



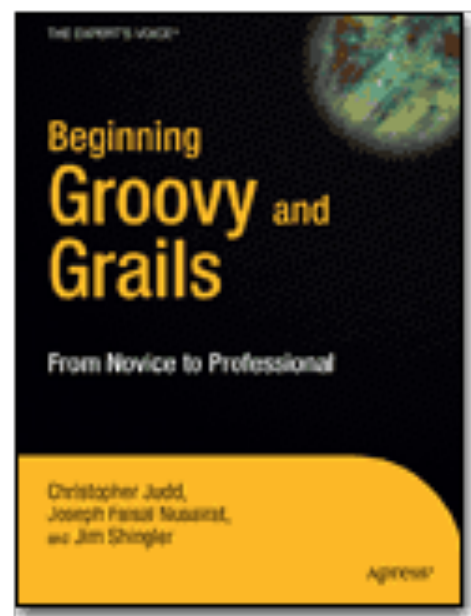
Christopher M. Judd

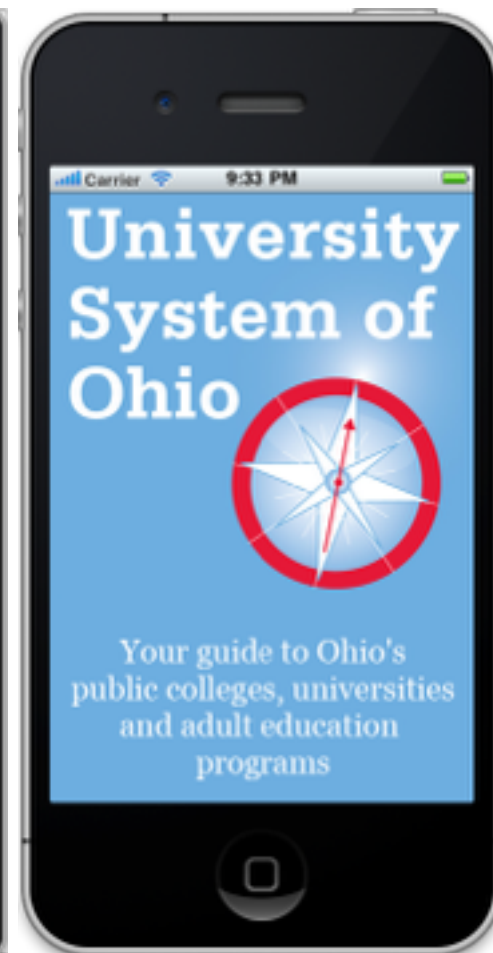
Christopher M. Judd

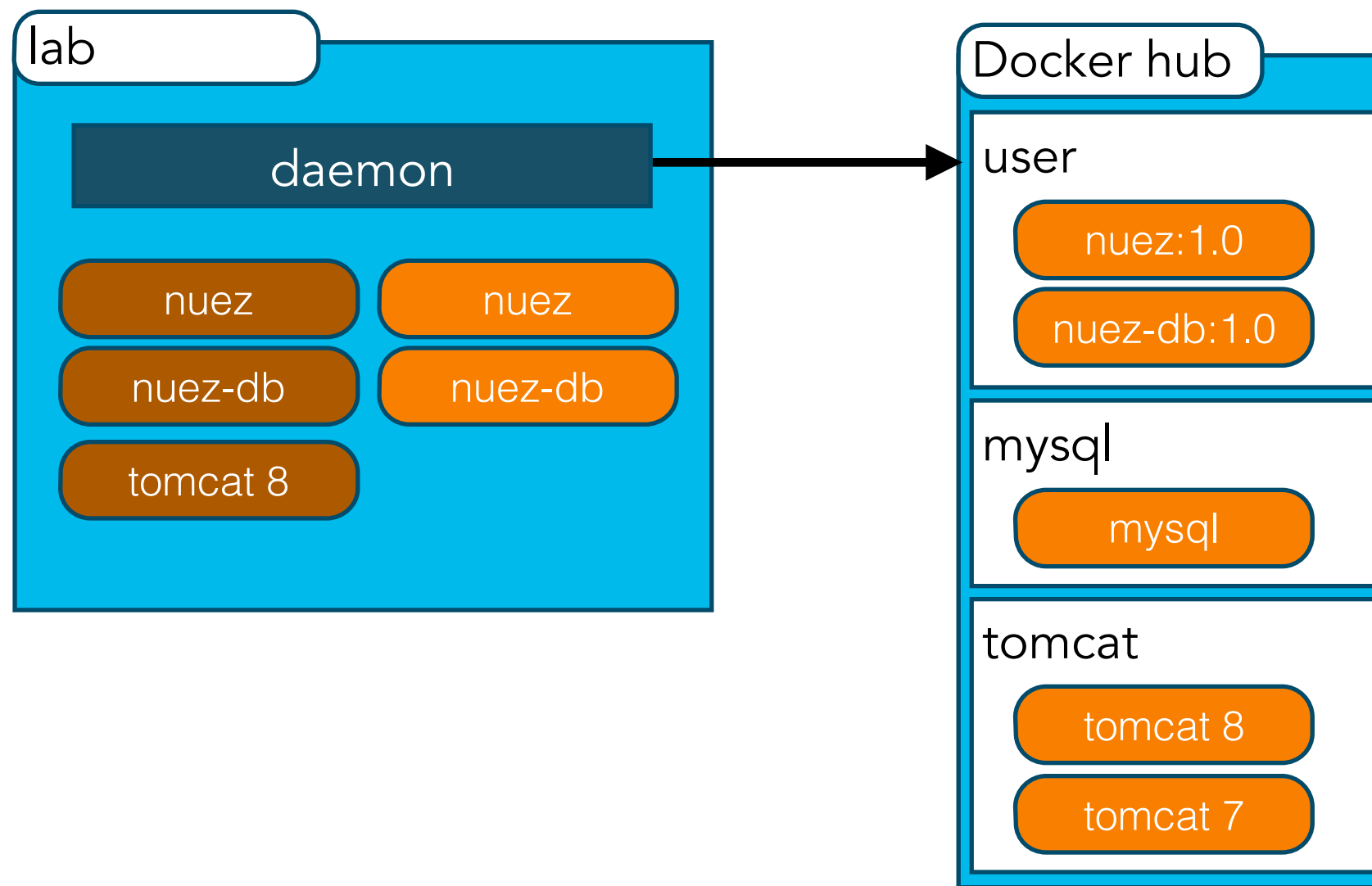
CTO and Partner at

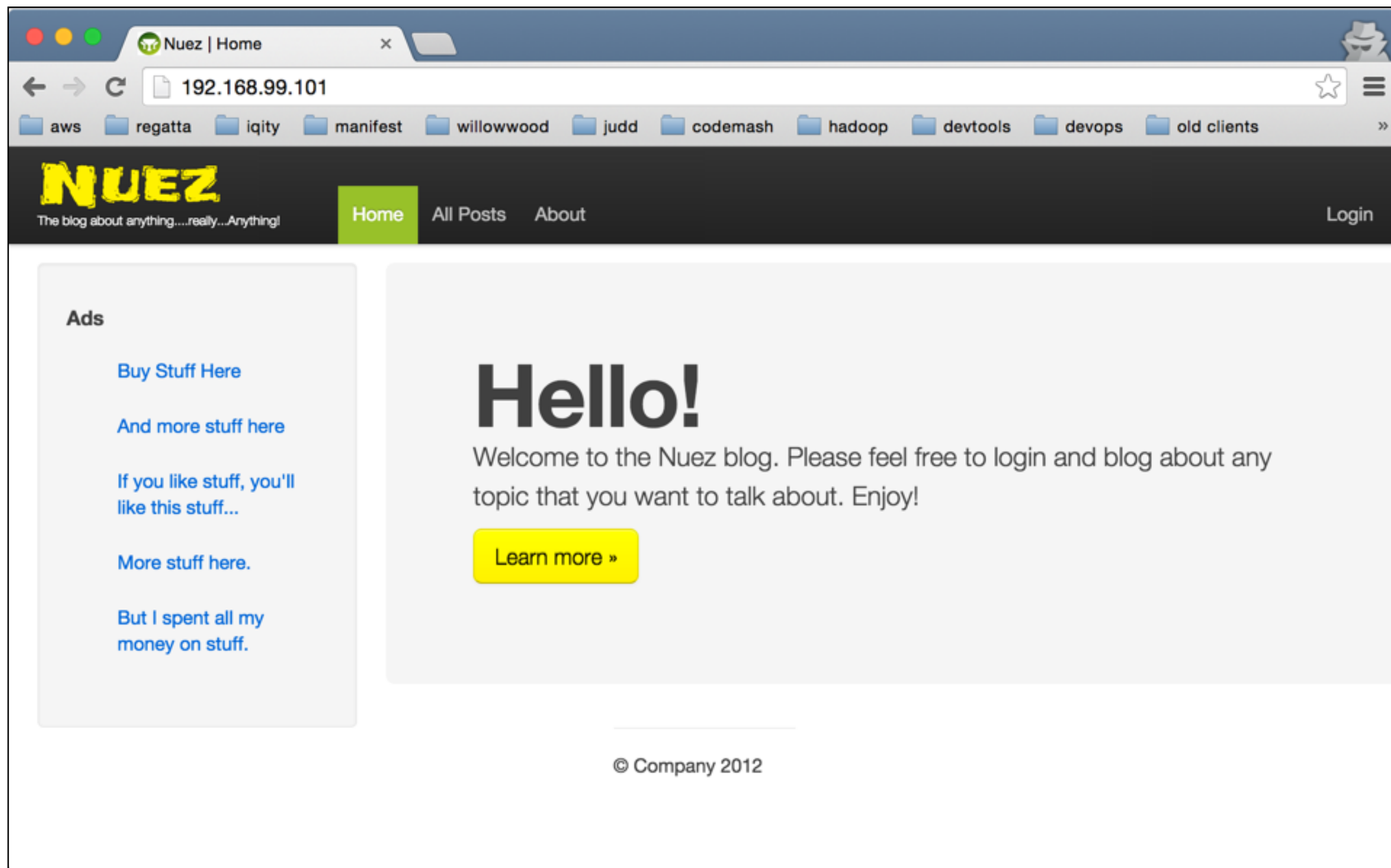


Central Ohio Java Users Group leader











[Pull requests](#) [Issues](#) [Gist](#)



[Contributions](#) [Repositories](#) [Public activity](#) [Edit profile](#)

[Search](#) [All](#) [Public](#) [Private](#) [Sources](#) [Forks](#) [Mirrors](#) [New](#)

Christopher M. Judd

cjudd

 Judd Solutions

 Columbus Ohio

 github@juddsolutions.com

 <http://juddsolutions.blogspot.c...>

 Joined on Jan 3, 2009

32

Followers

16

Starred

0

Following

Organizations

nuez-compose

Docker composition for the [nuez-db](#) and [nuez](#) web application.

Updated 44 minutes ago

nuez-docker

An example of a docker file that creates an image for [nuez](#) from a URL.

Updated 20 hours ago

nuez

Y forked from [zendern/nuez](#)

Updated 21 hours ago

Groovy ★ 1 0 3

nuez-db

Docker configuration for [nuez-db](#)

Updated a day ago

<https://github.com/cjudd/nuez>
<https://github.com/cjudd/nuez-db>
<https://github.com/cjudd/nuez-docker>
<https://github.com/cjudd/nuez-compose>



<https://s3.amazonaws.com/cmj-presentations/docker-cloudddevelop-2015/index.html>

MY JOURNEY TO ENLIGHTENMENT



FreeK Box

FreeK Box

Start

Ivan 



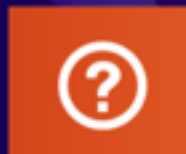
Skype



Reading List



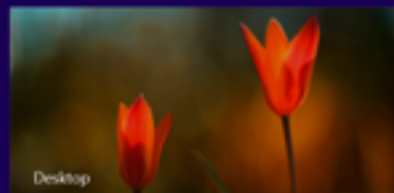
Internet Explorer



Help + Tips



SkyDrive



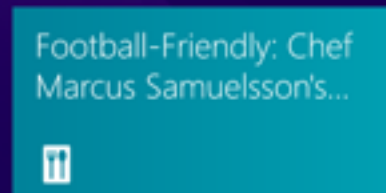
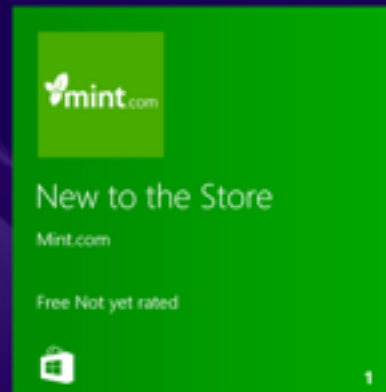
Desktop



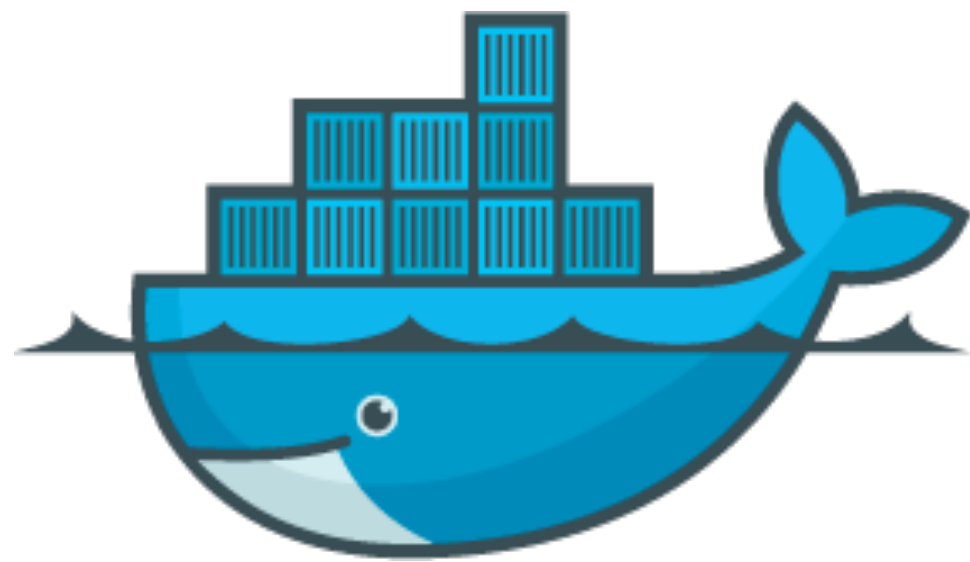
Weather



Photos



Maps



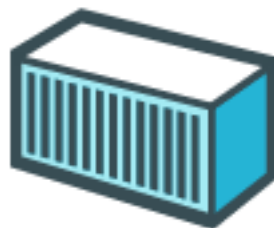
docker



An open platform for distributed applications for developers and sysadmins



Build



Ship

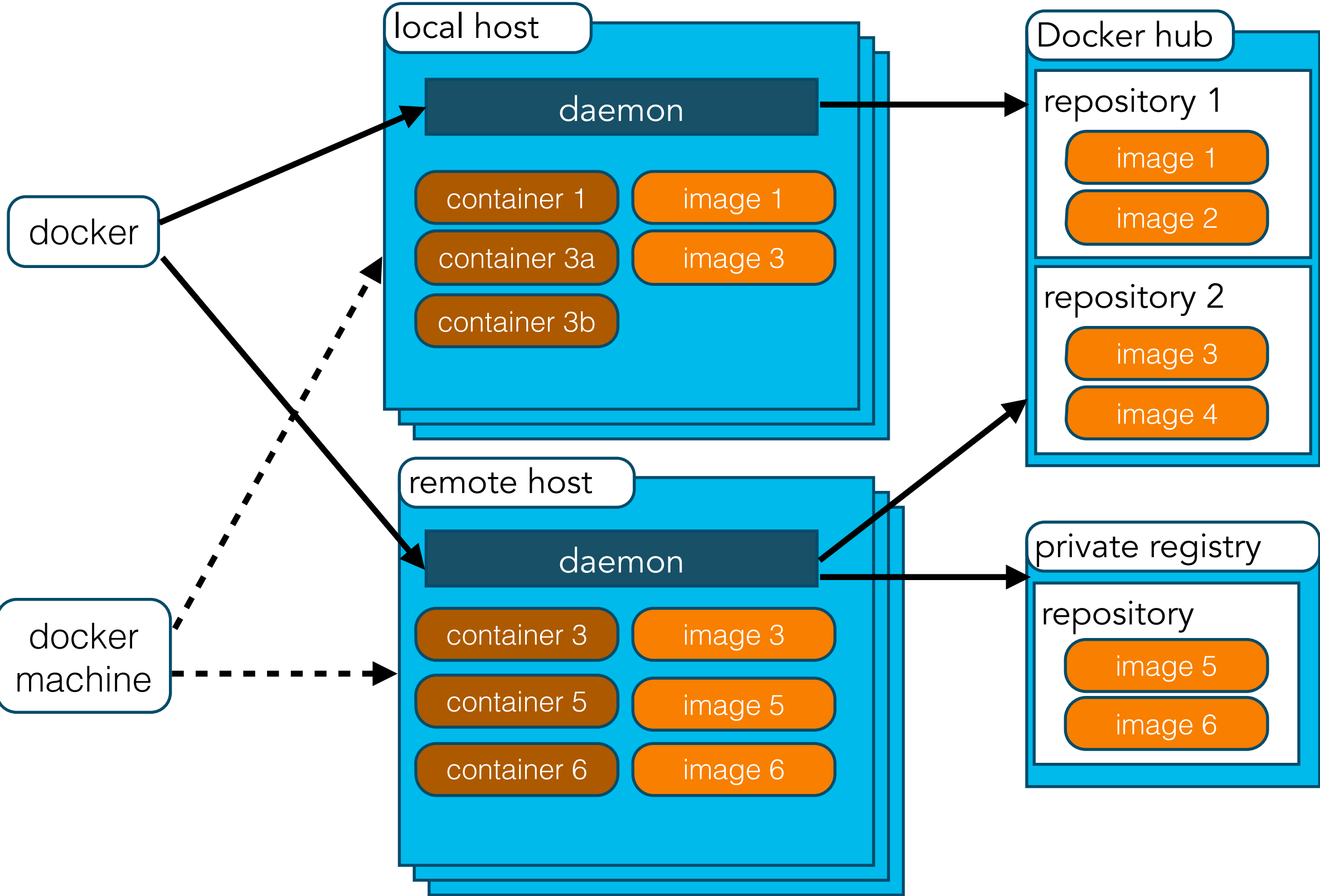


Run

Clients

Hosts

Registries





- 📌 infrastructure automation
- 📌 sandboxing/isolation
- 📌 maintain consistency
- 📌 better resource utilization
- 📌 easy experimentation

infrastructure automation

Run the following commands

1. `apt-get mysql` unless you are on CentOS
2. `vim /etc/timezone`
and add `America/New_York`
3. `vim /etc/mysql/my.cnf`
and add `federated`
and add `lower_case_table_names = 1`

