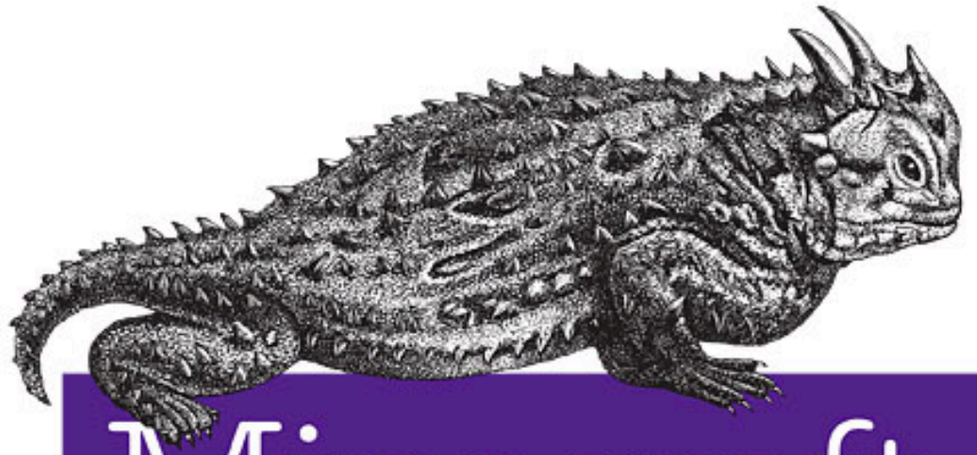
Docker Introduction

Arun Gupta
Vice President, Developer Advocacy
@arungupta, blog.arungupta.me
arun@couchbase.com

DEVOXX™ 4KIDS USA

O'REILLY®



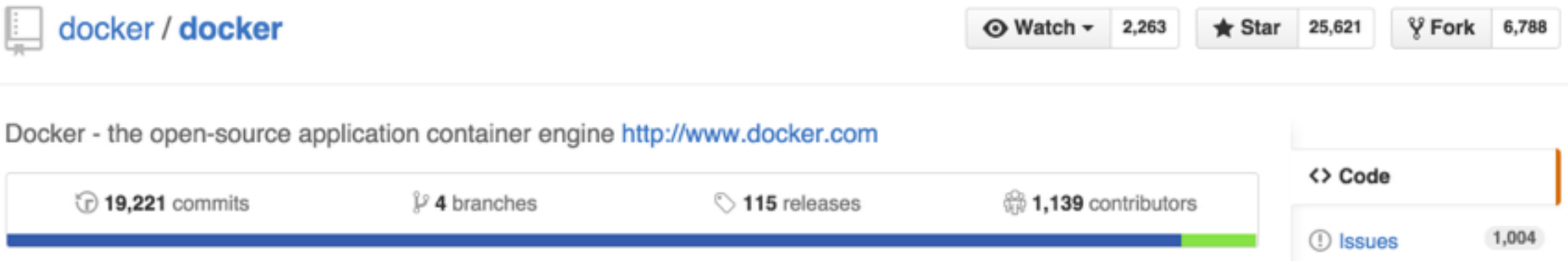
Minecraft Modding with Forge

A FAMILY-FRIENDLY GUIDE TO BUILDING FUN MODS IN JAVA

Arun Gupta & Aditya Gupta

What is Docker?

- Open source project and company



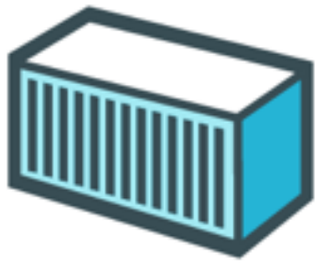
- Used to create containers for software applications
- Package Once Deploy Anywhere (PODA)

Advantages of Containers

- Immutability
- Reproducibility
- Isolation
- Faster deployments
- Portability - “it works on my machine”
- Snapshotting
- Security sandbox
- Limit resource usage
- Simplified dependency
- Sharing

Build

Develop an app using Docker containers with any language and any toolchain.

