Docker on a developer's desktop
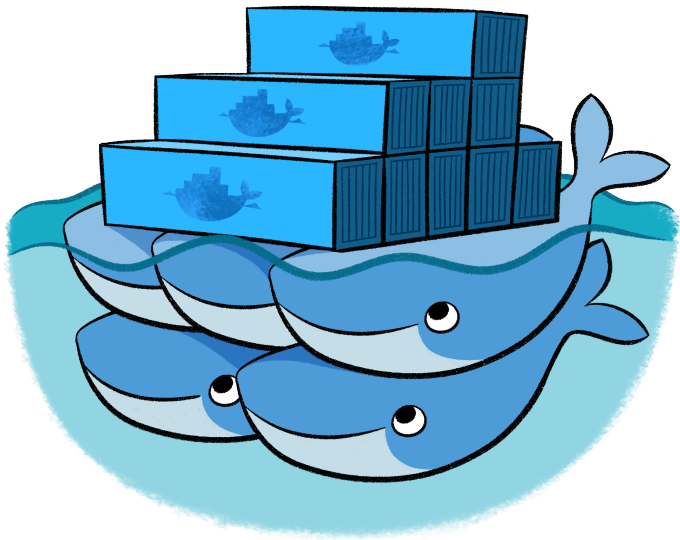
JavaCamp #13

Łukasz Proszek
proszek@gmail.com

2015-02-12

Common theme

- It runs on my machine.

Common theme

- It runs on my machine.
- We don't ship your machine!

Common theme

- It runs on my machine.
- We don't ship your machine!
- Windows on a workstation, Linux on the server

Common theme

- It runs on my machine.
- We don't ship your machine!
- Windows on a workstation, Linux on the server
- Different configuration on each env

Common theme

- It runs on my machine.
- We don't ship your machine!
- Windows on a workstation, Linux on the server
- Different configuration on each env
- Technical gap between teams:

Common theme

- It runs on my machine.
- We don't ship your machine!
- Windows on a workstation, Linux on the server
- Different configuration on each env
- Technical gap between teams:
- don't expect a tester to configure a JDBC driver

Common theme

- It runs on my machine.
- We don't ship your machine!
- Windows on a workstation, Linux on the server
- Different configuration on each env
- Technical gap between teams:
- don't expect a tester to configure a JDBC driver
- don't expect your sysadmin to debug your code

Common theme

- It runs on my machine.
- We don't ship your machine!
- Windows on a workstation, Linux on the server
- Different configuration on each env
- Technical gap between teams:
- don't expect a tester to configure a JDBC driver
- don't expect your sysadmin to debug your code
- don't expect your developer to configure the routing

Common theme

- It runs on my machine.
- We don't ship your machine!
- Windows on a workstation, Linux on the server
- Different configuration on each env
- Technical gap between teams:
- don't expect a tester to configure a JDBC driver
- don't expect your sysadmin to debug your code
- don't expect your developer to configure the routing
- why does it build on machine A and not on B nor C

Virtualization

- A resource hog

Virtualization

- A resource hog
- Disk

Virtualization

- A resource hog
- Disk
- Memory

Virtualization

- A resource hog
- Disk
- Memory
- Slightly Awkward to configure

Virtualization

- A resource hog
- Disk
- Memory
- Slightly Awkward to configure
- Once set up, begins to deteriorate

Virtualization

- A resource hog
- Disk
- Memory
- Slightly Awkward to configure
- Once set up, begins to deteriorate
- PITA to redeploy a clean environment

Virtualization

- A resource hog
- Disk
- Memory
- Slightly Awkward to configure
- Once set up, begins to deteriorate
- PITA to redeploy a clean environment
- I'm quite confident that I will spin up a full docker env in the time that takes you to spin up one vmdk.
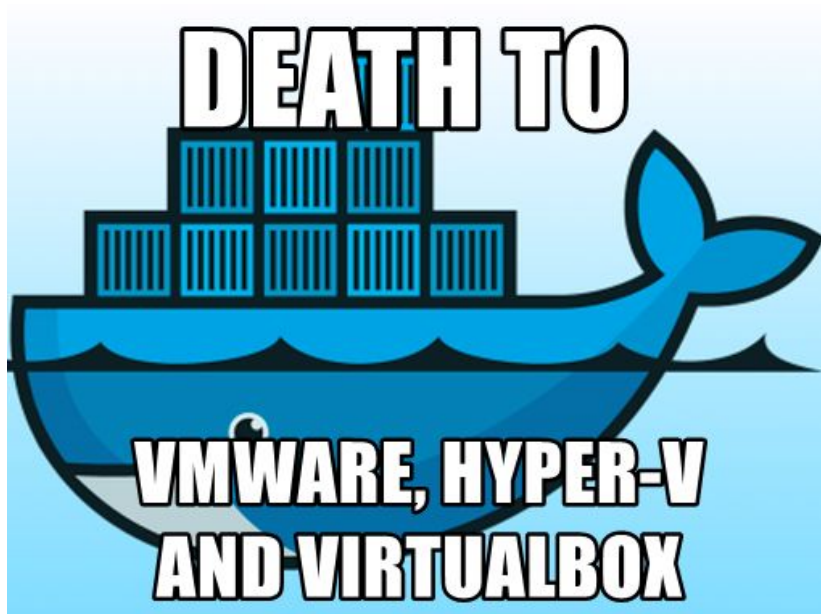
Common Advantages