infrastructure automation

```
FROM mysql:5.5.45
```

```
COPY my-custom-entriypoint.sh /
```

```
COPY docker-entriypoint-initdb.d /docker-entriypoint-initdb.d/
```

```
RUN echo America/New_York | tee /etc/timezone
```

```
&& dpkg-reconfigure --frontend noninteractive tzdata
```

```
RUN echo "federated" >> /etc/mysql/my.cnf
```

```
RUN echo "lower_case_table_names = 1" >> /etc/mysql/my.cnf
```

```
ENTRYPOINT ["/my-custom-entriypoint.sh"]
```

```
CMD ["mysqld"]
```

sandboxing/isolation

apache-maven-2.2.1
apache-maven-3.0.4
apache-maven-3.2.1
apache-tomcat-5.5.17
apache-tomcat-6.0.29
apache-tomcat-7.0.37
apache-tomcat-8.0.21
apache-tomcat-8.0.27
grails-1.3.5
grails-1.3.6
grails-1.3.7
grails-2.0.0
grails-2.1.1
grails-2.1.4
grails-2.2.1
grails-2.2.2
grails-2.3.10
groovy-1.7.10
groovy-2.0.5
groovy-2.1.1
groovy-2.2.2
jdk-1.6.0_65-b14-462
jdk-1.7.0_45-b18
jdk-1.8.0_25-b17
mysql-5.6.15

sandboxing/isolation

client 1/project 1

daemon

jvm 1.7/tc 7

jvm 1.7/ tc 7

jvm 1.8/tc 8

jvm 1.8/tc 8

mysql 5.5

mysql 5.5

client 2/project 2

daemon

nginx 1.95

nginx 1.95

node 4.2.1

node 4.2.1

mongoDB 3

mongoDB 3

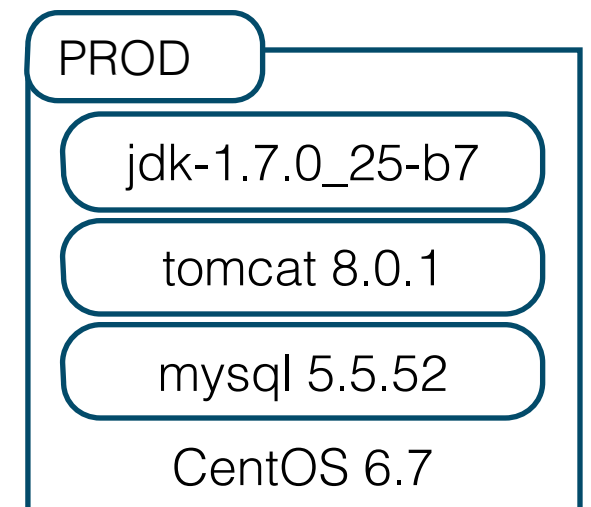
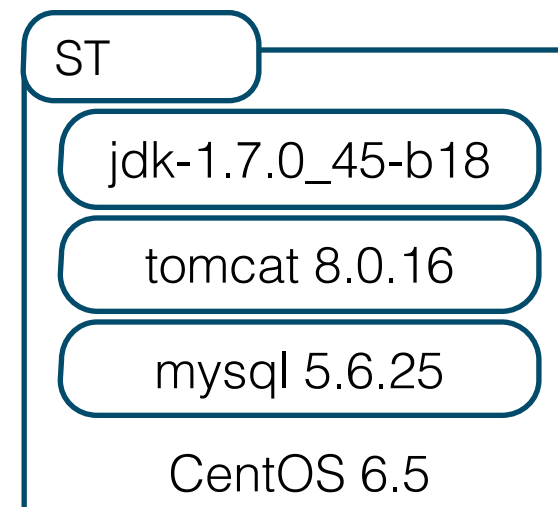
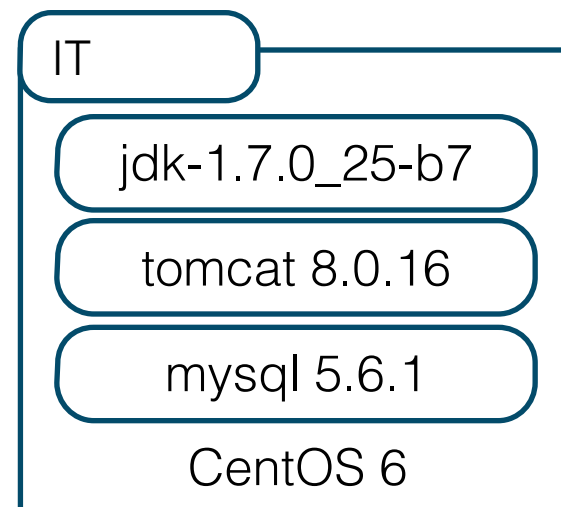
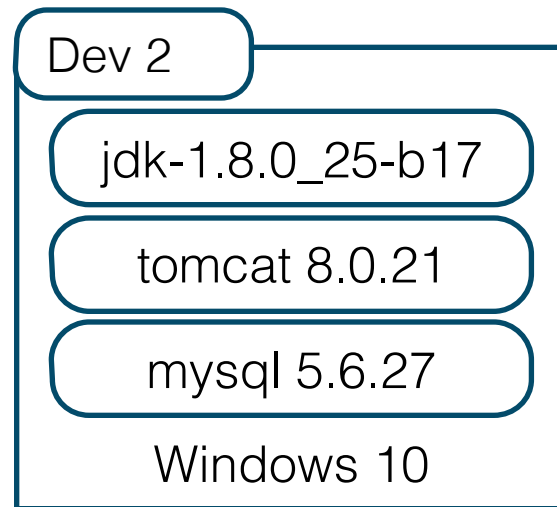
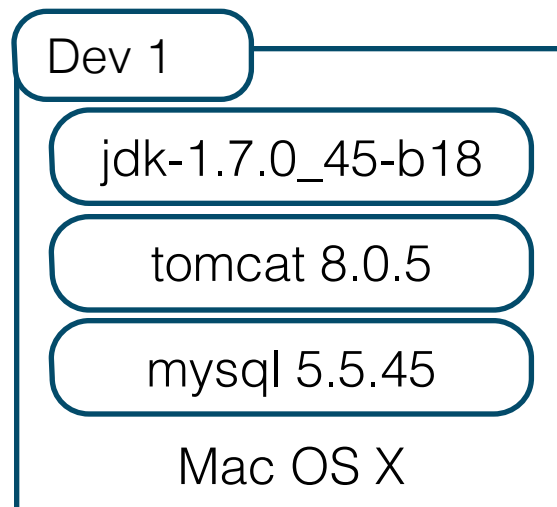
mysql 5.6

mysql 5.6

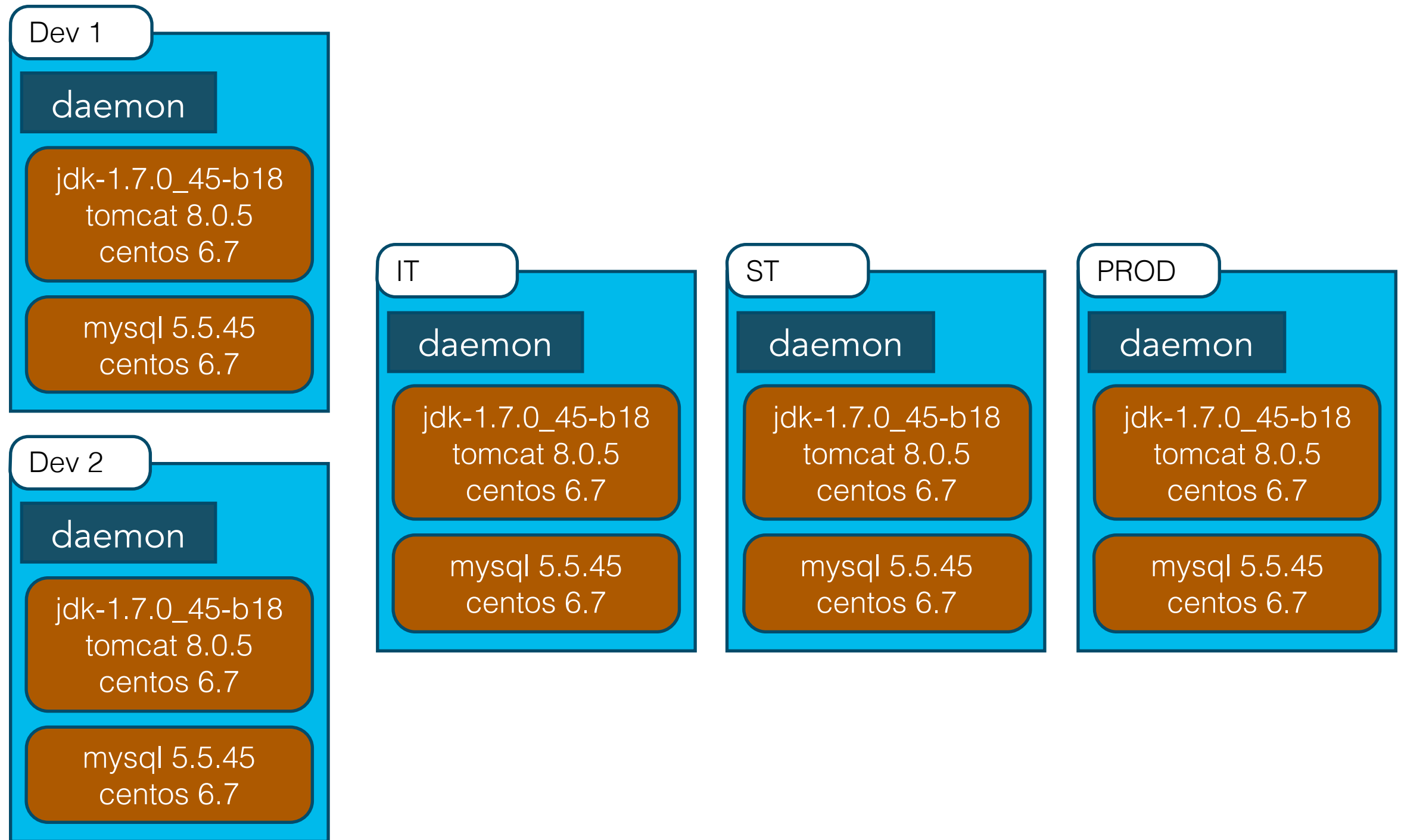
jvm 1.6

jvm 1.6

maintain consistency

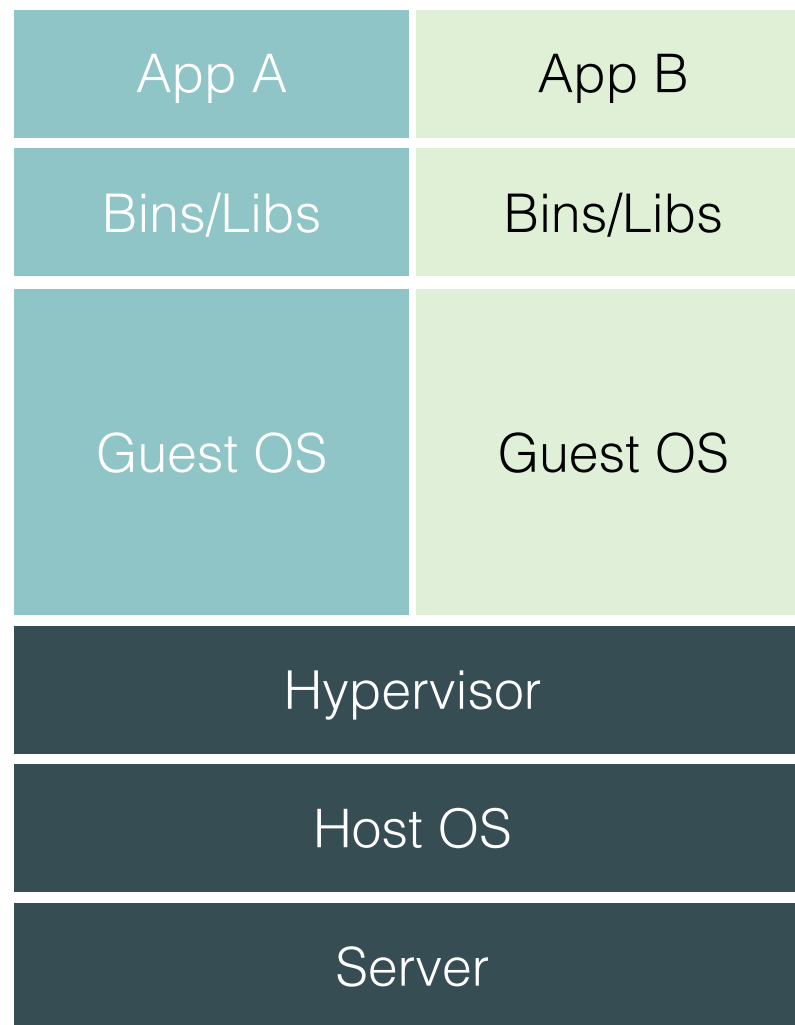


maintain consistency

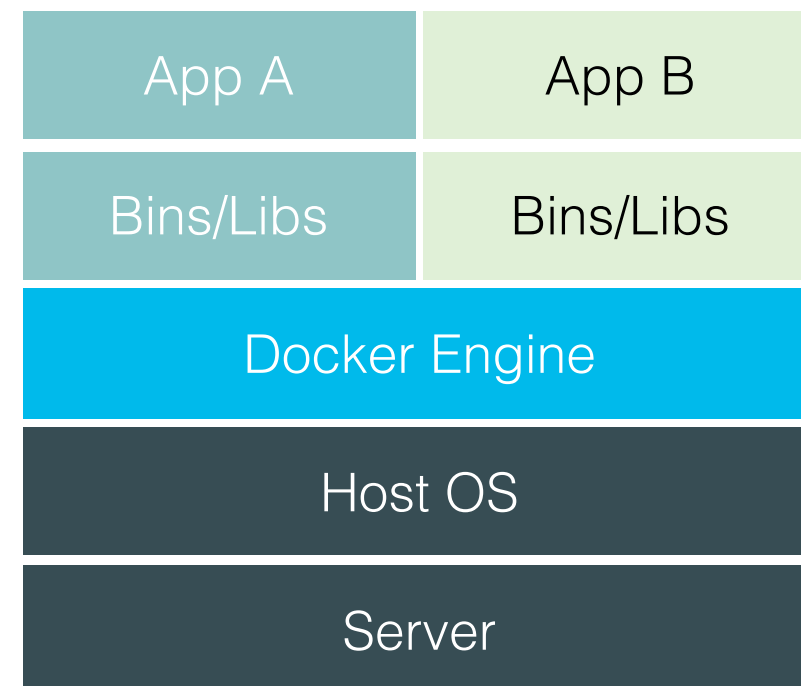


better resource utilization

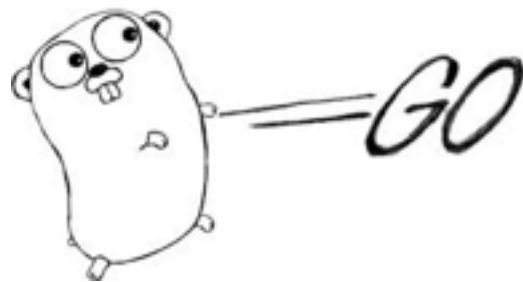
Virtual Machine



Docker



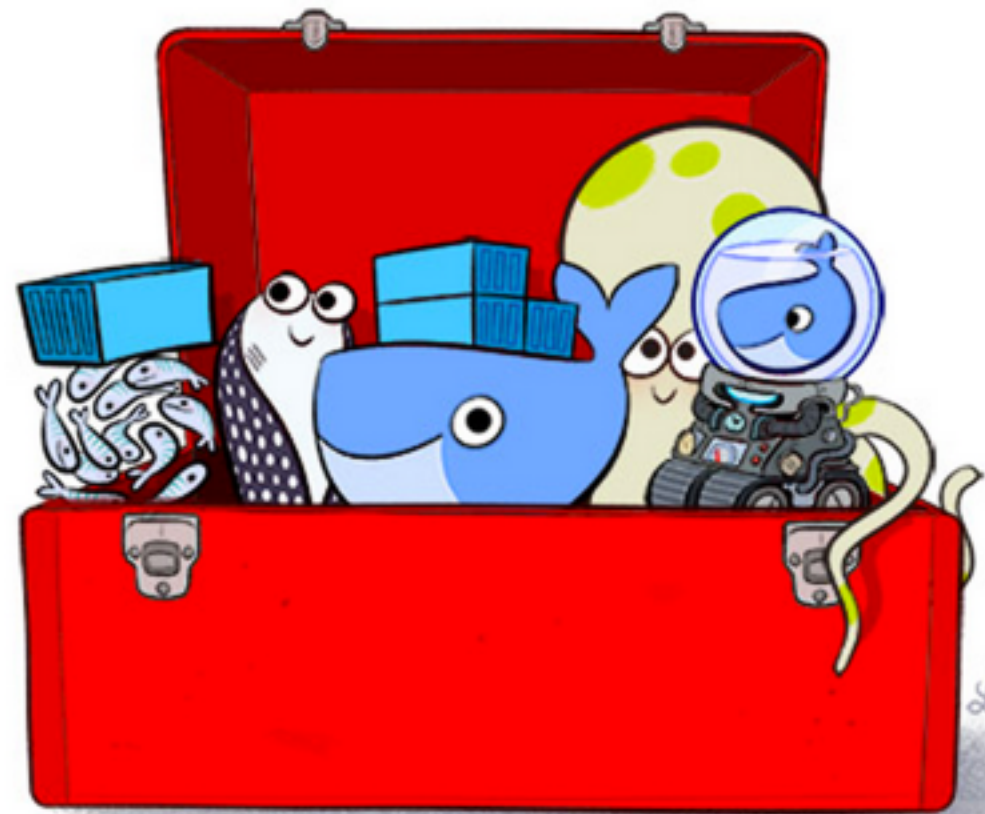
easy experimentation



SETUP DOCKER

Docker Toolbox

[Getting Started Guide \(Mac\)](#) | [Getting Started Guide \(Windows\)](#) | [Contribute to Toolbox](#)



 [Download \(Mac\)](#)

[Download \(Windows\)](#)

Compatible with Mac OS X 10.8+ and Windows 7+

<https://www.docker.com/toolbox>

Supported installation

Docker supports installation on the following:

- Amazon EC2 Installation
- Arch Linux
- Microsoft Azure platform
- Installation from binaries
- CentOS
- CRUX Linux
- Debian
- Fedora
- FrugalWare
- Gentoo
- Google Cloud Platform
- Install on Joyent Public Cloud
- Mac OS X
- Oracle Linux
- Rackspace Cloud
- Red Hat Enterprise Linux
- IBM SoftLayer
- openSUSE and SUSE Linux Enterprise
- Ubuntu
- Windows