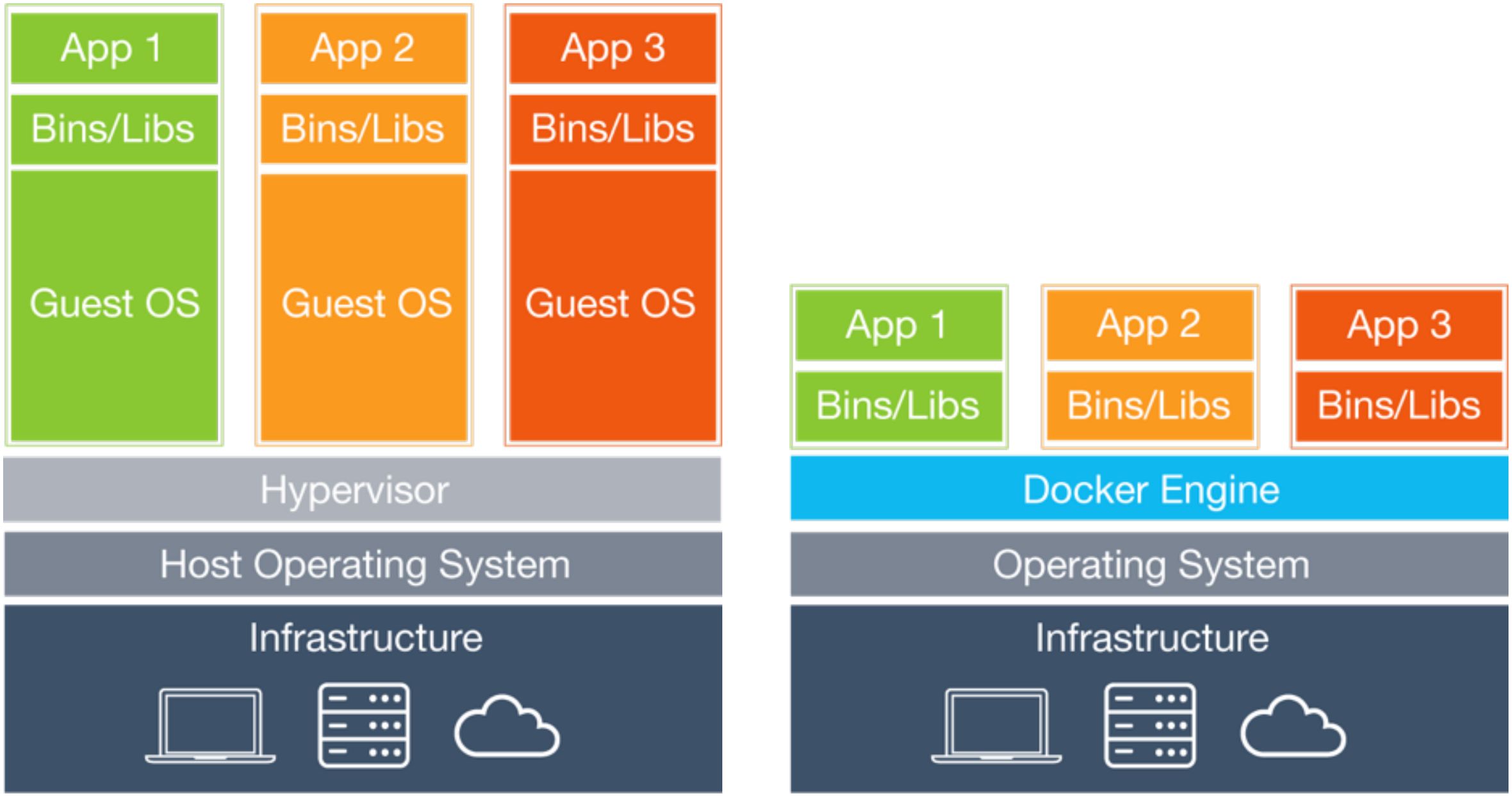


Ship

Ship the “Dockerized” app and dependencies anywhere - to QA, teammates, or the cloud - without breaking anything.

Run

Scale to 1000s of nodes, move between data centers and clouds, update with zero downtime and more.


App 1
Bins/Libs
Guest OS
App 2
Bins/Libs
Guest OS
App 3
Bins/Libs
Guest OS
Hypervisor
Host Operating System
Infrastructure
App 1
Bins/Libs
App 2
Bins/Libs
App 3
Bins/Libs
Docker Engine
Operating System
Infrastructure

- Image defined in text-based **Dockerfile**
- List of commands to build the image

Build

Develop an app using Docker containers with any language and any toolchain.

