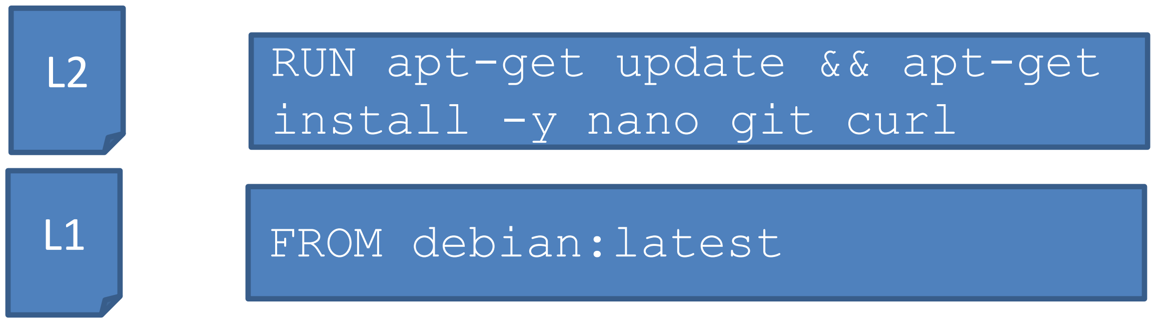
```
RUN apt-get update && apt-get
install -y nano git curl
```

```
FROM debian:latest
```

Dockerfile (line by line)

```
EXPOSE 8080 28017
```

```
Expose ports outside of container.
  8080
  28017
```



```
RUN apt-get update && apt-get
install -y nano git curl
```

```
FROM debian:latest
```

Dockerfile (Complete)

```
FROM debian:latest
MAINTAINER name <email>

RUN apt-get update && apt-get install -y nano git curl

VOLUME ["/data/db"]

EXPOSE 27017
EXPOSE 28017
```

Expose Ports - Link Containers

Client Browser

Port 9000 exposed to docker containers that are linked only

Port 8080 exposed outside container on port 80 with –p 80:8080

nginx

EXPOSE 8080

app

EXPOSE 9000

db

VOLUME /data

docker run –name dbServer -v /c/data:/data db
Execute db container with c:\data on the host used as volume /data (in container)

docker run –name appServer –link dbServer:db app
Execute app named AppServer with link to dbServer container

docker run –name webServer –link appServer:app -p 80:8080 nginx
Execute nginx container named webServer link to appServer. Expose port 8080 outside of container

Build Image

- **docker build -t "brianc/app" .**

Build image tagging (-t) image name as brianc/app. Use current directory (.) for Dockerfile location

Run image

- Detached

docker run --name *runningContainerName* -d *imagename cmdToExec*

-d : detached, keeps container running

- Interactive

Docker run -t -i *imagename /bin/bash*

-t : allocate a psuedo tty

-i: run interactive

Share image

- **Using Docker Hub Repository**
  - docker **pull** centos
  - docker **push** yourname/imagename
- **By Tar file**
  - save container
    - docker save -o brianc-test.tar brianc/test
  - load container
    - docker load -i brianc-test.tar

Common Practice

- Create data-only containers for persistent databases, configuration files, data files, etc…
  - Use **volume-from** command

Debug

- Docker logs <name>

List logs of container

Future

- Docker for Windows coming

Docker Cloud Hosting

- https://www.dotcloud.com
- https://www.tutum.co (free while in beta)
- https://stackdock.com
- https://quay.io

Boot2Docker

- Lightweight linux, based on tiny core linux, to run docker containers
- Virtual Box on Mac and Windows

Boot2Docker is a VM and has its own IP address.

http://boot2docker.io

CoreOS

- Lightweight Linux OS for running docker containers
- Cluster
- Painless Updating

https://coreos.com/

Vagrant

- For creating development environments
- Docker provisioner
- Docker provider

https://www.vagrantup.com/