<https://docs.docker.com/installation/>

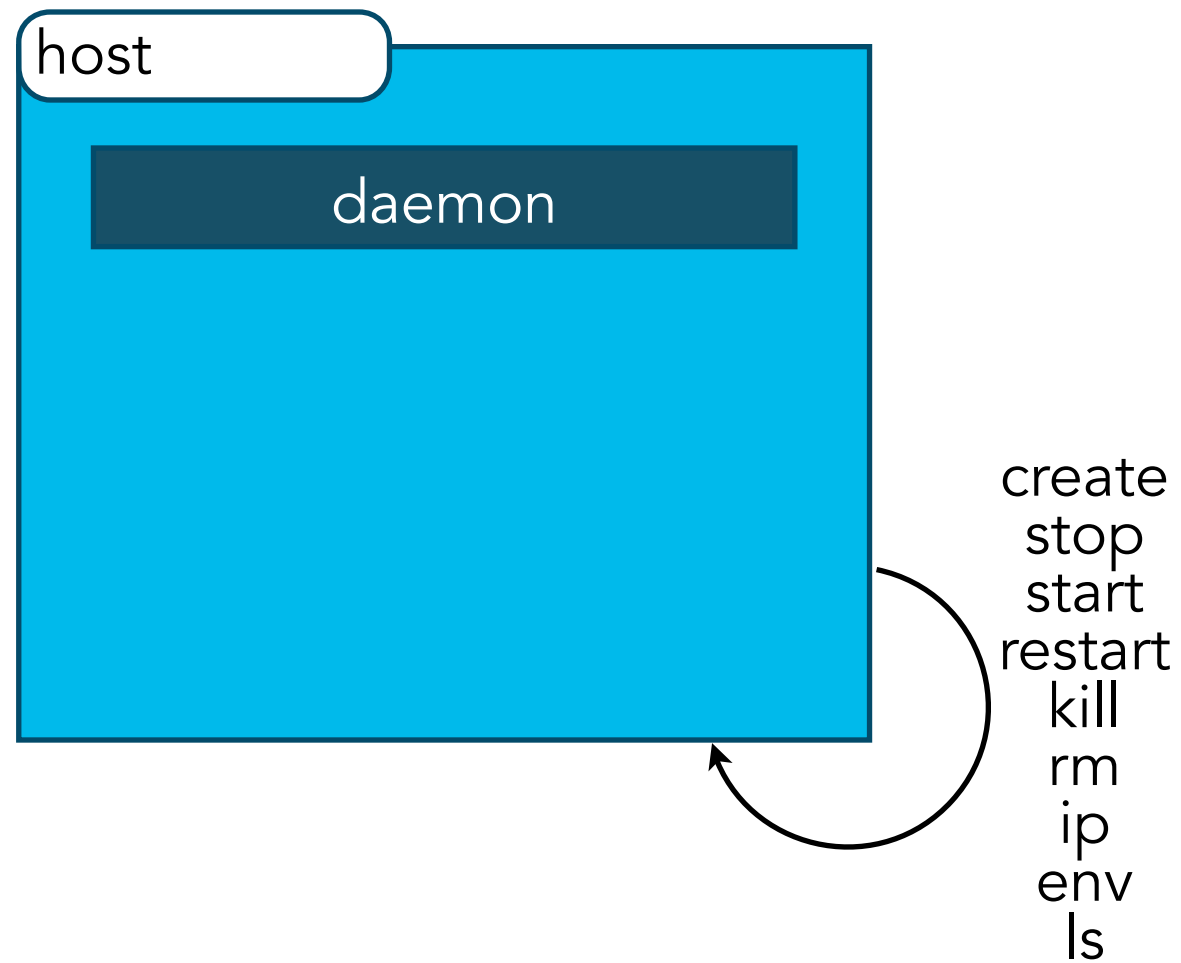
Lab I

I. Install Docker for your specific platform

CREATE MACHINE

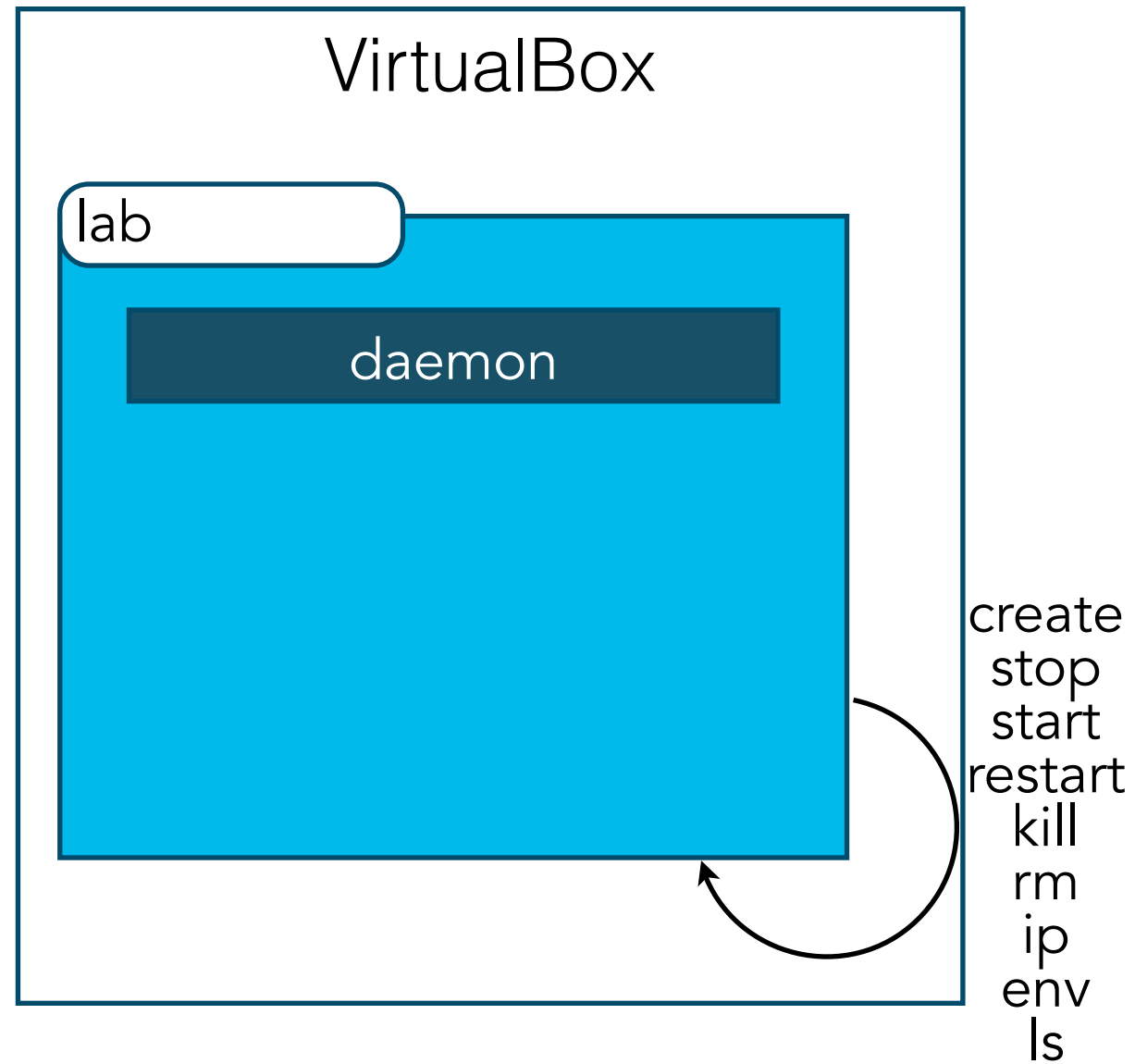


docker Lifecycle



```
docker-machine [OPTIONS] COMMAND [arg...]
```

docker Lifecycle



```
docker-machine [OPTIONS] COMMAND [arg...]
```

```
docker-machine create --driver=virtualbox lab
```

```
Creating VirtualBox VM...
```

```
Creating SSH key...
```

```
Starting VirtualBox VM...
```

```
Starting VM...
```

```
To see how to connect Docker to this machine, run: docker-machine env lab
```

docker-machine ls

NAME	ACTIVE	DRIVER	STATE	URL	SWARM
default		virtualbox	Stopped		
dev					
client1		virtualbox	Running	tcp://192.168.99.100:2376	
client2					
lab		virtualbox	Running		

`docker-machine env lab`

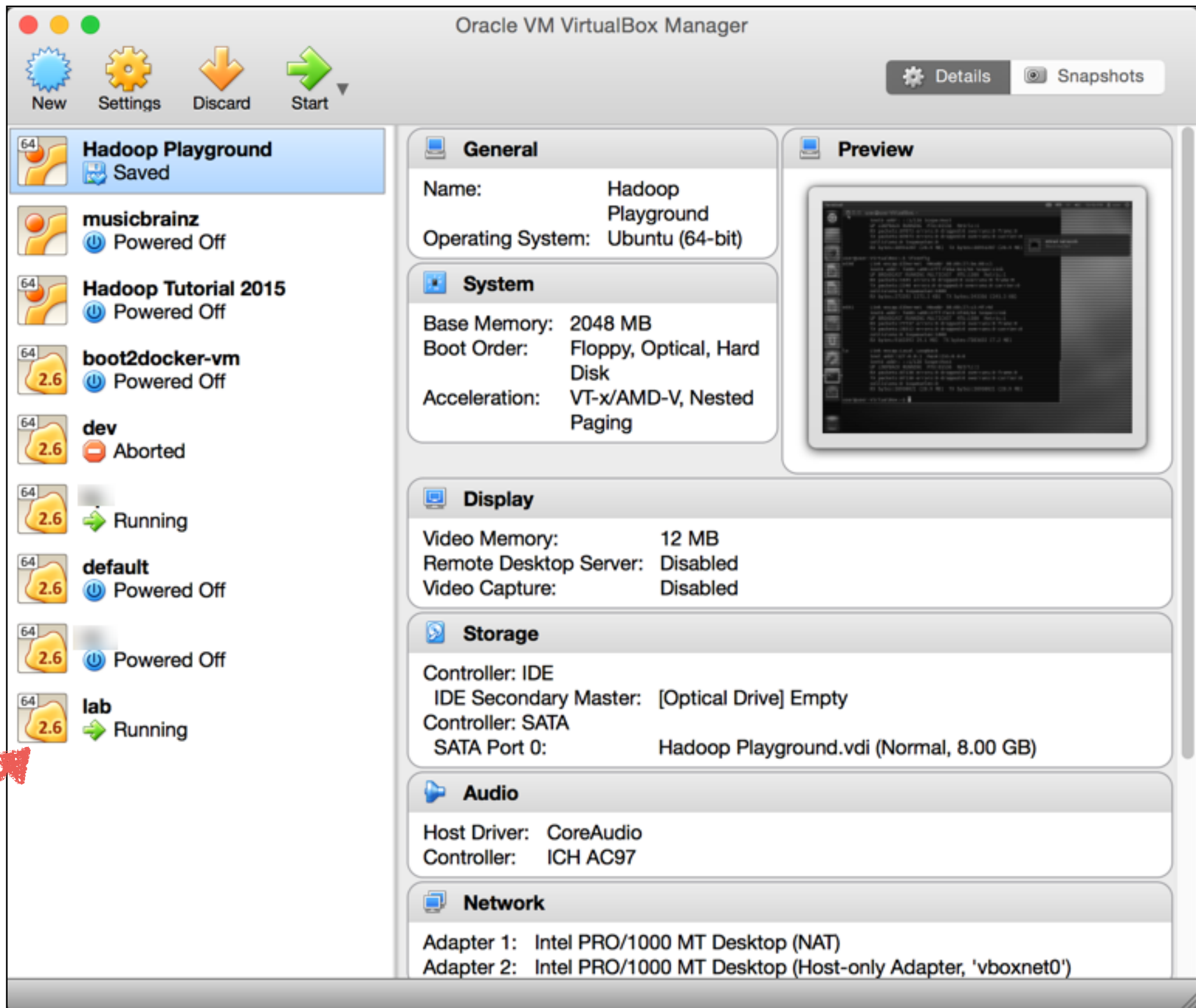
```
export DOCKER_TLS_VERIFY="1"
export DOCKER_HOST="tcp://192.168.99.102:2376"
export DOCKER_CERT_PATH="/Users/user
export DOCKER_MACHINE_NAME="lab"
# Run this command to configure your shell:
# eval "$$(docker-machine env lab)"
```



`eval "$$(docker-machine env lab)"`
`docker-machine ls`

NAME	ACTIVE	DRIVER	STATE	URL	SWARM
default		virtualbox	Stopped		
dev		virtualbox	Running	tcp://192.168.99.101:2376	
client1					
client2		virtualbox	Stopped		
lab	*	virtualbox	Running		



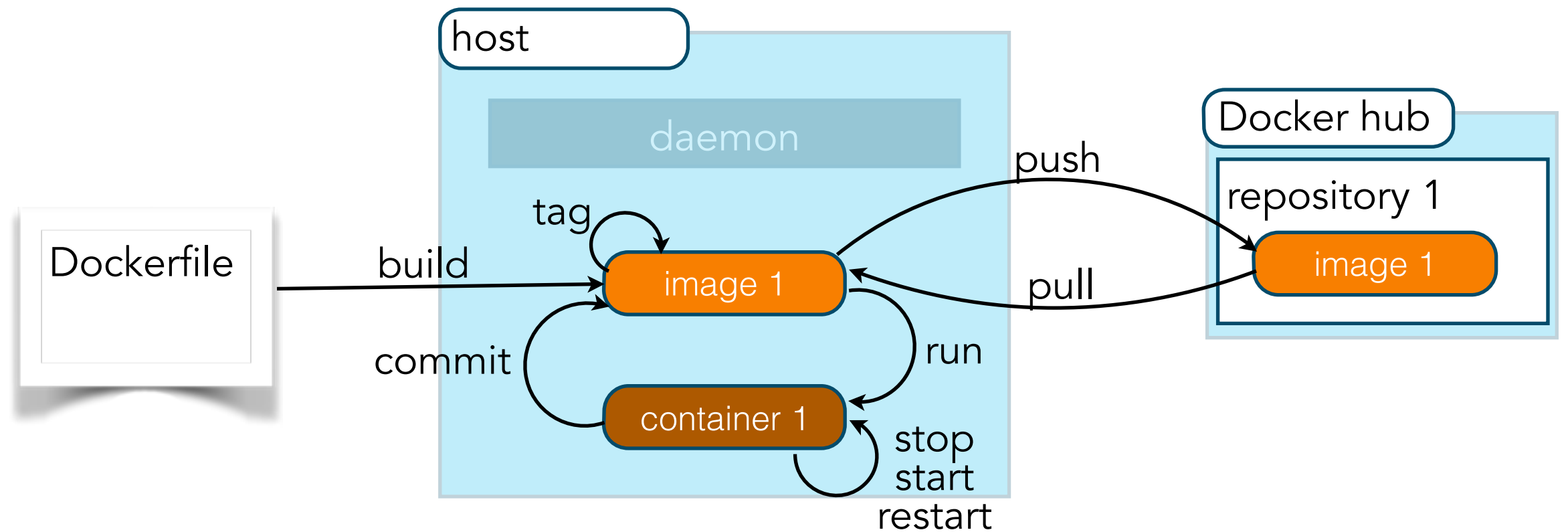


Lab 2

- I. Create a new Docker Machine called lab

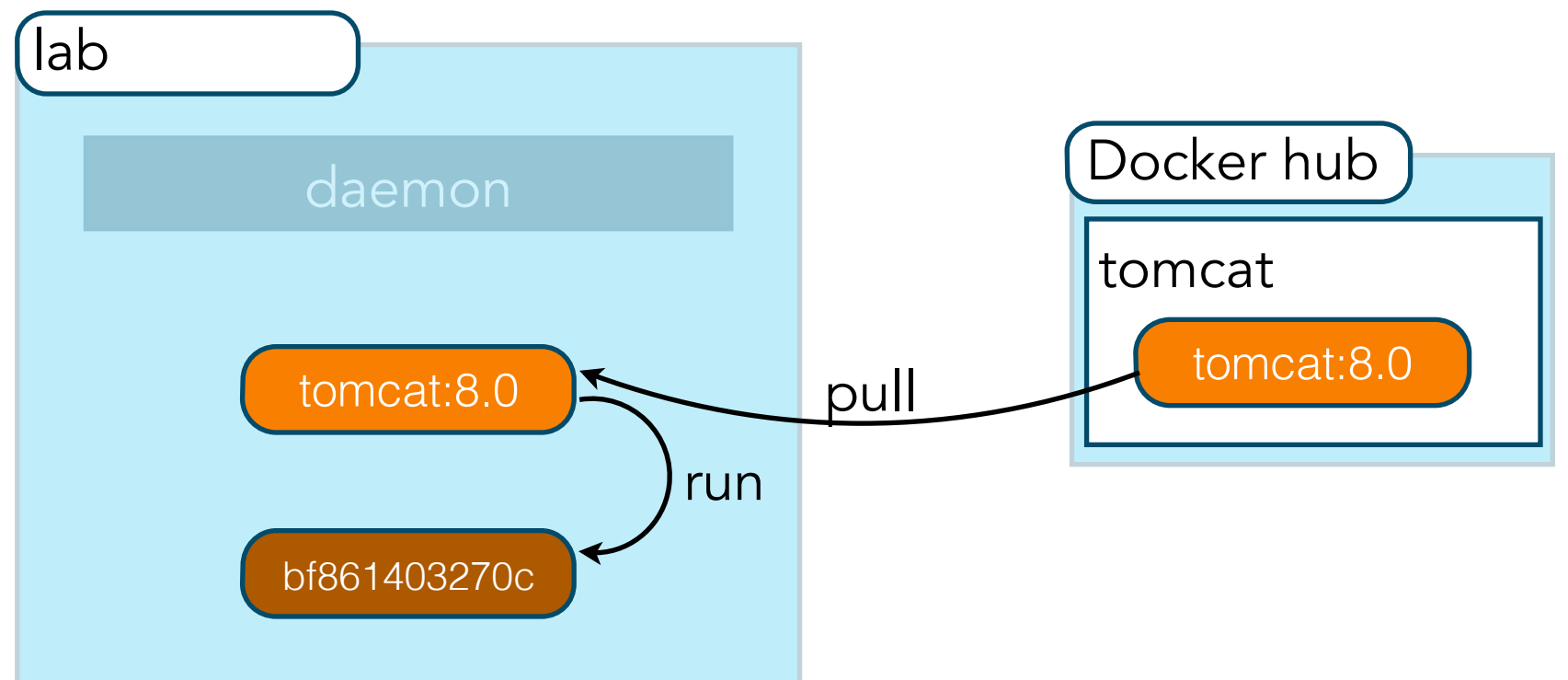
DOCKER LIFECYCLE

docker Lifecycle



`docker [OPTIONS] COMMAND [arg...]`

FINDING & RUNNING CONTAINERS





Repositories (1181)

All



tomcat
official

297
STARS

2.0 M
PULLS

>
DETAILS



cloudesire/tomcat
public | automated build

3
STARS

2.1 K
PULLS

>
DETAILS



dordoka/tomcat
public | automated build

8
STARS

6.1 K
PULLS

>
DETAILS



inspectit/tomcat
public | automated build

0
STARS

236
PULLS

>
DETAILS



cirit/tomcat
public | automated build

1
STARS

341
PULLS

>
DETAILS



andreptb/tomcat
public | automated build

1
STARS

244
PULLS

>
DETAILS

<https://hub.docker.com/>



OFFICIAL REPOSITORY

tomcat

Last pushed: 4 days ago

[Repo Info](#)[Tags](#)

Short Description

Apache Tomcat is an open source implementation of the Java Servlet and JavaServer Pages technologies

Docker Pull Command



```
docker pull tomcat
```

Full Description

Supported tags and respective Dockerfile links

- [6.0.44-jre7](#) , [6.0-jre7](#) , [6-jre7](#) , [6.0.44](#) , [6.0](#) , [6](#) ([6-jre7/Dockerfile](#))
- [6.0.44-jre8](#) , [6.0-jre8](#) , [6-jre8](#) ([6-jre8/Dockerfile](#))
- [7.0.64-jre7](#) , [7.0-jre7](#) , [7-jre7](#) , [7.0.64](#) , [7.0](#) , [7](#) ([7-jre7/Dockerfile](#))
- [7.0.64-jre8](#) , [7.0-jre8](#) , [7-jre8](#) ([7-jre8/Dockerfile](#))
- [8.0.28-jre7](#) , [8.0-jre7](#) , [8-jre7](#) , [jre7](#) , [8.0.28](#) , [8.0](#) , [8](#) , [latest](#) ([8-jre7/Dockerfile](#))
- [8.0.28-jre8](#) , [8.0-jre8](#) , [8-jre8](#) , [jre8](#) ([8-jre8/Dockerfile](#))

For more information about this image and its history, please see [the relevant manifest file](#)

([3 images for tomcat](#)). This image is updated via pull requests to [the docker](#)

How to use this image.