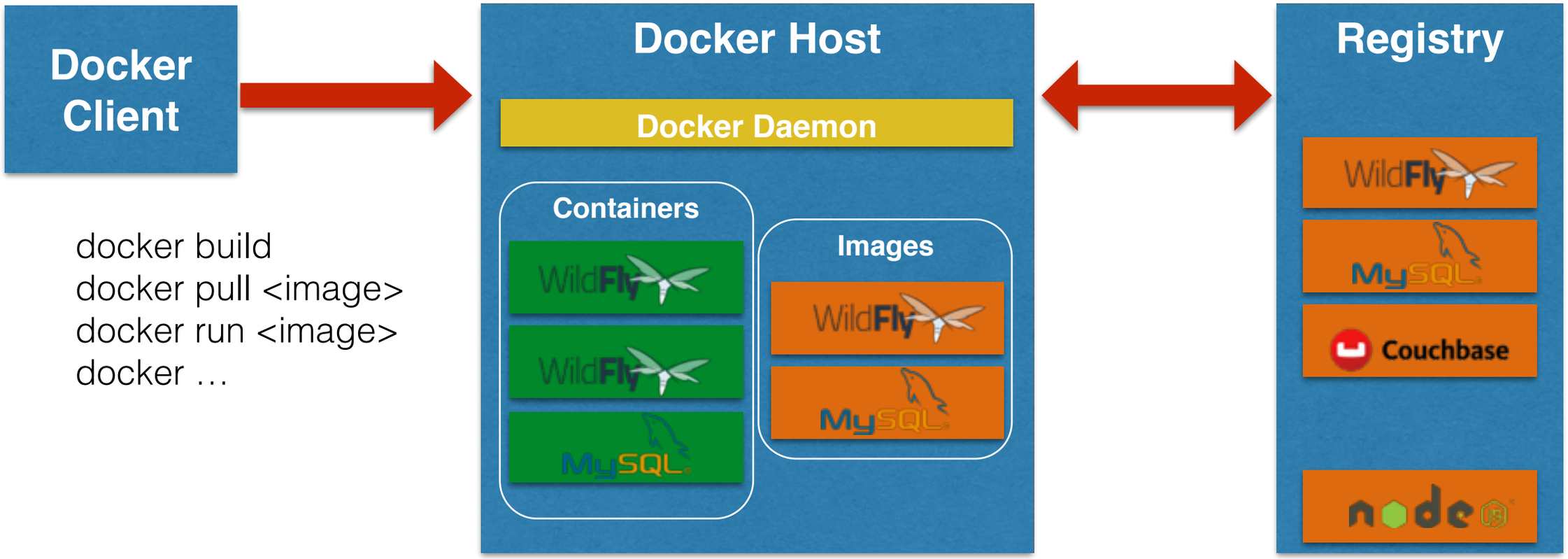
```
FROM fedora:latest

CMD echo “Hello world”
```