- All containers share the same base image - less download

Common Advantages

- All containers share the same base image - less download
- Updates are just diff snapshots

Common Advantages

- All containers share the same base image - less download
- Updates are just diff snapshots
- Semi platform independent

Common Advantages

- All containers share the same base image - less download
- Updates are just diff snapshots
- Semi platform independent
- Provides a comfortable level of isolation

Docker is like a Ogre: it has layers
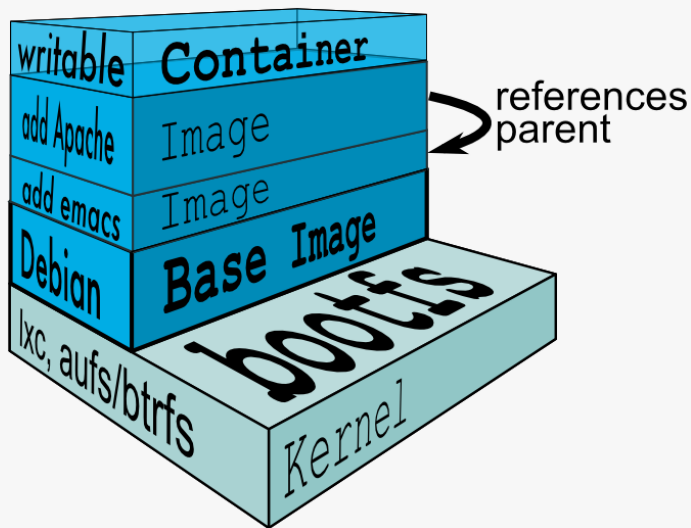
On a desktop?

- ▶ Run insecure applications in an isolated, yet seamless environment

On a desktop?

- ▶ Run insecure applications in an isolated, yet seamless environment
- ▶ Run the same scenario on the same configuration, on different app versions, at the same time

On a desktop?

- ▶ Run insecure applications in an isolated, yet seamless environment
- ▶ Run the same scenario on the same configuration, on different app versions, at the same time

- ▶

Webbrowser with an insecure plugin - Dockerfile

```
FROM debian
RUN apt-get update && apt-get install -y iceweasel
RUN export uid=1000 gid=1000
RUN mkdir -p /home/developer
RUN echo "developer:x:${uid}:${gid}:Developer,,,:/h
RUN echo "developer:x:${uid}:" >> /etc/group && \
RUN chown ${uid}:${gid} -R /home/developer
RUN wget --no-check-certificate https://swdl.blueje
RUN dpkg -i bjnplugin_2.90.399.5-1_amd64.deb
USER developer
ENV HOME /home/developer
CMD /usr/bin/iceweasel
```

```
docker build −t bj − < Dockerfile.bj
docker run \
    −−env=DISPLAY=$DISPLAY \
    −−volume=/tmp/.X11−unix:/tmp/.X11−unix \
    −−rm \
    −−tty \
    bj
```

Endless possibilities

Exaplme docker code

```
1  func init() {
2      dispatch = map[string]func(string) ....
3          "user"      : parseString,
4          "onbuild"   : parseSubCommand,
5          "workdir"   : parseString,
6          "env"       : parseEnv,
7          "maintainer": parseString,
8          "from"      : parseString,
9          "add"       : parseMaybeJSONToList,
10         "copy"      : parseMaybeJSONToList,
11         "run"       : parseMaybeJSON,
12         "cmd"       : parseMaybeJSON,
13         "entrypoint": parseMaybeJSON,
14         "expose"    : parseStringsWhitespaceDelimited,
15         "volume"    : parseMaybeJSONToList,
16         "insert"    : parseIgnore,
17  }
```

Thanks @GrzegorzNosek

Thank You