Run the default Tomcat server (CMD ["catalina.sh", "run"]):

```
$ docker run -it --rm tomcat:8.0
```

You can test it by visiting `http://container-ip:8080` in a browser or, if you need access outside the host, on port 8888:

```
$ docker run -it --rm -p 8888:8080 tomcat:8.0
```

You can then go to `http://localhost:8888` or `http://host-ip:8888` in a browser.

The default Tomcat environment in the image for versions 7 and 8 is:

```
CATALINA_BASE:  /usr/local/tomcat
CATALINA_HOME:  /usr/local/tomcat
CATALINA_TMPDIR: /usr/local/tomcat/temp
JRE_HOME:       /usr
CLASSPATH:      /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bi
```

The default Tomcat environment in the image for version 6 is:

```
CATALINA_BASE:  /usr/local/tomcat
CATALINA_HOME:  /usr/local/tomcat
CATALINA_TMPDIR: /usr/local/tomcat/temp
JRE_HOME:       /usr
CLASSPATH:      /usr/local/tomcat/bin/bootstrap.jar
```

The configuration files are available in `/usr/local/tomcat/conf/` . By default, no user is included in the "manager-gui" role required to operate the `/manager/html` web application. If you wish to use this app, you must define such a user in `tomcat-users.xml` .

License

View [license information](#) for the software contained in this image.

This image is officially supported on Docker version 1.8.3.

Support for older versions (down to 1.6) is provided on a best-effort basis.

Please see [the Docker installation documentation](#) for details on how to upgrade your Docker daemon.

User Feedback

Documentation

Documentation for this image is stored in the `tomcat/` [directory](#) of the `docker-library/docs` [GitHub repo](#). Be sure to familiarize yourself with the [repository's README.md file](#) before attempting a pull request.

Issues

If you have any problems with or questions about this image, please contact us through a [GitHub issue](#).

You can also reach many of the official image maintainers via the `#docker-library` IRC channel on [Freenode](#).

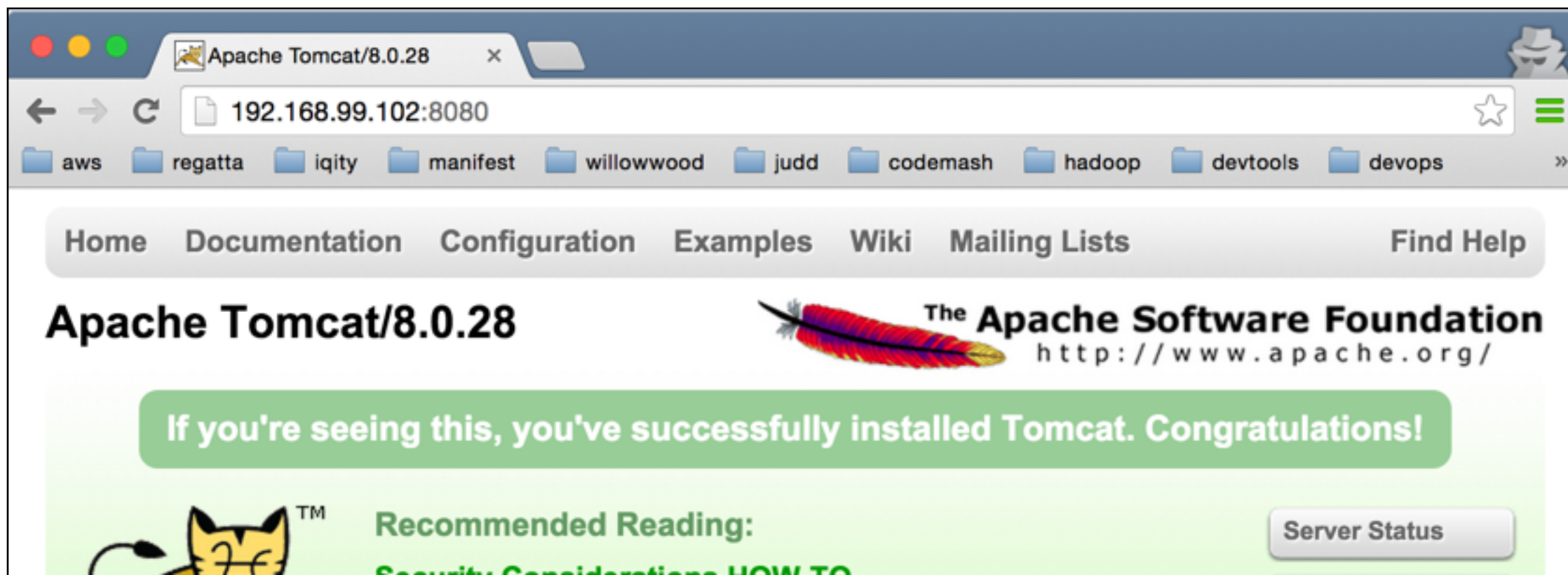
Contributing

You are invited to contribute new features, fixes, or updates, large or small; we are always thrilled to receive pull requests, and do our best to process them as fast as we can.

Before you start to code, we recommend discussing your plans through a [GitHub issue](#), especially for more ambitious contributions. This gives other contributors a chance to point you in the right direction, give you feedback on your design, and help you find out if someone else is working on the same thing.

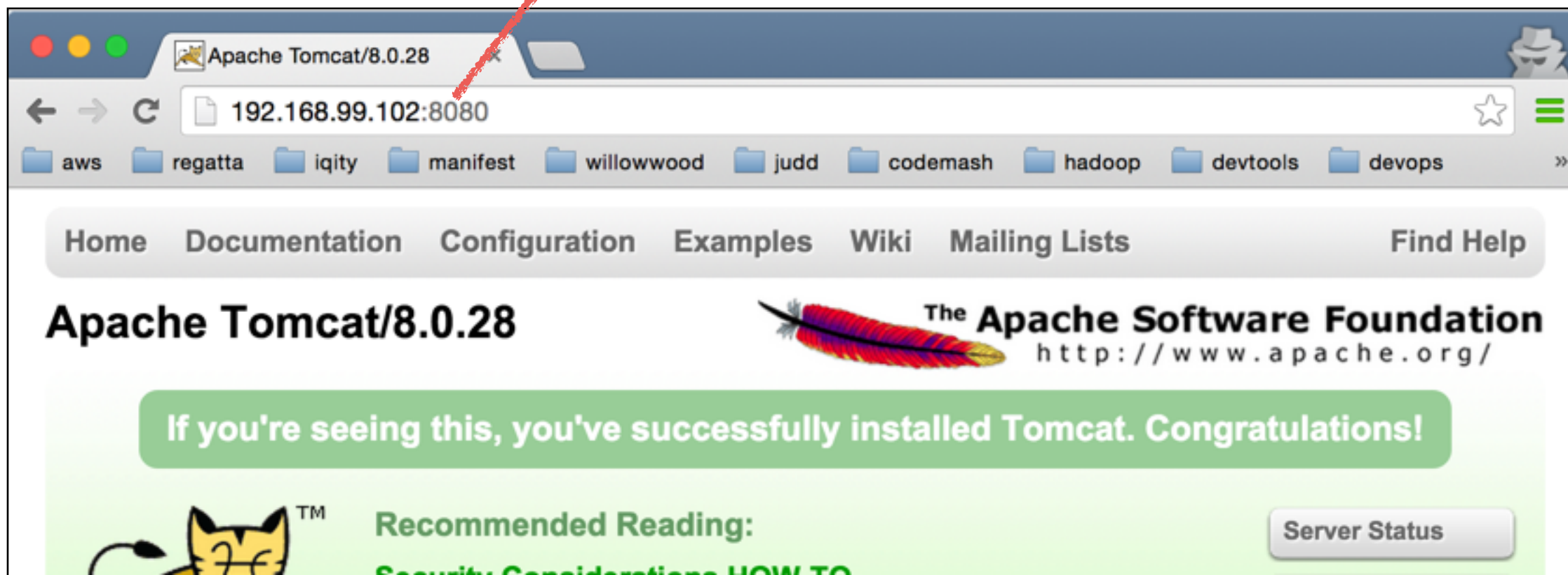
```
docker run -it --rm -p 8080:8080 tomcat:8.0
```

```
Using CATALINA_BASE: /usr/local/tomcat
Using CATALINA_HOME: /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME: /usr
Using CLASSPATH: /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar
18-Oct-2015 23:34:47.365 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server version: Apache Tomcat/8.0.28
18-Oct-2015 23:34:47.366 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server built: Oct 7 2015 18:25:21 UTC
18-Oct-2015 23:34:47.367 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server number: 8.0.28.0
18-Oct-2015 23:34:47.367 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Name: Linux
18-Oct-2015 23:34:47.368 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Version: 4.0.9-boot2docker
18-Oct-2015 23:34:47.368 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Architecture: amd64
18-Oct-2015 23:34:47.369 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Java Home: /usr/lib/jvm/java-7-openjdk-amd64/jre
18-Oct-2015 23:34:47.369 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Version: 1.7.0_79-b14
18-Oct-2015 23:34:47.370 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Vendor: Oracle Corporation
18-Oct-2015 23:34:47.370 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_BASE: /usr/local/tomcat
18-Oct-2015 23:34:47.371 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_HOME: /usr/local/tomcat
18-Oct-2015 23:34:47.371 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.util.logging.config.file=/usr/local/tomcat/conf/logging.properties
18-Oct-2015 23:34:47.372 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -
Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager
18-Oct-2015 23:34:47.372 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.endorsed.dirs=/usr/local/tomcat/endorsed
18-Oct-2015 23:34:47.372 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Dcatalina.base=/usr/local/tomcat
18-Oct-2015 23:34:47.373 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Dcatalina.home=/usr/local/tomcat
18-Oct-2015 23:34:47.373 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.io.tmpdir=/usr/local/tomcat/temp
18-Oct-2015 23:34:47.373 INFO [main] org.apache.catalina.core.AprLifecycleListener.lifecycleEvent The APR based Apache Tomcat Native library which allows optimal performance in production environments was not found on the java.library.path: /usr/java/packages/lib/amd64:/usr/lib/x86_64-linux-gnu/jni:/lib/x86_64-linux-gnu:/usr/lib/x86_64-linux-gnu:/usr/lib/jni:/lib:/usr/lib
18-Oct-2015 23:34:47.519 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["http-nio-8080"]
18-Oct-2015 23:34:47.546 INFO [main] org.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared selector for servlet write/read
18-Oct-2015 23:34:47.559 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["ajp-nio-8009"]
18-Oct-2015 23:34:47.561 INFO [main] org.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared selector for servlet write/read
18-Oct-2015 23:34:47.562 INFO [main] org.apache.catalina.startup.Catalina.load Initialization processed in 812 ms
18-Oct-2015 23:34:47.607 INFO [main] org.apache.catalina.core.StandardService.startInternal Starting service Catalina
18-Oct-2015 23:34:47.618 INFO [main] org.apache.catalina.core.StandardEngine.startInternal Starting Servlet Engine: Apache Tomcat/8.0.28
18-Oct-2015 23:34:47.647 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory /usr/local/tomcat/webapps/ROOT
```



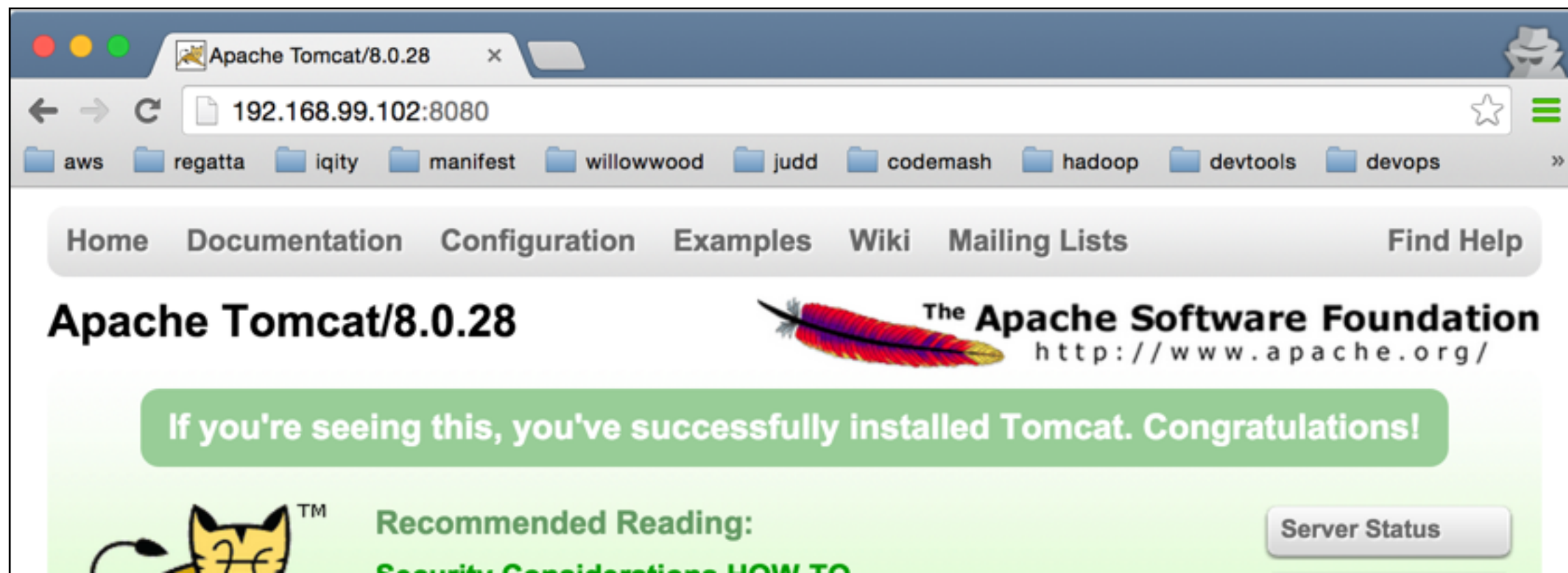
`docker run -it --rm -p 8080:8080 tomcat:8.0`

```
Using CATALINA_BASE: /usr/local/tomcat
Using CATALINA_HOME: /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME: /usr
Using CLASSPATH: /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar
18-Oct-2015 23:34:47.365 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server version: Apache Tomcat/8.0.28
18-Oct-2015 23:34:47.366 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server built: Oct 2015 18:25:21 UTC
18-Oct-2015 23:34:47.367 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server number: 8.0.28.0
18-Oct-2015 23:34:47.367 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Name: Linux
18-Oct-2015 23:34:47.368 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Version: 4.0.6-1-boot2docker
18-Oct-2015 23:34:47.368 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Architecture: amd64
18-Oct-2015 23:34:47.369 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Java Home: /usr/lib/jvm/java-7-openjdk-amd64/jre
18-Oct-2015 23:34:47.369 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Version: 1.7.0_79-b14
18-Oct-2015 23:34:47.370 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Vendor: Oracle Corporation
18-Oct-2015 23:34:47.370 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_BASE: /usr/local/tomcat
18-Oct-2015 23:34:47.371 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_HOME: /usr/local/tomcat
18-Oct-2015 23:34:47.371 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.util.logging.config.file=/usr/local/tomcat/conf/logging.properties
18-Oct-2015 23:34:47.372 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -
Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager
18-Oct-2015 23:34:47.372 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.endorsed.dirs=/usr/local/tomcat/endorsed
18-Oct-2015 23:34:47.372 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Dcatalina.base=/usr/local/tomcat
18-Oct-2015 23:34:47.373 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Dcatalina.home=/usr/local/tomcat
18-Oct-2015 23:34:47.373 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.io.tmpdir=/usr/local/tomcat/temp
18-Oct-2015 23:34:47.373 INFO [main] org.apache.catalina.core.AprLifecycleListener.lifecycleEvent The APR based Apache Tomcat Native library which allows optimal performance in production environments was not found on the java.library.path: /usr/java/packages/lib/amd64:/usr/lib/x86_64-linux-gnu/jni:/lib/x86_64-linux-gnu:/usr/lib/x86_64-linux-gnu:/usr/lib/jni:/lib:/usr/lib
18-Oct-2015 23:34:47.519 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["http-nio-8080"]
18-Oct-2015 23:34:47.546 INFO [main] org.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared selector for servlet write/read
18-Oct-2015 23:34:47.559 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["ajp-nio-8009"]
18-Oct-2015 23:34:47.561 INFO [main] org.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared selector for servlet write/read
18-Oct-2015 23:34:47.562 INFO [main] org.apache.catalina.startup.Catalina.load Initialization processed in 812 ms
18-Oct-2015 23:34:47.607 INFO [main] org.apache.catalina.core.StandardService.startInternal Starting service Catalina
18-Oct-2015 23:34:47.618 INFO [main] org.apache.catalina.core.StandardEngine.startInternal Starting Servlet Engine: Apache Tomcat/8.0.28
18-Oct-2015 23:34:47.647 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory /usr/local/tomcat/webapps/ROOT
```