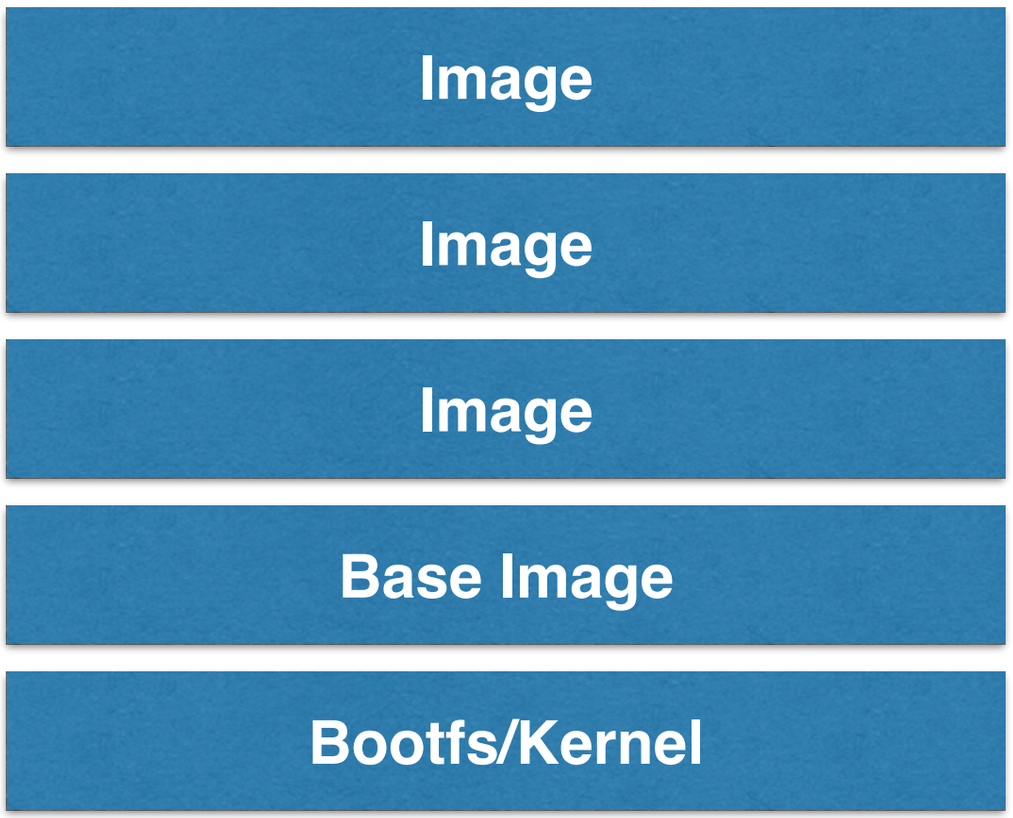
```
FROM jboss/wildfly

RUN curl -L https://github.com/javaee-samples/javaee7-hol/raw/master/solution/
movieplex7-1.0-SNAPSHOT.war -o /opt/jboss/wildfly/standalone/deployments/
movieplex7-1.0-SNAPSHOT.war
```

Docker Workflow



docker build
docker pull <image>
docker run <image>
docker …



jboss/wildfly

jboss/base-jdk:8

jboss/base

centos:7

Docker Machine

- Create Docker Host on computer or cloud provider

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

  - Configure Docker client to talk to host
  - Create and pull images
  - Start, stop, restart containers
  - Upgrade Docker
- Not recommended for production yet

Docker Machine Providers

Microsoft Azure

Google

openstack™

DigitalOcean

SOFTLAYER

exoscale

Couchbase

Docker Compose

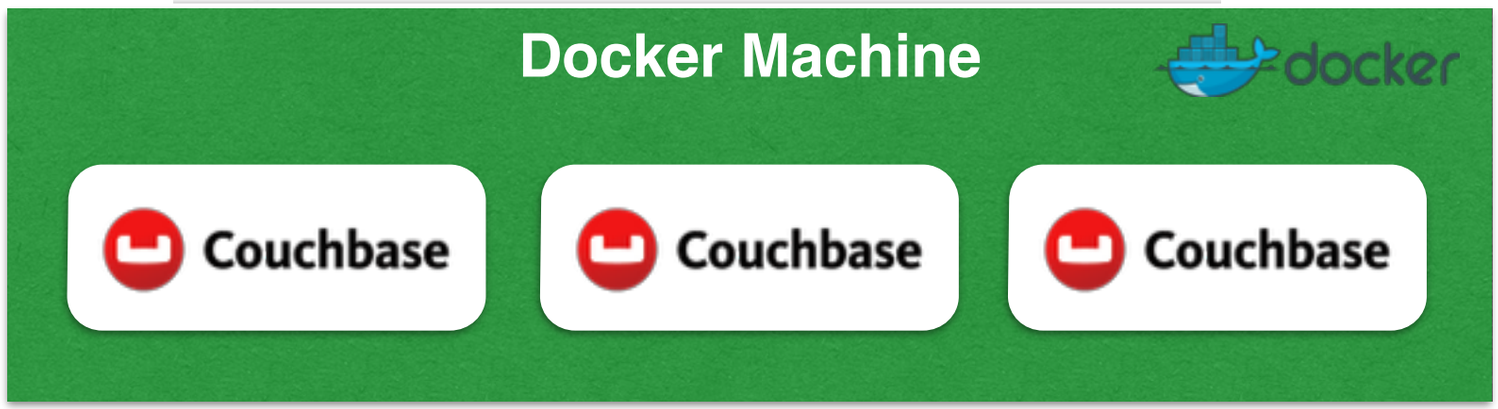
- Defining and running multi-container applications
- Configuration defined in a single file
- Great for dev, staging, and CI
- Not recommended for production yet

docker-compose.yml

```
mysqldb:
  image: mysql
  environment:
    MYSQL_DATABASE: sample
    MYSQL_USER: mysql
    MYSQL_PASSWORD: mysql
    MYSQL_ROOT_PASSWORD: supersecret
mywildfly:
  image: arungupta/wildfly-mysql-javaee7
  links:
    - mysqldb:db
```

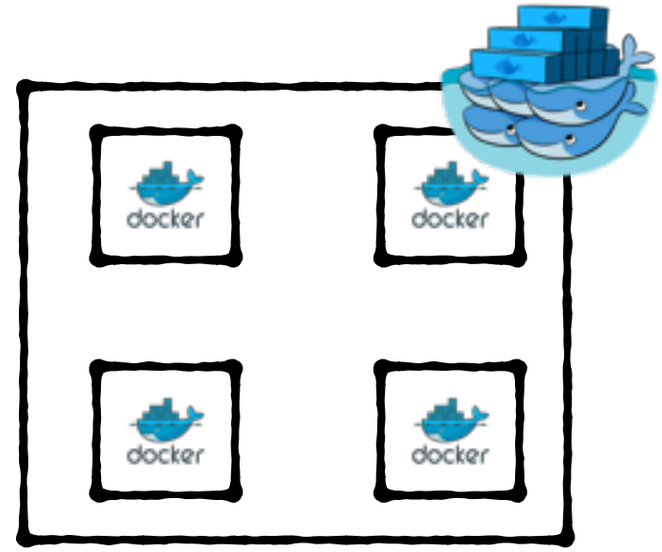
Couchbase Cluster using Docker Compose

```
couchbase1:
  image: couchbase/server
  volumes:
    - ~/couchbase/node1:/opt/couchbase/var
couchbase2:
  image: couchbase/server
  volumes:
    - ~/couchbase/node2:/opt/couchbase/var
couchbase3:
  image: couchbase/server
  volumes:
    - ~/couchbase/node3:/opt/couchbase/var
  ports:
    - 8091:8091
    - 8092:8092
    - 8093:8093
    - 11210:11210
```



Docker Swarm

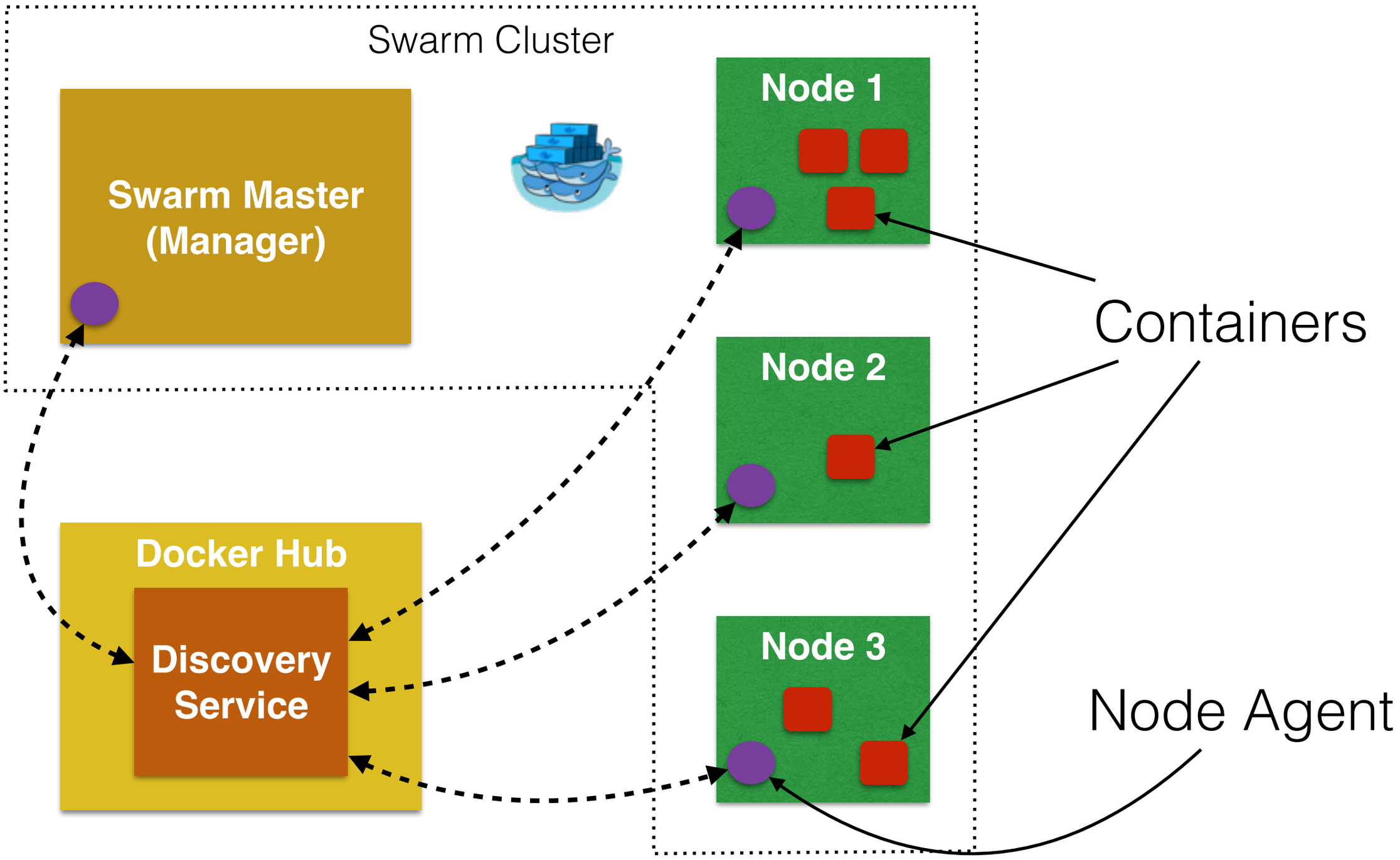
- Native clustering for Docker
- Provides a unified interface to a pool of Docker hosts
- Fully integrated with Machine
- Serves the standard Docker API
- Partially integrated with Compose
- 1.0.0 - Ready for production