docker-machine ip lab

192.168.99.102

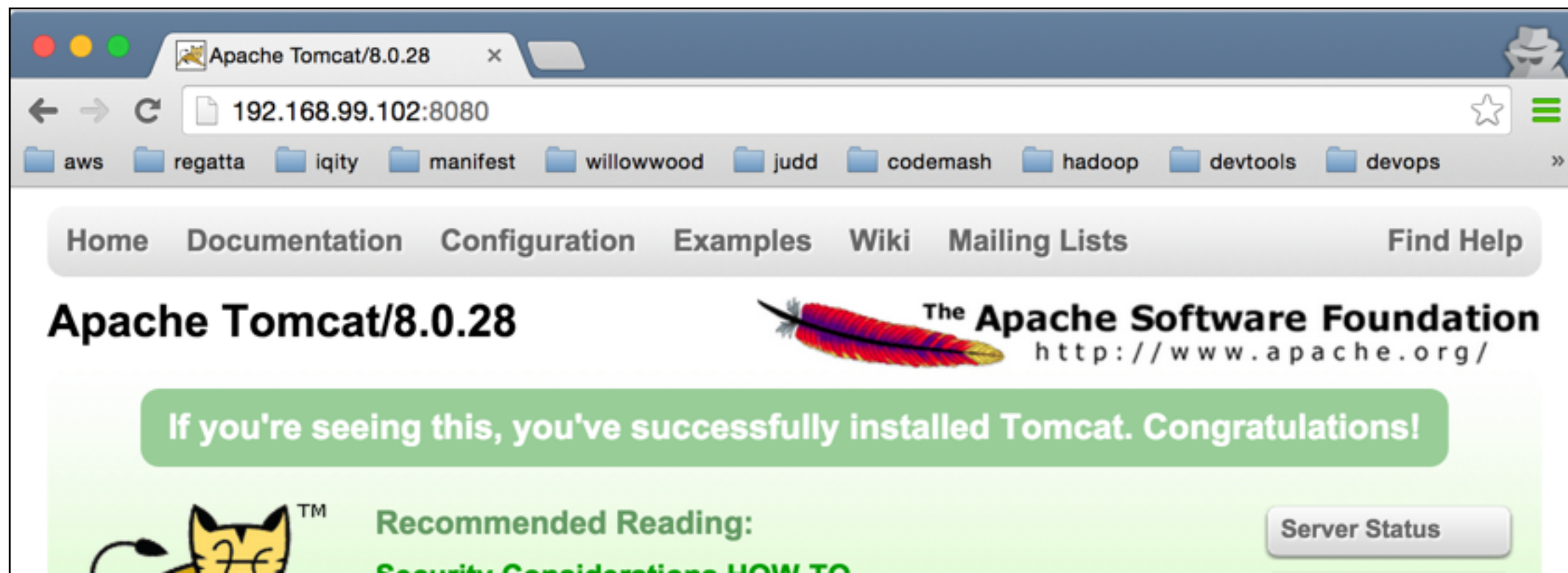


```
docker-machine ip lab
```

```
192.168.99.102
```

```
docker-machine ls
```

NAME	ACTIVE	DRIVER	STATE	URL	SWARM
default		virtualbox	Stopped		
dev		virtualbox	Running	tcp://192.168.99.101:2376	
client1		virtualbox	Running	tcp://192.168.99.100:2376	
client2		virtualbox	Stopped		
lab		virtualbox	Running	tcp://192.168.99.102:2376	

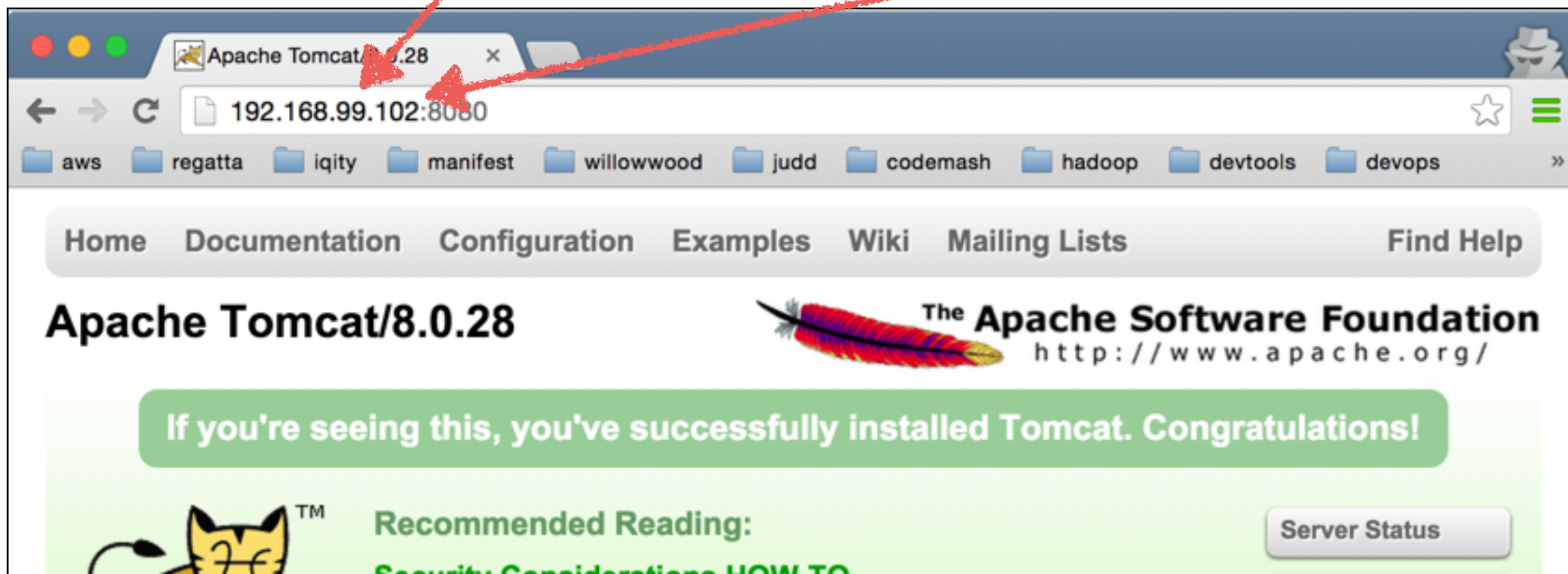


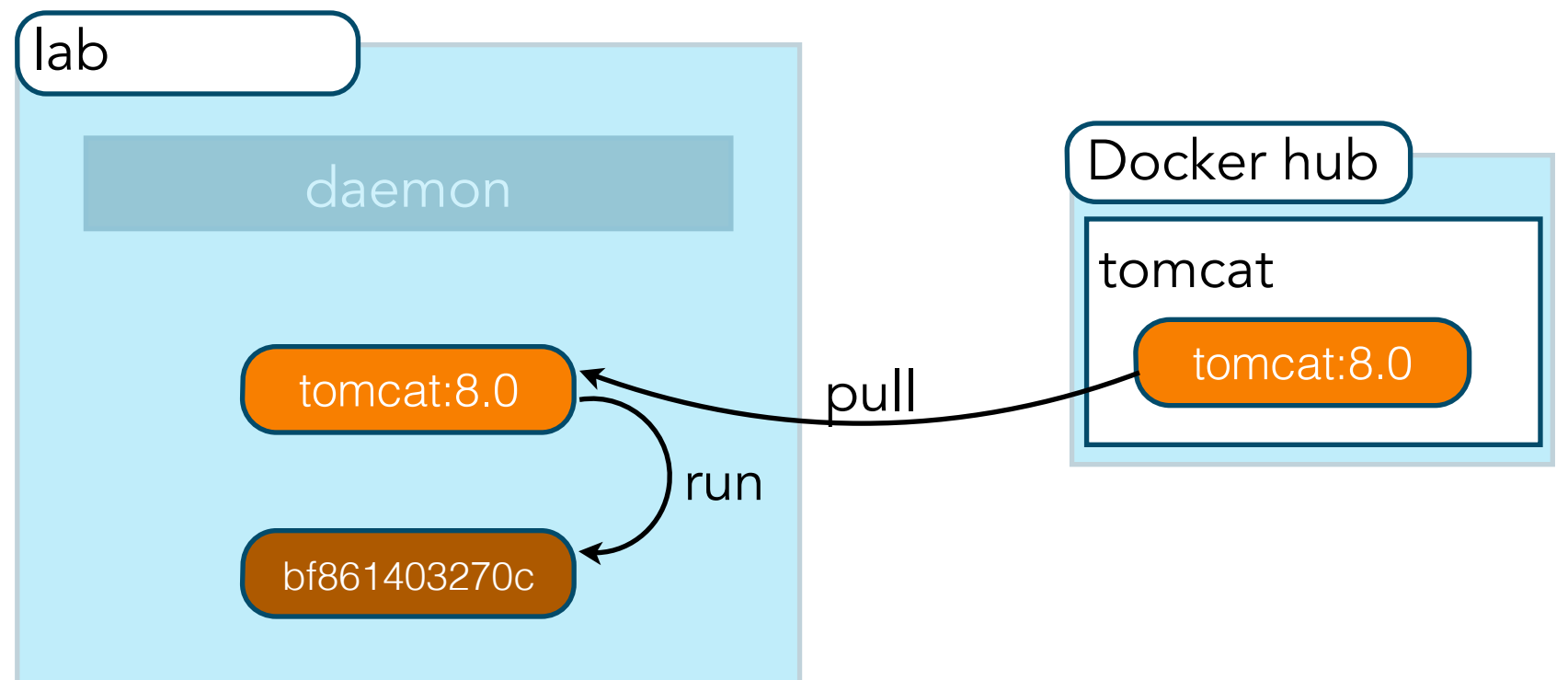
```
docker-machine ip lab
```

```
192.168.99.102
```

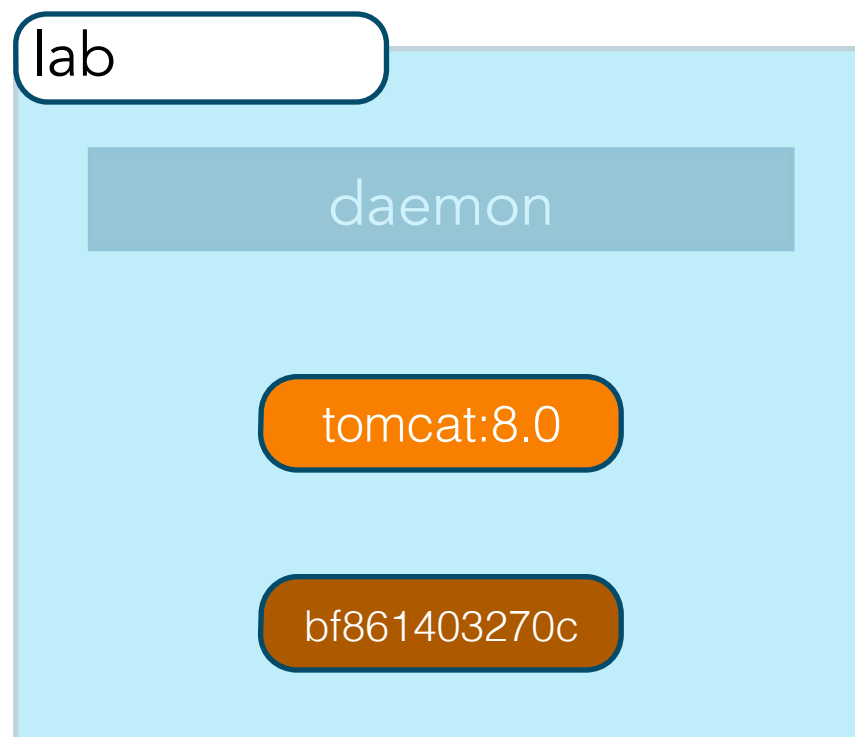
```
docker-machine ls
```

NAME	ACTIVE	DRIVER	STATE	URL	SWARM
default		virtualbox	Stopped		
dev		virtualbox	Running	tcp://192.168.99.101:2376	
client1		virtualbox	Running	tcp://192.168.99.100:2376	
client2		virtualbox	Stopped		
lab		virtualbox	Running	tcp://192.168.99.102:2376	



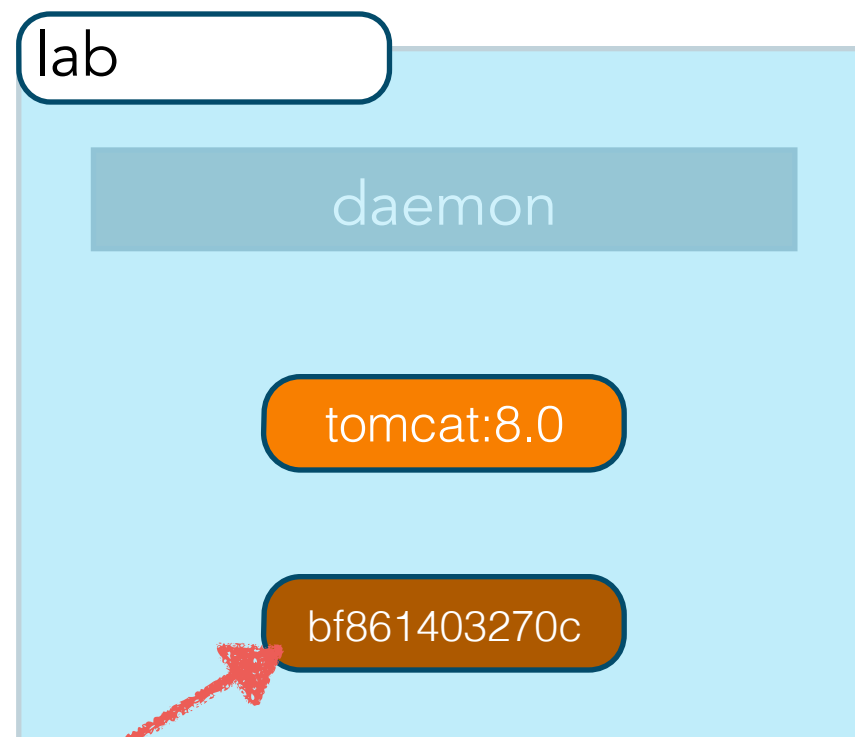


```
docker run -it --rm -p 8080:8080 tomcat:8.0
```



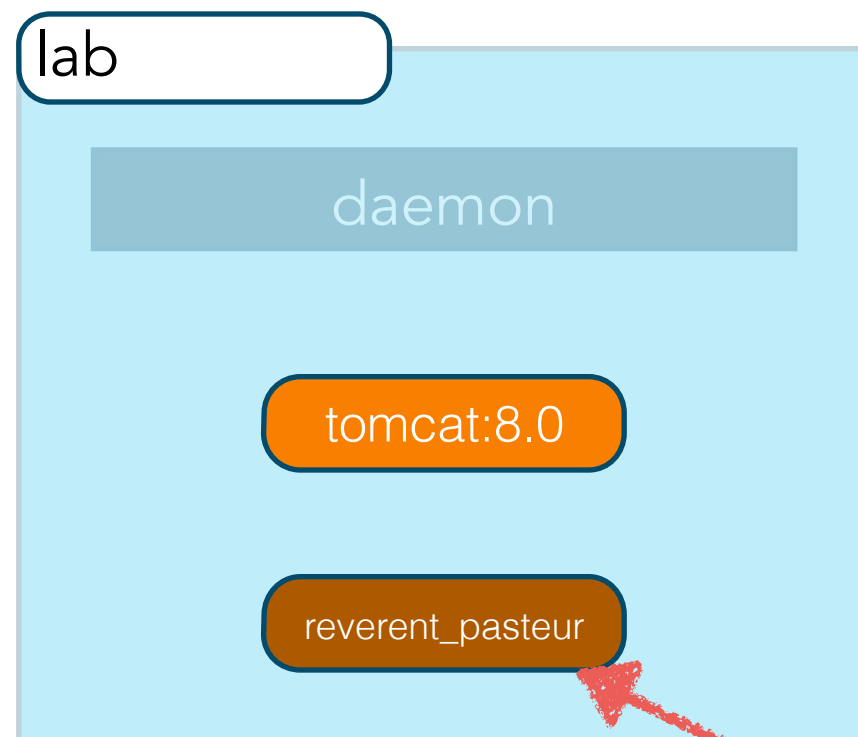
docker ps

CONTAINER ID	IMAGE	COMMAND	STATUS	PORTS	NAMES
bf861403270c	tomcat:8.0	"catalina.sh run"	Up 2 seconds	0.0.0.0:8080->8080/tcp	reverent_pasteur



docker ps

CONTAINER ID	IMAGE	COMMAND	STATUS	PORTS	NAMES
bf861403270c	tomcat:8.0	"catalina.sh run"	Up 2 seconds	0.0.0.0:8080->8080/tcp	reverent_pasteur



docker ps

CONTAINER ID	IMAGE	COMMAND	STATUS	PORTS	NAMES
bf861403270c	tomcat:8.0	"catalina.sh run"	Up 2 seconds	0.0.0.0:8080->8080/tcp	reverent_pasteur

Lab 3

1. Run Tomcat 8 container
2. Validate you can see the Tomcat home page
3. Determine the name & id of your container
4. Shut the container down