Docker Swarm

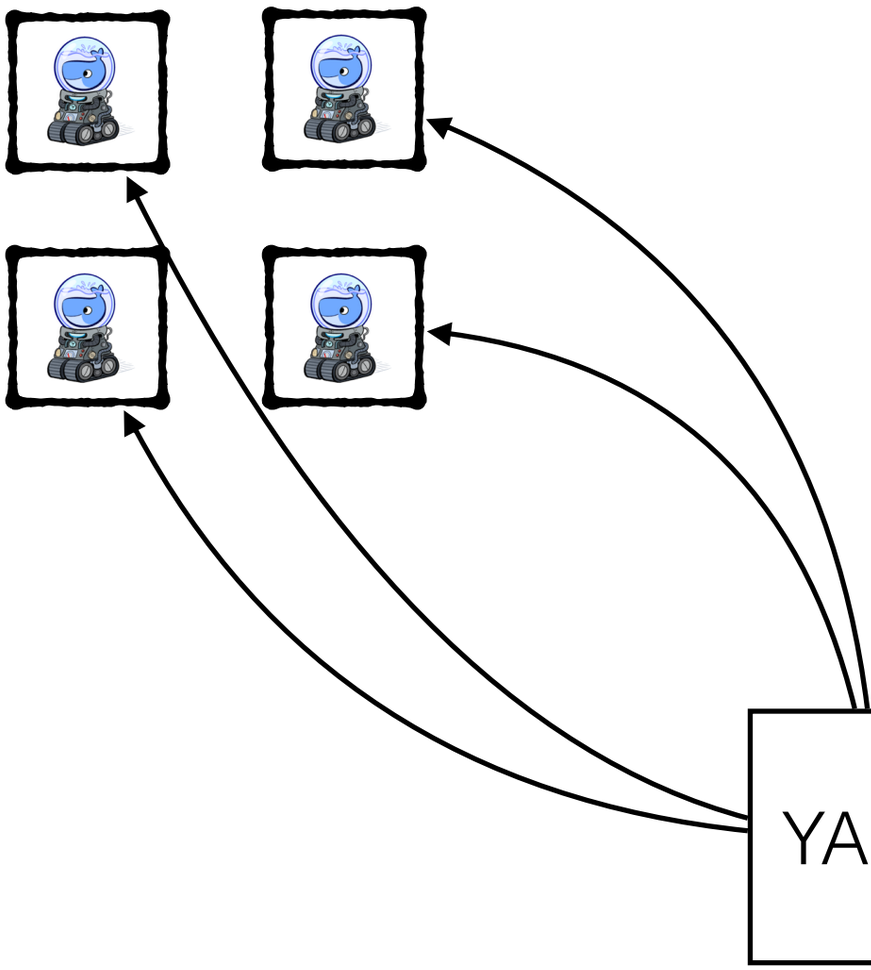
- Simple scheduling backends included, API for pluggable backends (e.g. Mesos) coming
  - Based on CPU (`-c`), RAM (`-m`), number of containers
  - `spread` (default): node with least number of running containers
  - `binpack`: node with most number of running containers
  - `random`: mostly for debugging
  - Specified using `--strategy`

Docker Swarm

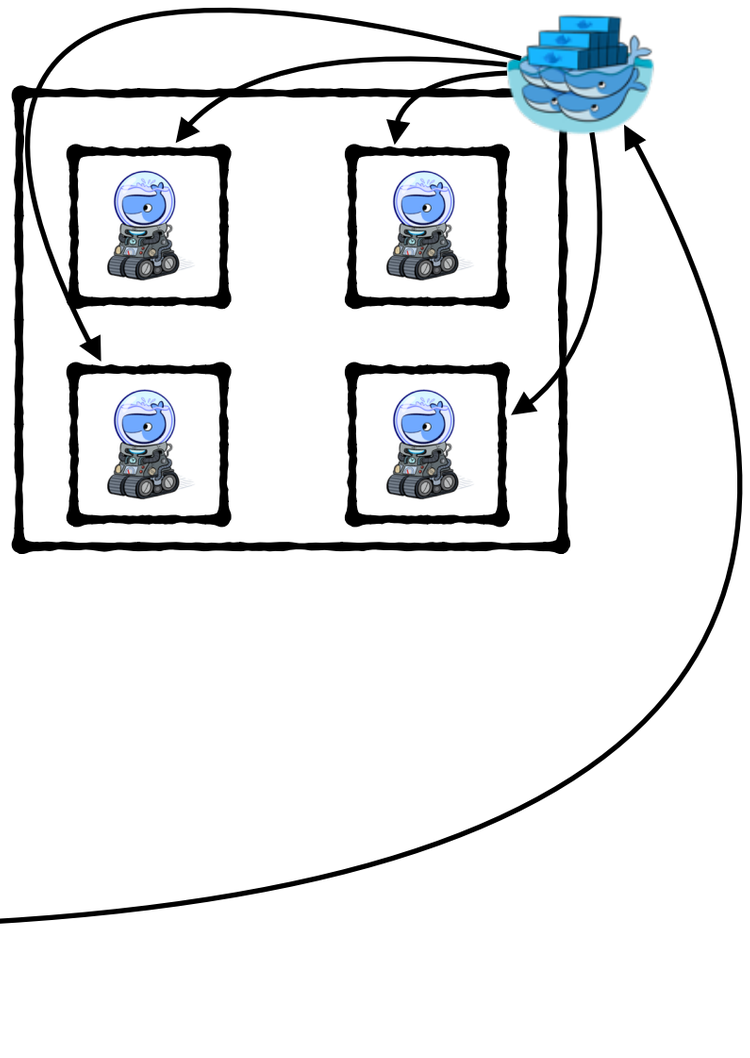


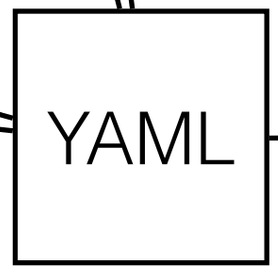
http://blog.arungupta.me/clustering-docker-swarm-techtip85/

Machine



Swarm





Compose

Docker Toolbox

- Docker Client 1.9.0
- Docker Machine 0.5.0
- Docker Compose 1.5.0 ~~(Mac only)~~
- Docker Kitematic 0.9.3
- Boot2Docker ISO 1.9.0
- Virtualbox 5.0.8

https://www.docker.com/toolbox

Persistent Storage

- Data volumes - used to persist data independent of container's lifecycle
- Multiple plugins: Flocker, Ceph, . . .
- `docker run -d -P -v /opt/couchbase/var couchbase/server`
- `docker run -d -P -v ~/couchbase/:/opt/couchbase/var couchbase/server`

Virtual Networks

- Backed by multiple plugins
  - Calico, Cisco, Weave, . . .

Continuous Delivery using Docker and Jenkins Workflow