Hint: it might want to use two command prompts to do this

CREATING IMAGES

- Run container and manually install software
- Dockerfile

- ~~Run container and manually install software~~
- Dockerfile

Dockerfile

```
FROM mysql:5.5.45
```

```
COPY docker-entrypoint-initdb.d /docker-entrypoint-initdb.d/
```

```
RUN echo America/New_York | tee /etc/timezone && dpkg-reconfigure --frontend noninteractive tzdata
```

```
RUN echo "lower_case_table_names = 1" >> /etc/mysql/my.cnf
```

```
ENTRYPOINT ["/entrypoint.sh"]
```

```
CMD ["mysqld"]
```

docker-entrypoint-initdb.d/01-create-database.sql

```
CREATE DATABASE neuz;
```

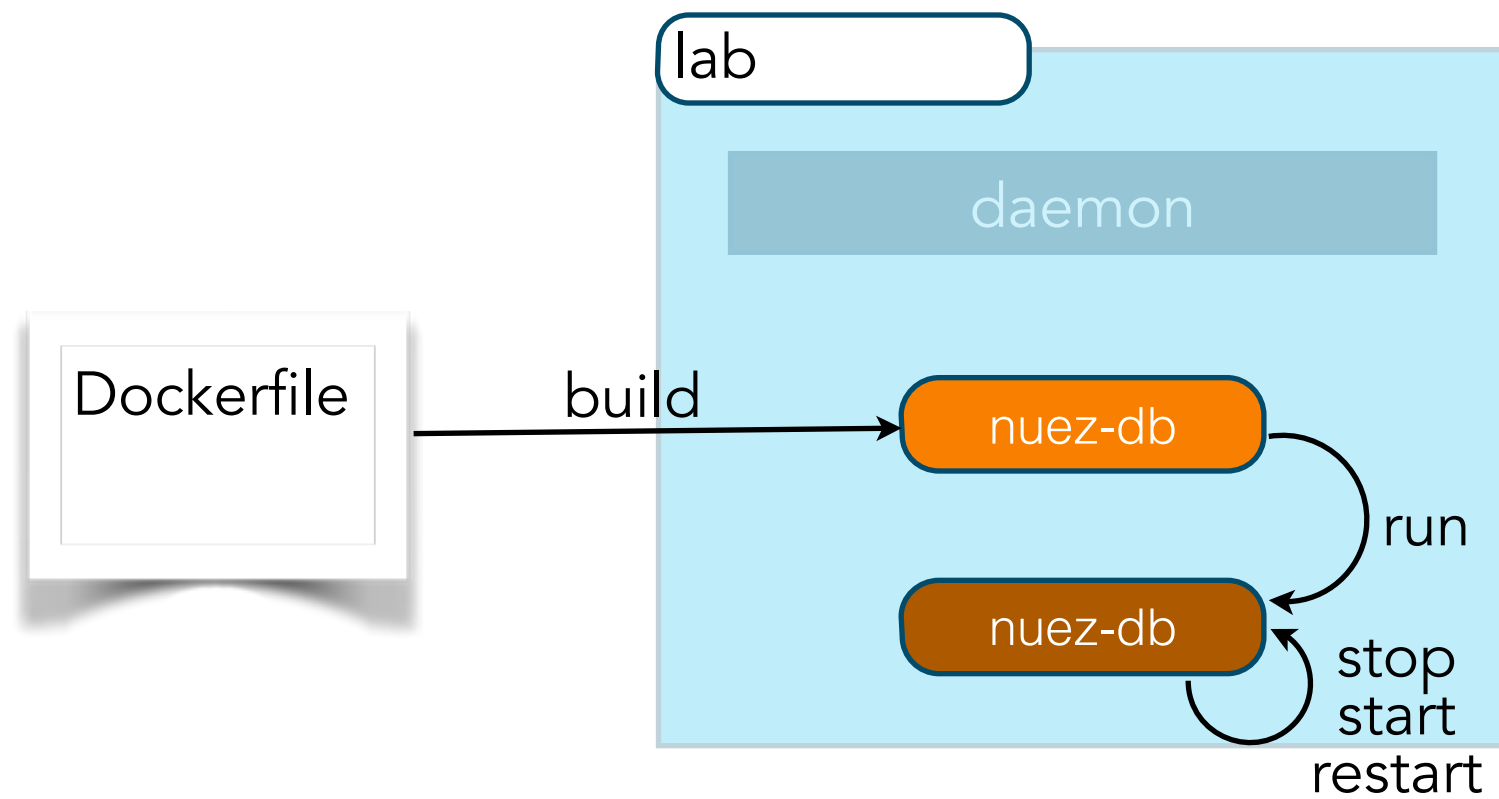
docker-entrypoint-initdb.d/02-create-users.sql

```
CREATE USER 'nuez'@'%' IDENTIFIED BY 'nuez+1';
```

```
GRANT ALL PRIVILEGES ON neuz.* TO 'nuez'@'%';
```

```
FLUSH privileges;
```

Instruction	Description
FROM	This is must be the first instruction in the Dockerfile and identifies the image to
MAINTAINER	Provides visibility and credit to the author of the image.
RUN	Executes a Linux command for configuring and installing.
ENTRYPOINT	The final script or application used bootstrap the container making it an
CMD	Provide default arguments to the ENTRYPOINT using a JSON array format.
LABEL	Name/value metadata about the image.
ENV	Sets environment variables.
COPY	Copies files into the container.
ADD	Alternative to copy.
WORKDIR	Sets working directory for RUN, CMD, ENTRYPOINT, COPY and/or ADD
EXPOSE	Ports the container will listen on.
VOLUME	Creates a mount point.
USER	User to run RUN, CMD and/or ENTRYPOIN instructions..



```
docker build -t nuez-db .
```

docker build -t nuev-db .

Sending build context to Docker daemon 35.33 kB

Step 0 : FROM mysql:5.5.45

5.5.45: Pulling from library/mysql

ba249489d0b6: Pull complete

19de96c112fc: Pull complete

2e32b26a94ed: Pull complete

637386aea7a0: Pull complete

f40aa7fe5d68: Pull complete

f9cc53679b1f: Pull complete

6b4bdd0d9a0c: Pull complete

066f73588b2c: Pull complete

8c7b0f689d1a: Pull complete

ea70677dc485: Pull complete

9d511b9dbf89: Pull complete

0896de2c270f: Pull complete

a29e02ef2642: Pull complete

bb322ed102e5: Pull complete

1c764f968123: Pull complete

2b884c0bb51d: Pull complete

0da0b10c6fd8: Pull complete

Digest: sha256:72a09a61824bdaf652e701fcbf0ee12f5b132d8fdeaf1629ce42960375de03cb

Status: Downloaded newer image for mysql:5.5.45

----> 0da0b10c6fd8

Step 1 : COPY docker-entrypoint-initdb.d /docker-entrypoint-initdb.d/

----> 46223bd058b1

Removing intermediate container bccb6f31fed3

Step 2 : RUN echo America/New_York | tee /etc/timezone && dpkg-reconfigure --frontend noninteractive tzdata

----> Running in cb82914c1be9

America/New_York

Current default time zone: 'America/New_York'

Local time is now: Sun Oct 18 20:53:55 EDT 2015.

Universal Time is now: Mon Oct 19 00:53:55 UTC 2015.

----> 928355f67c39

Removing intermediate container cb82914c1be9

Step 3 : RUN echo "lower_case_table_names = 1" >> /etc/mysql/my.cnf

----> Running in 6ccd82a7286c

----> 6d23b9680c54

Removing intermediate container 6ccd82a7286c

Step 4 : ENTRYPOINT /entrypoint.sh

----> Running in c4c00462c691

----> 62ad0d0bc850

Removing intermediate container c4c00462c691

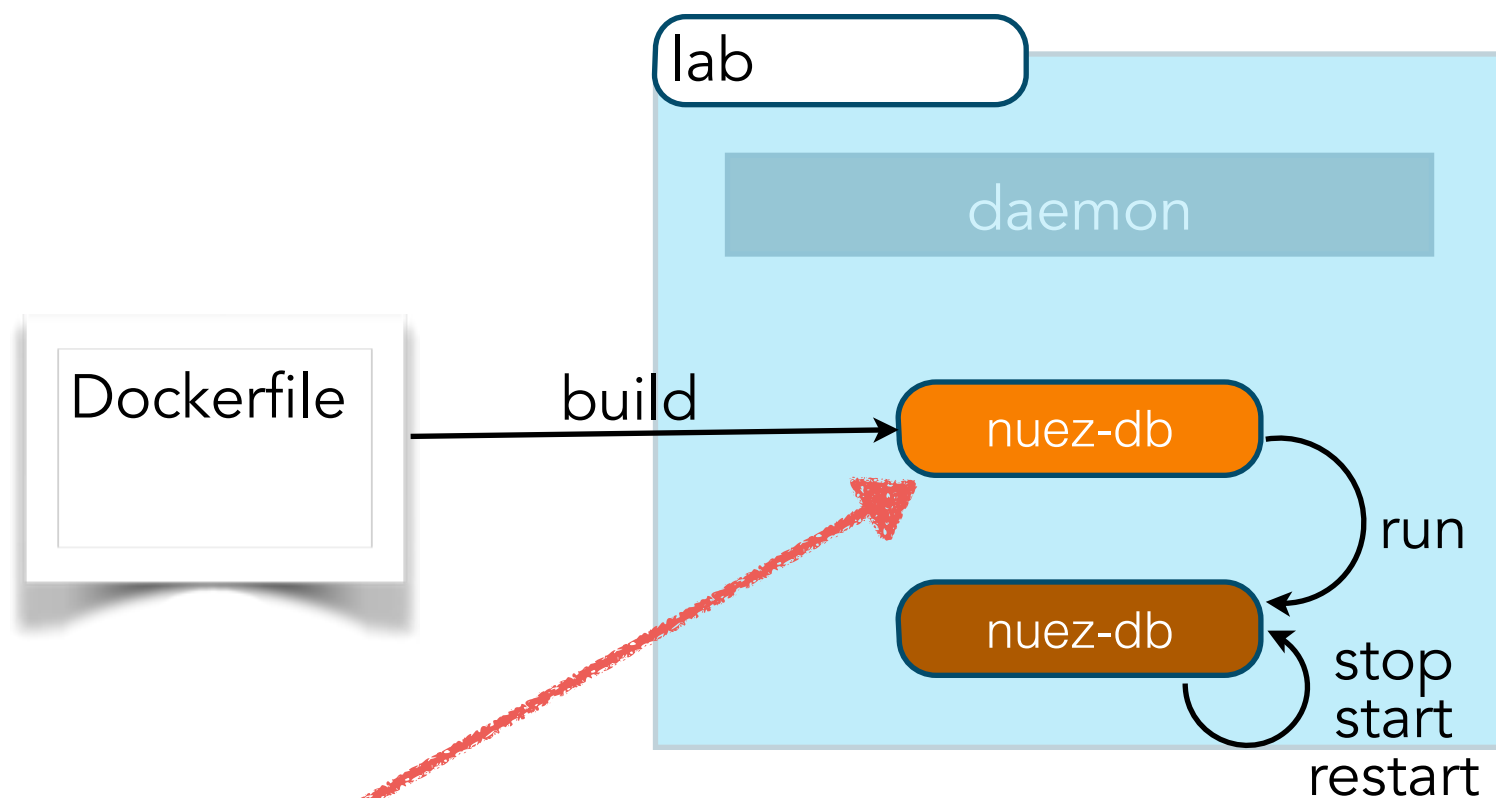
Step 5 : CMD mysqld

----> Running in 1cd40a7e38f8

----> 16f80cf33da2

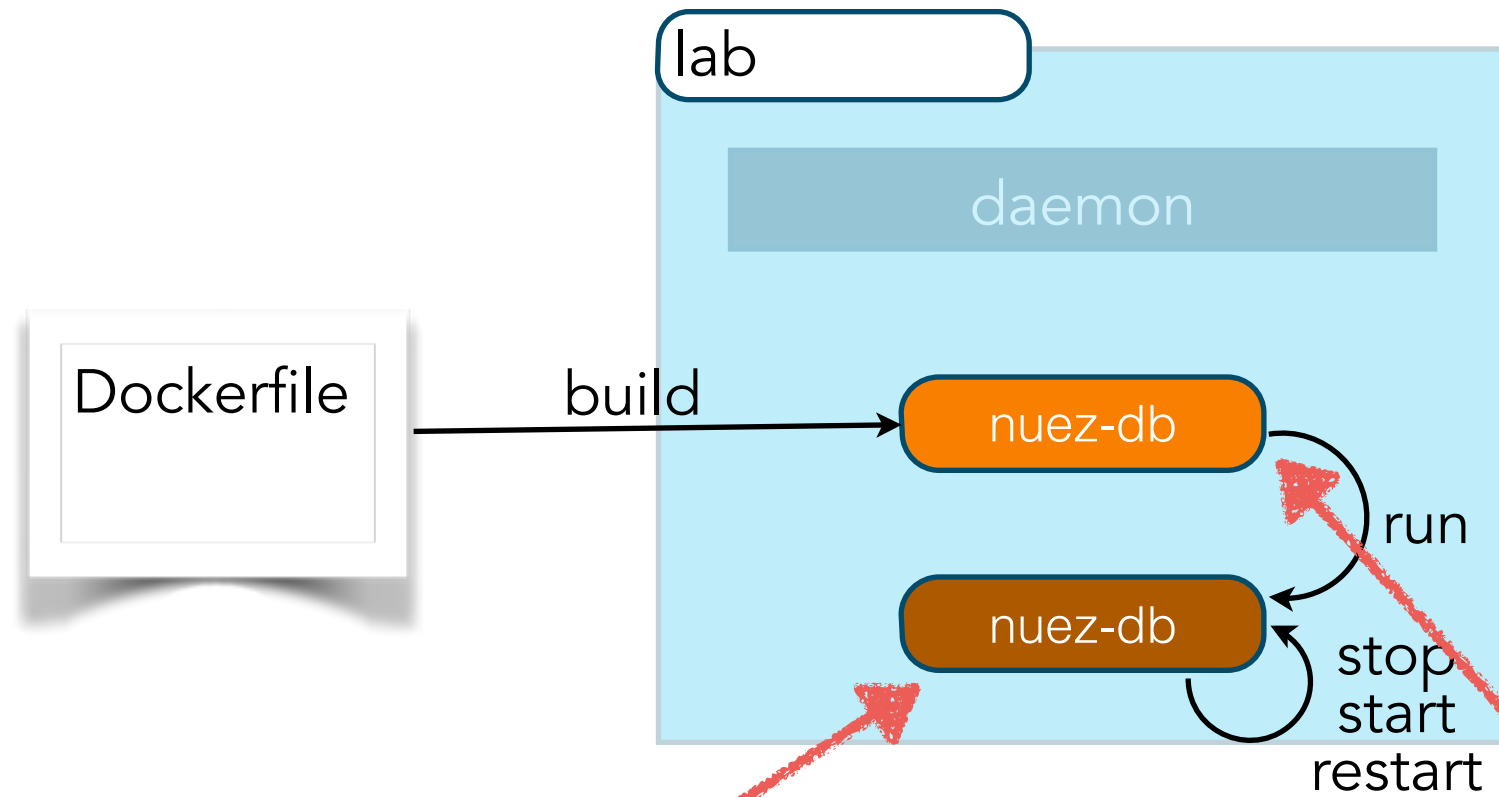
Removing intermediate container 1cd40a7e38f8

Successfully built 16f80cf33da2

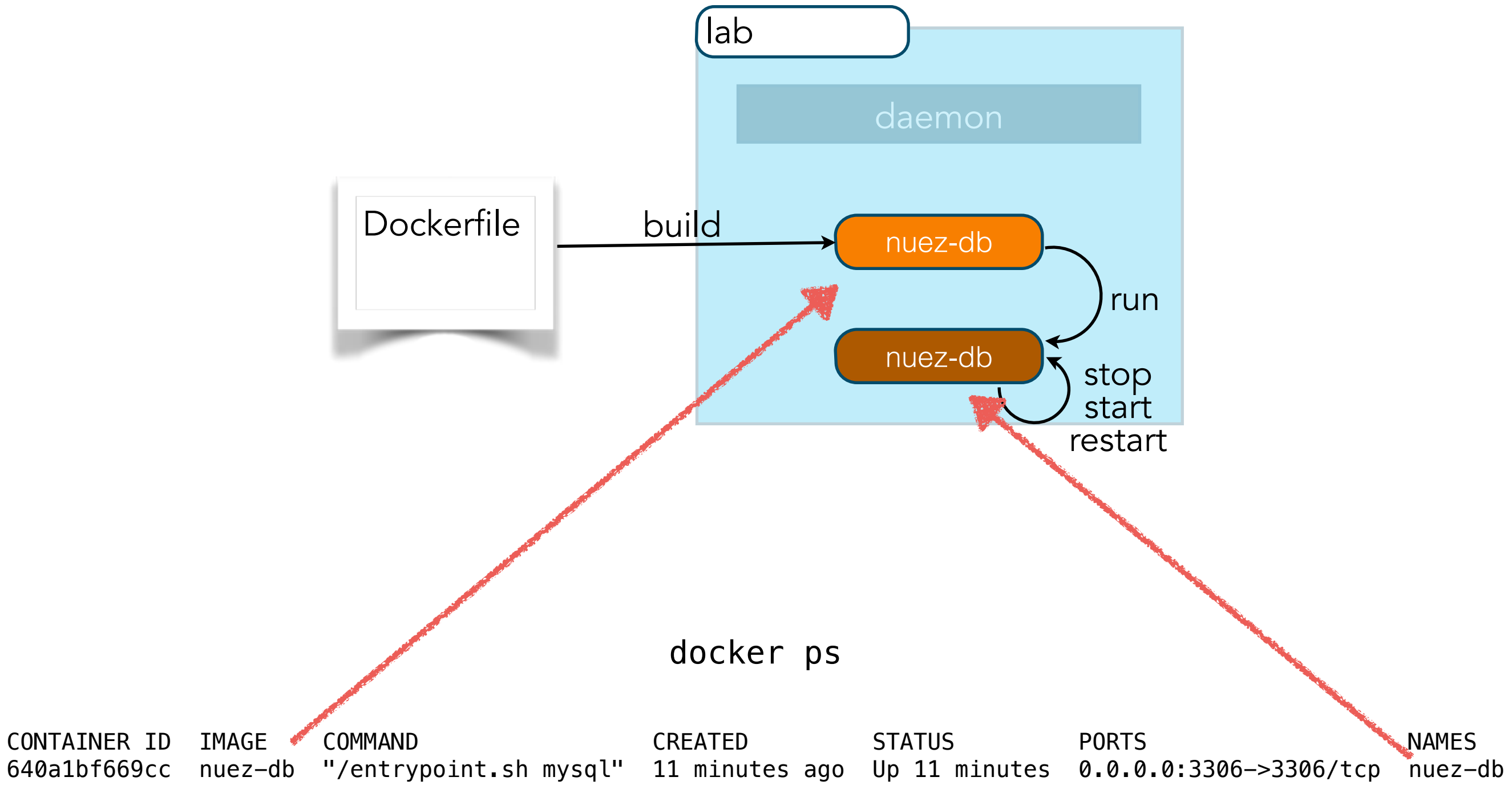


docker images

REPOSITORY SIZE	TAG	IMAGE ID	CREATED	VIRTUAL
nuez-db	latest	16f80cf33da2	8 minutes ago	214.4 MB
tomcat	8.0	c6838f52cb84	4 days ago	347.7 MB
mysql	5.5.45	0da0b10c6fd8	5 weeks ago	213.5 MB



```
docker run --name nuez-db -e MYSQL_USER=nuez-app -e MYSQL_PASSWORD=nuez+1  
-e MYSQL_DATABASE=nuez -e MYSQL_ROOT_PASSWORD=root+1 -p 3306:3306 -d nuez-db  
640a1bf669cc97763d51bc28865a5115a103e953336f8a7e90d50d61a108d0ec
```



Lab 4

1. Create a nuev-db image with MySQL
2. Run nuev-db container as a daemon

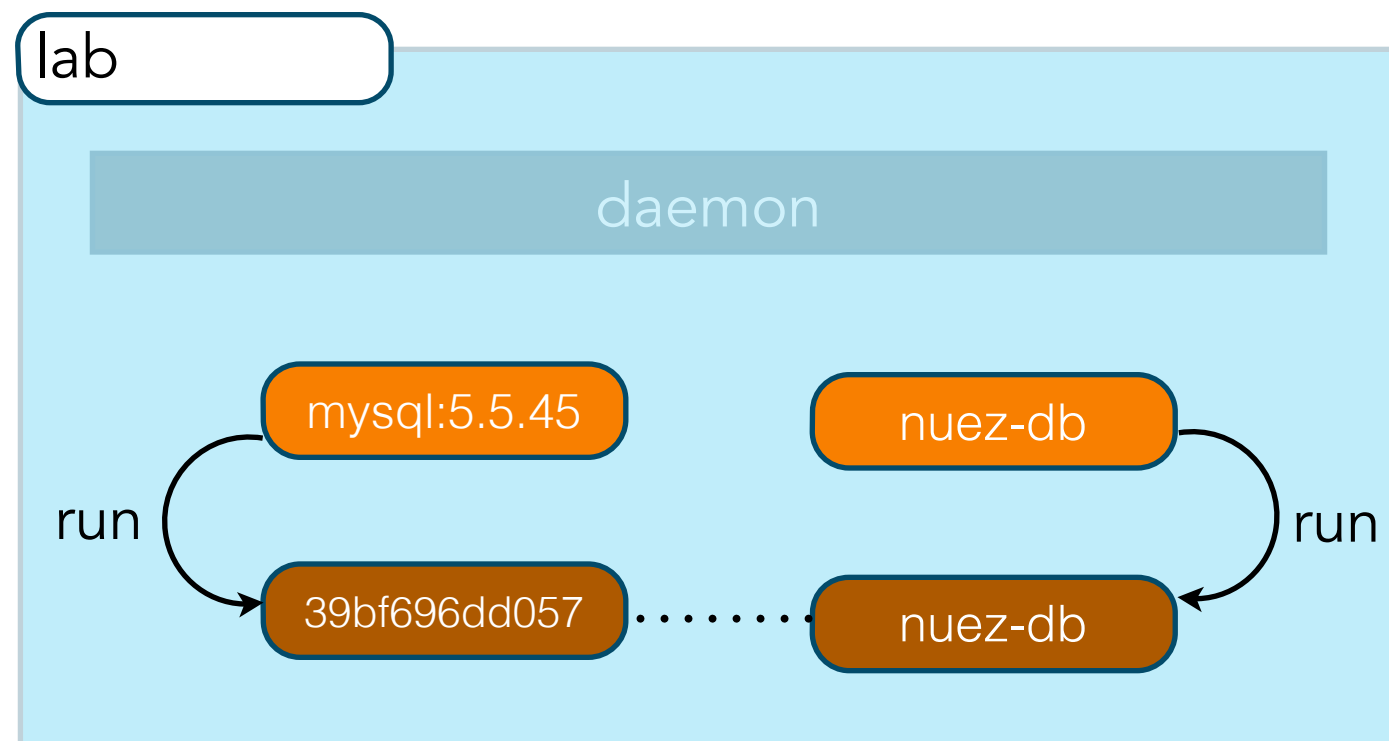
LINKING TO CONTAINER

```
mysql -h 192.168.99.102 -u root -proot+1 nuev
```

or

```
docker run -it --link nuev-db:mysql --rm mysql:5.5.45 bash
```

```
mysql -h $MYSQL_PORT_3306_TCP_ADDR -u $MYSQL_ENV_MYSQL_USER -p$MYSQL_ENV_MYSQL_PASSWORD $MYSQL_ENV_MYSQL_DATABASE
```



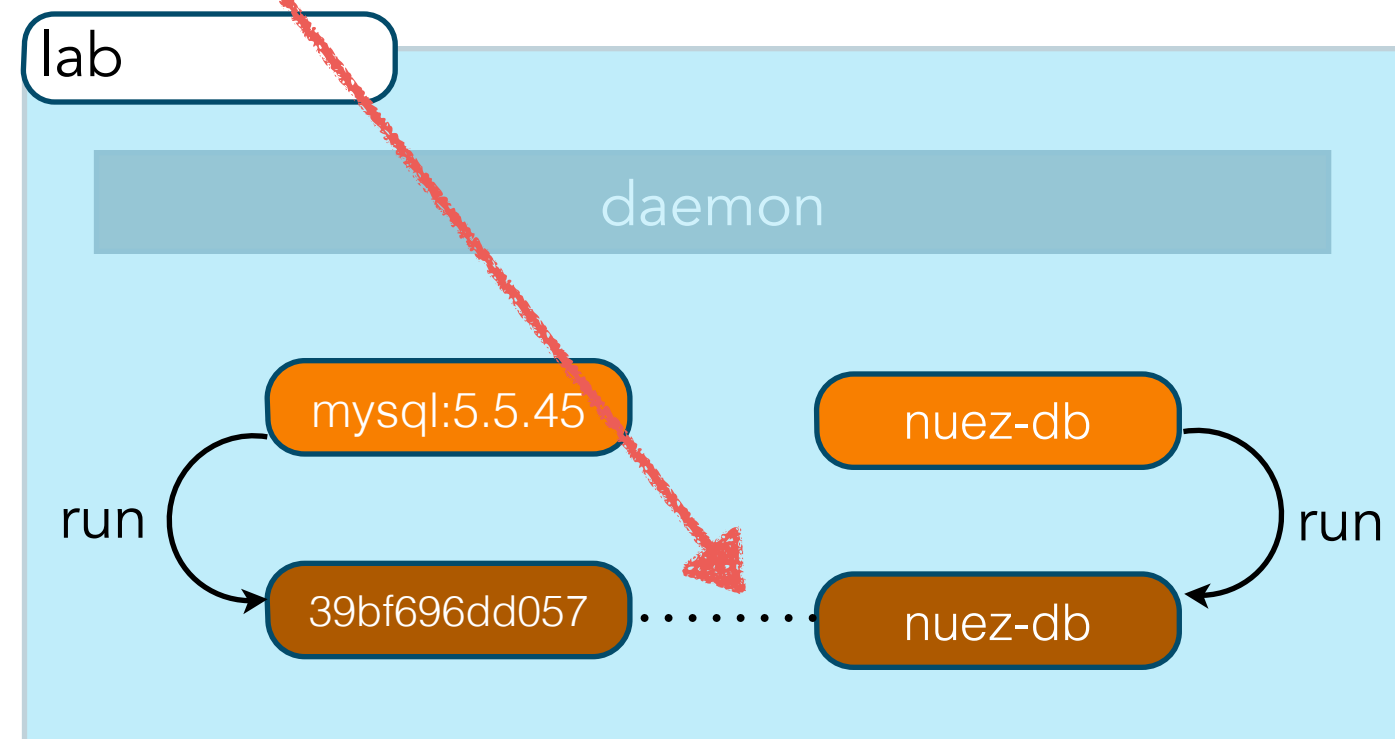
```
mysql -h 192.168.99.102 -u root -proot+1 nuev
```

or

```
docker run -it --link nuev-db:mysql --rm mysql:5.5.45 bash
```

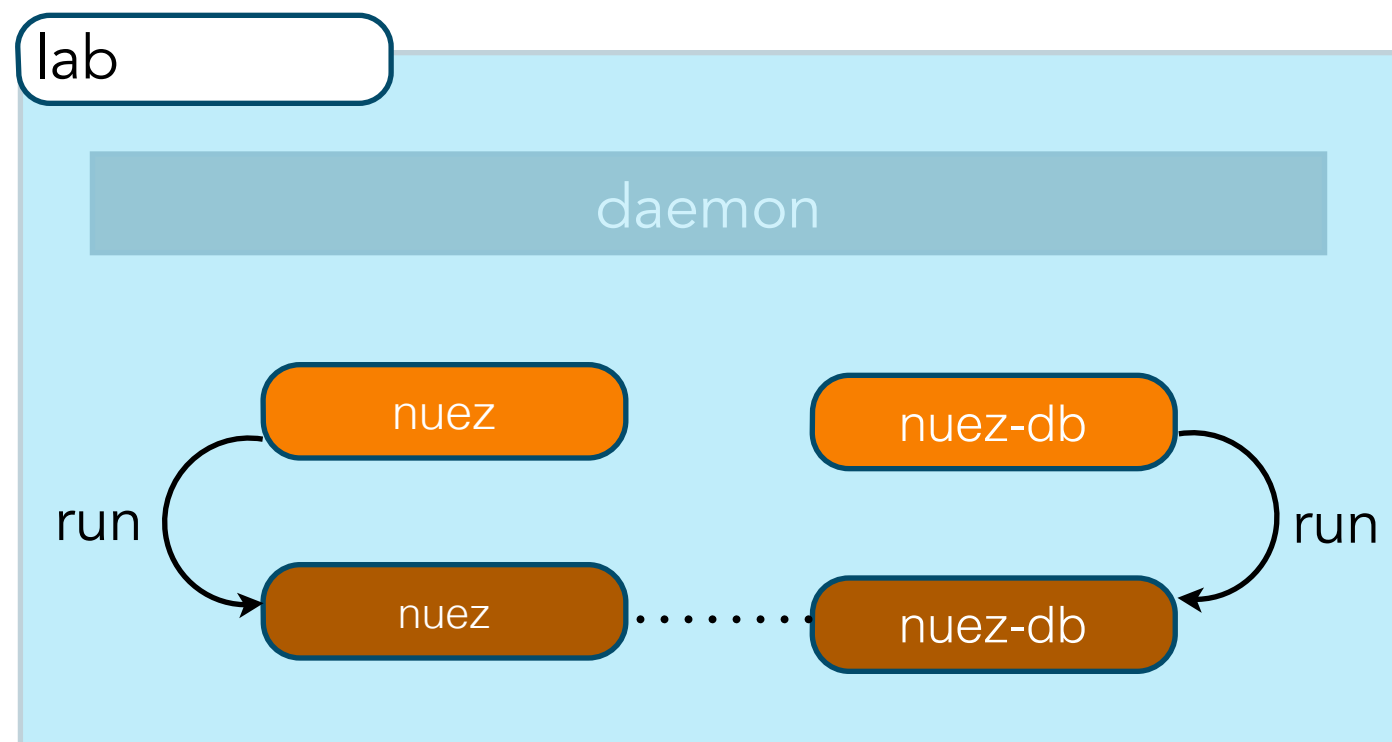
```
mysql -h $MYSQL_PORT_3306_TCP_ADDR -u $MYSQL_ENV_MYSQL_USER -p$MYSQL_ENV_MYSQL_PASSWORD $MYSQL_ENV_MYSQL_DATABASE
```

```
HOSTNAME=05f85a8745c2
MYSQL_ENV_MYSQL_DATABASE=nuev
MYSQL_ENV_MYSQL_ROOT_PASSWORD=root+1
TERM=xterm
MYSQL_VERSION=5.5.45
MYSQL_PORT_3306_TCP_PORT=3306
MYSQL_PORT_3306_TCP=tcp://172.17.0.28:3306
PATH=/usr/local/sbin:/usr/local/bin
MYSQL_ENV_MYSQL_USER=nuev-app
PWD=/
MYSQL_ENV_MYSQL_PASSWORD=nuev+1
MYSQL_ENV_MYSQL_VERSION=5.5.45
HOME=/root
SHLVL=1
MYSQL_PORT_3306_TCP_PROTO=tcp
MYSQL_NAME=/elated_franklin/mysql
MYSQL_MAJOR=5.5
MYSQL_PORT_3306_TCP_ADDR=172.17.0.28
MYSQL_ENV_MYSQL_MAJOR=5.5
MYSQL_PORT=tcp://172.17.0.28:3306
_=/usr/bin/env
```



Lab 5

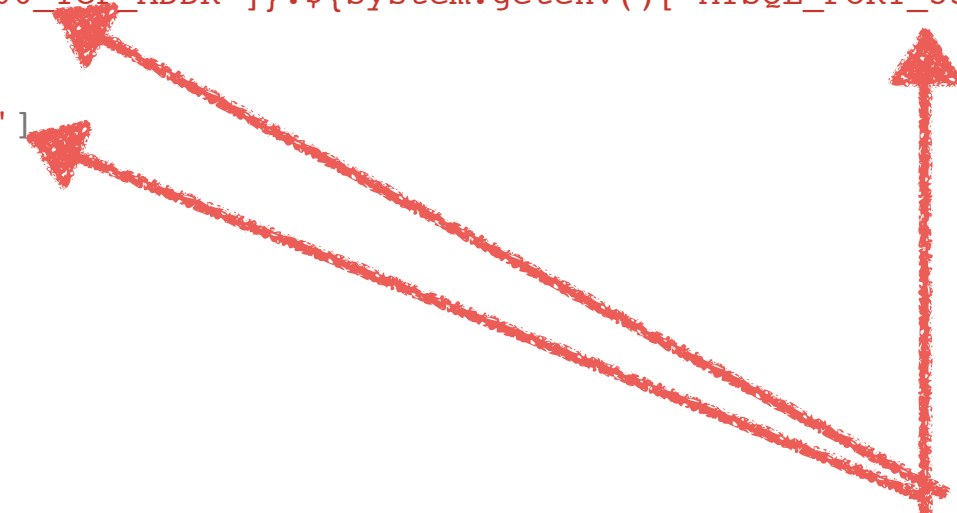
- I. Run a mysql command-line in a container and connect to the nuev-db.



```
// remove for brevity

docker {
  dataSource {
    dbCreate = "update"
    url = "jdbc:mysql://${System.getenv()['MYSQL_PORT_3306_TCP_ADDR']}:${System.getenv()['MYSQL_PORT_3306_TCP_PORT']}/nuez"
    driverClassName = "com.mysql.jdbc.Driver"
    username = System.getenv()['MYSQL_ENV_MYSQL_USER']
    password = System.getenv()['MYSQL_ENV_MYSQL_PASSWORD']
    pooled = true
    properties {
      maxActive = -1
      minEvictableIdleTimeMillis=1800000
      timeBetweenEvictionRunsMillis=1800000
      numTestsPerEvictionRun=3
      testOnBorrow=true
      testWhileIdle=true
      testOnReturn=true
      validationQuery="SELECT 1"
    }
  }
}

// remove for brevity
```



Three red arrows originate from a common point on the right side of the image and point to the following placeholders in the code:

- `MYSQL_PORT_3306_TCP_ADDR` in the `url` property.
- `MYSQL_PORT_3306_TCP_PORT` in the `url` property.
- `MYSQL_ENV_MYSQL_PASSWORD` in the `password` property.

Dockerfile

```
FROM tomcat:7.0.65-jre7
```

```
RUN rm -rf $CATALINA_HOME/webapps/ROOT
```

```
RUN rm -rf $CATALINA_HOME/webapps/ROOT.war
```

```
RUN rm -rf $CATALINA_HOME/webapps/docs
```

```
RUN rm -rf $CATALINA_HOME/webapps/examples
```

```
RUN rm -rf $CATALINA_HOME/webapps/manager
```

```
RUN rm -rf $CATALINA_HOME/webapps/host-manager
```

```
ENV CATALINA_OPTS -Dgrails.env=docker
```

```
ADD https://s3.amazonaws.com/cmj-presentations/docker-clouddevelop-2015/nuez-0.1.war  
    $CATALINA_HOME/webapps/ROOT.war
```

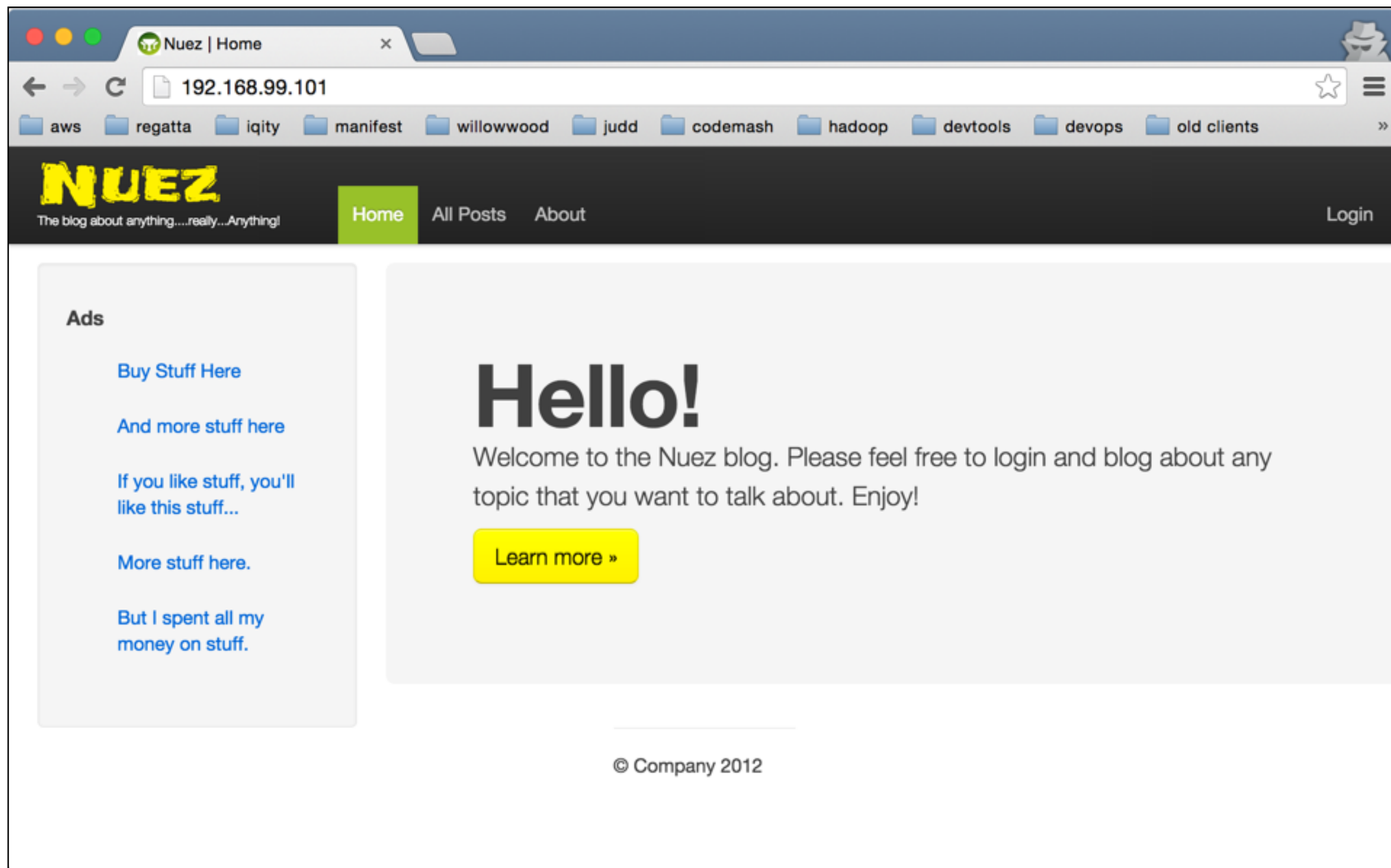
```
CMD ["catalina.sh", "run"]
```

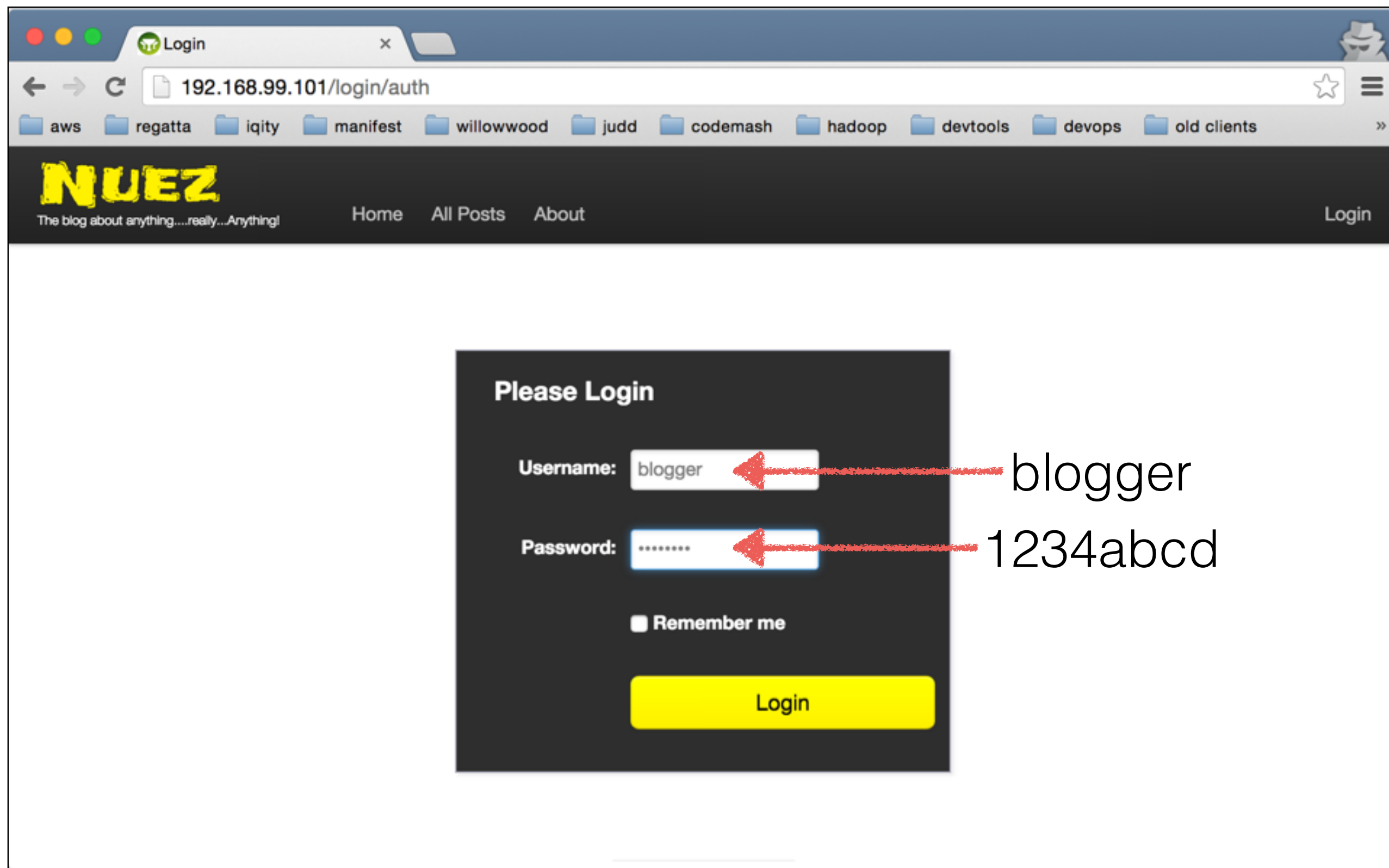
```
docker build -t nuez .
```

```
docker run -d --name nuev --link nuev-db:mysql -p 80:8080 nuev
```

or

```
docker run -it --rm --name nuev --link nuev-db:mysql -p 80:8080 nuev
```





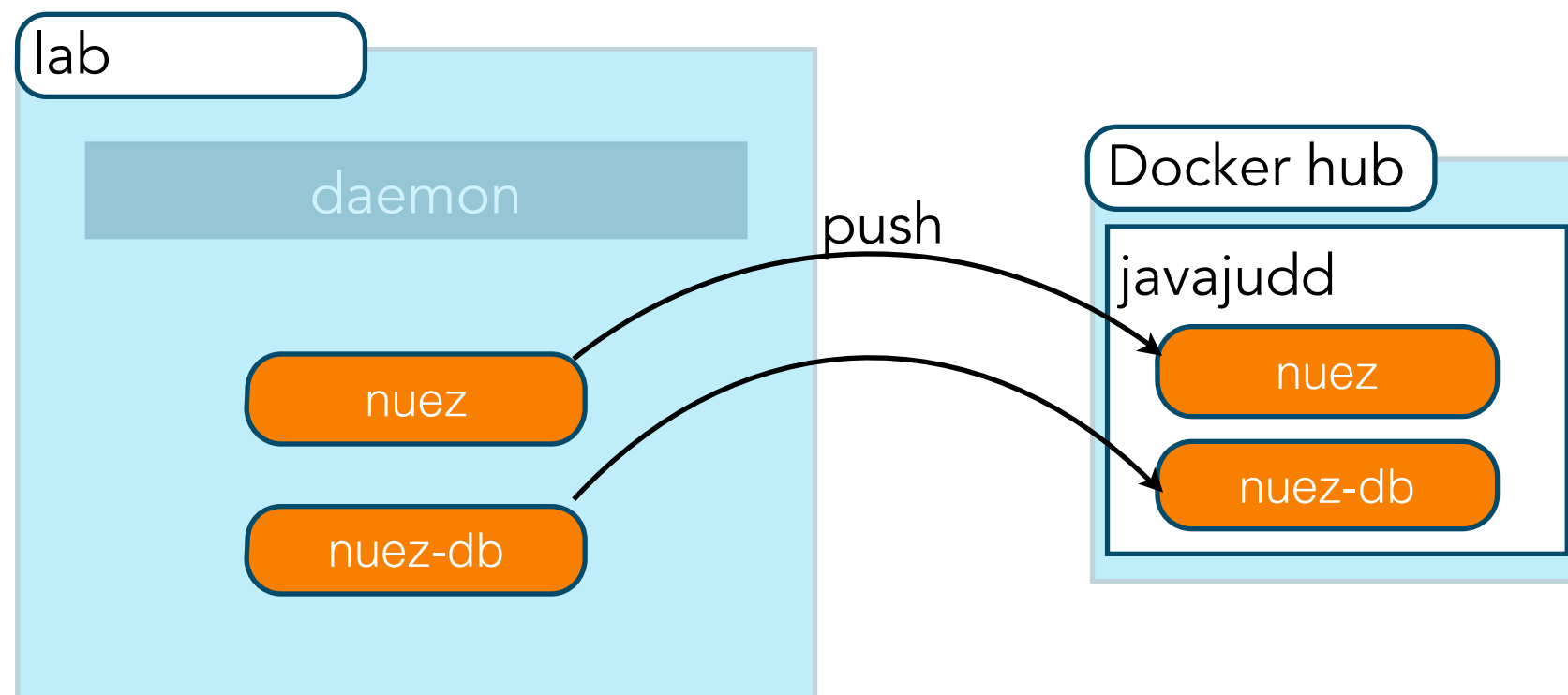
```
docker exec -it nuev bash
```

```
docker diff nuev
```

Lab 6

1. Create a new image containing Tomcat 7 and Java 7
2. Run the container
3. Test the container
4. Look at the Tomcat log files

SHARE IMAGES



```
docker build -t javajudd/nuez-db:1.0 .  
docker push javajudd/nuez-db
```

```
docker build -t javajudd/nuez:1.0 .  
docker push javajudd/nuez
```



PUBLIC REPOSITORY

javajudd/nuez-db ☆

Last pushed: a few seconds ago

[Repo Info](#)

[Tags](#)

[Collaborators](#)

[Webhooks](#)

[Settings](#)

Short Description



Mysql container for hosting the nuez database.

Full Description



Full description is empty for this repo.

Docker Pull Command



```
docker pull javajudd/nuez-db
```

Owner



javajudd

Comments (0)

[Add Comment](#)



PUBLIC REPOSITORY

[javajudd/nuez-db](#) ☆

Last pushed: a minute ago

[Repo Info](#)[Tags](#)[Collaborators](#)[Webhooks](#)[Settings](#)

Tags

Tag	Size
latest	70 MB
1.0	70 MB

Docker Pull Command



```
docker pull javajudd/nuez-db
```

Owner

[javajudd](#)

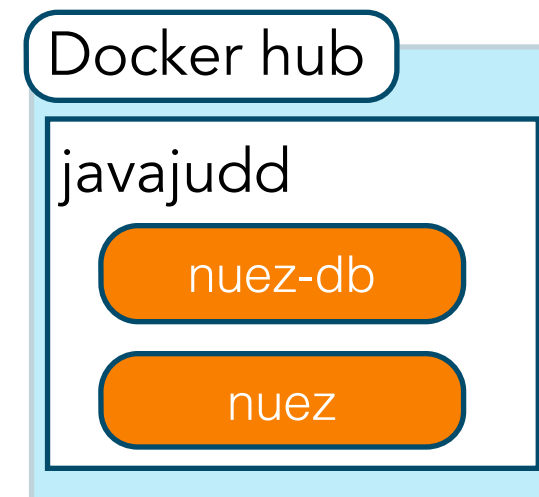
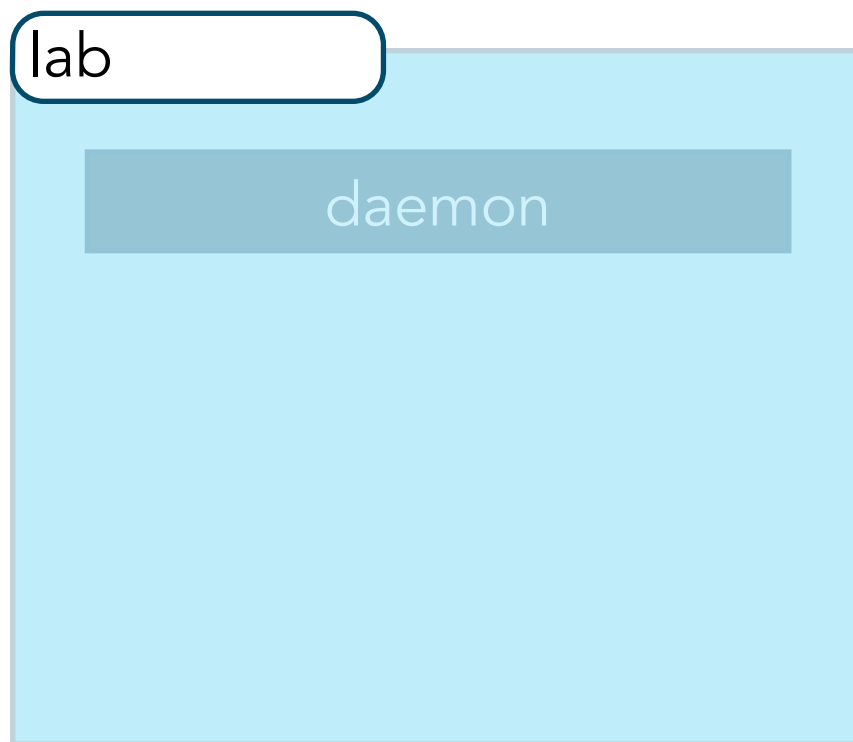
Comments (0)

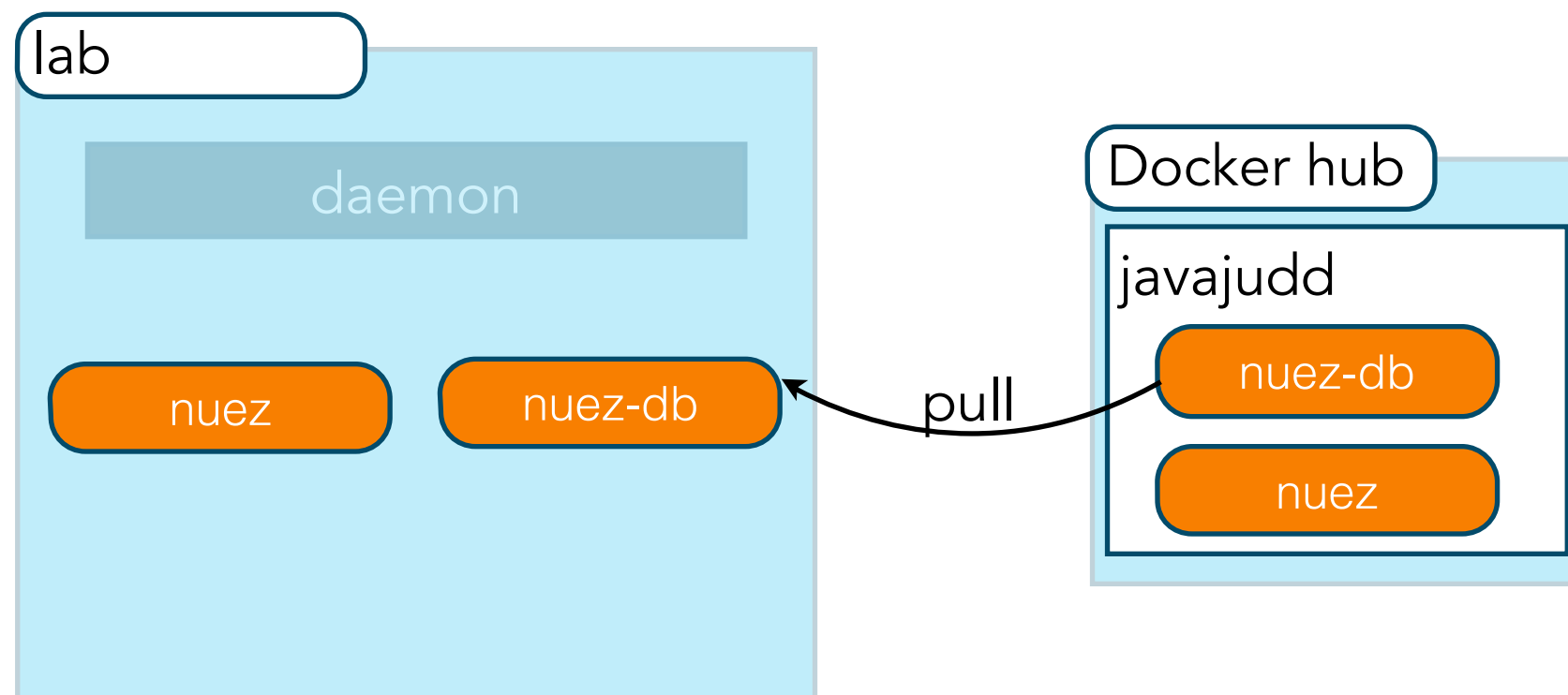
[Add Comment](#)

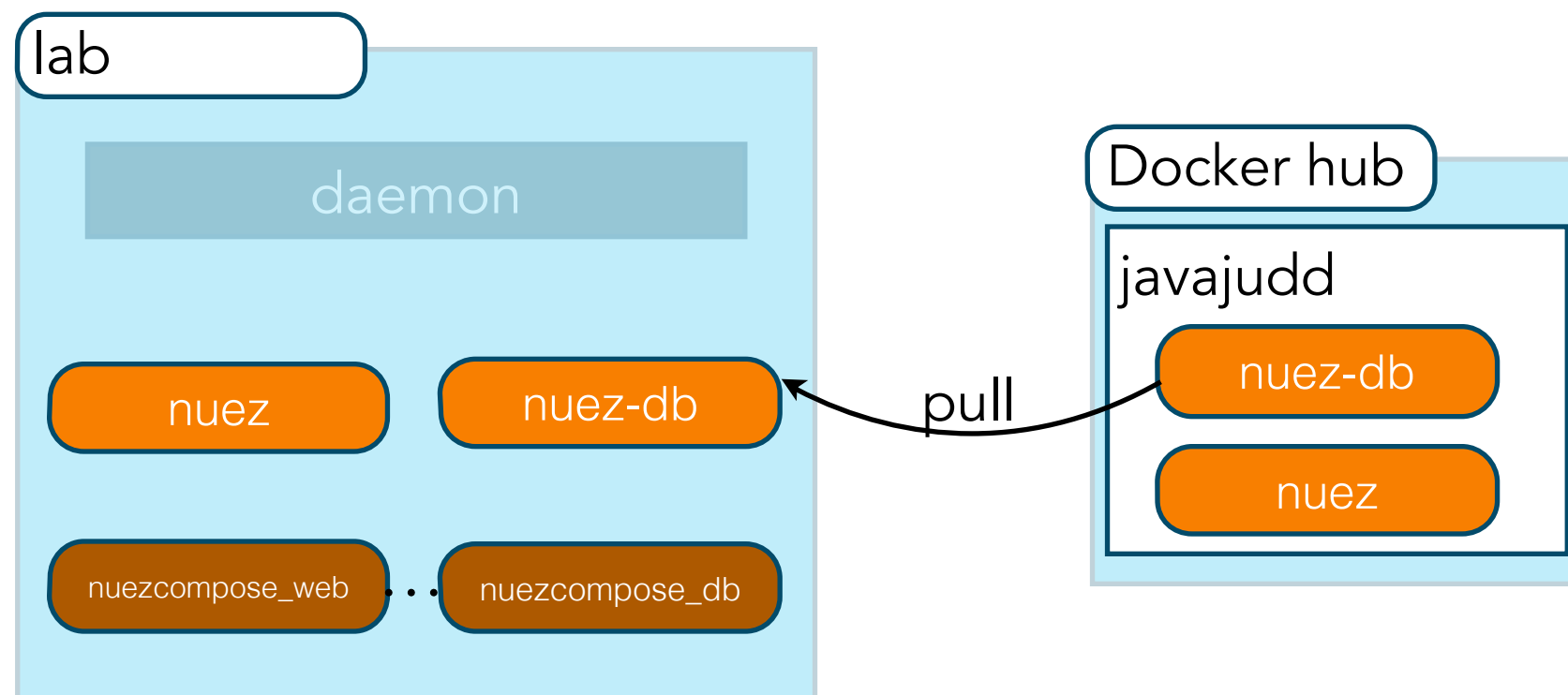
Lab 7

- I. Push nuev and nuev-db docker containers to your repository

COMPOSING







docker-compose.yml

```
web:
  image: javajudd/nuez
  ports:
    - "80:8080"
  links:
    - "db:mysql"
db:
  image: javajudd/nuez-db
  ports:
    - "3306:3306"
  environment:
    MYSQL_USER: nuez-app
    MYSQL_PASSWORD: nuez+1
    MYSQL_DATABASE: nuez
    MYSQL_ROOT_PASSWORD: root+1
```

`docker-compose up -d`

CONTAINER ID	IMAGE	COMMAND	STATUS	PORTS	NAMES
b61f6580e69e13879ea2ae23	j	javajudd/nuez-db	"/entrypoint.sh mysql"	Up 40 seconds	0.0.0.0:3306->3306/tcp
					nuezcompose_db_1

Lab 8

I. Compose nuev and nuev-db

RESOURCES

