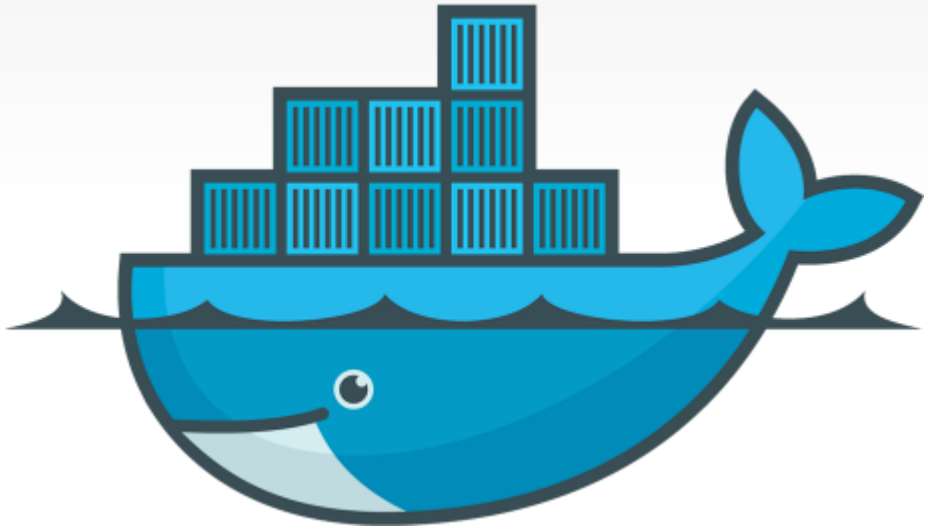
Creative Solutions

www.CSLWorld.com

Introduction To Docker



Kanishka Dilshan @kdkanishka
Eranga Bandara Herath @itseranga

About Pagero AB

#Domain

- ✓The global business network for e-invoice and e-order
- ✓Largest e-invoice/e-order provider in Europe

#Technologies

- ✓Scala
- ✓Java
- ✓Python
- ✓Microservices
- ✓Docker/Devops
- ✓Android
- ✓IOS

PAGERO
your financial supply chain network

The Challenge #Scenario

“There is an enterprise application which is deployed on older Jboss version (v4) and based on java 7, It also uses a postgresql 9.1 db and depends on few micro services which are written in Scala (2.11)”

- ✓ How do I deploy these application with its dependencies in staging(test environment) and production environment?
- ✓ How much of effort it required??
- ✓ What are the problems that I need to face when deploying?

The challenge #Problems

- ✓ How to guarantee same dependency versions deploying in all environments
- ✓ How easy to deploy the changes frequently(if you are using continuous deliveries)
- ✓ How to migrate version updates(For an instance update Jboss to version 5)
- ✓ If bug raises in production environment, how to reproduce it in local/staging environments
- ✓ etc…

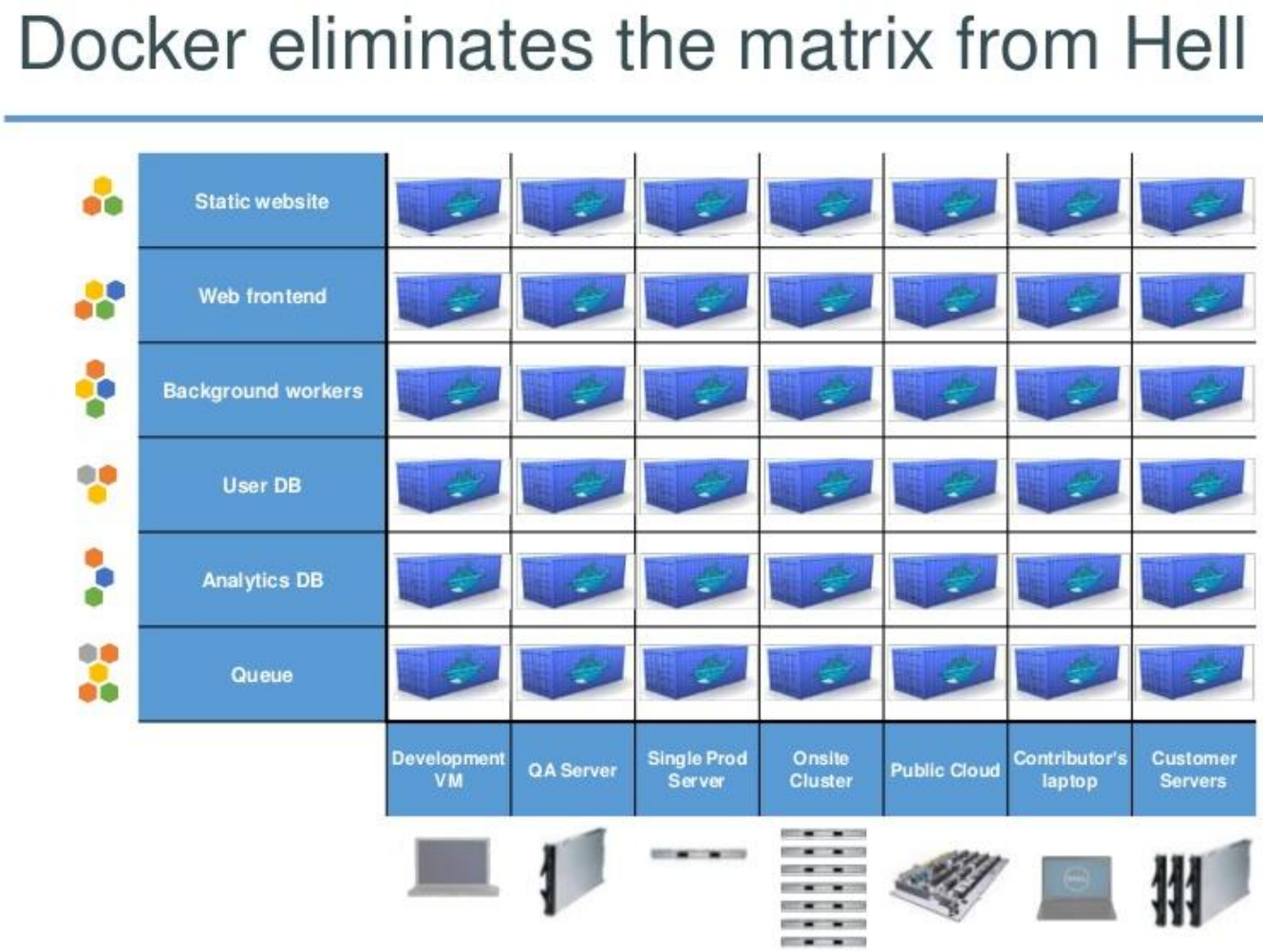
The Challenge

The Matrix From Hell

	Development VM	QA Server	Single Prod Server	Onsite Cluster	Public Cloud	Contributor's laptop	Customer Servers
Static website	?	?	?	?	?	?	?
Web frontend	?	?	?	?	?	?	?
Background workers	?	?	?	?	?	?	?
User DB	?	?	?	?	?	?	?
Analytics DB	?	?	?	?	?	?	?
Queue	?	?	?	?	?	?	?

docker

The Solution #Docker

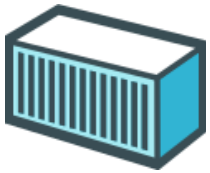


docker

Creative Solutions

Why Docker?

- Build



- Ship

- Run

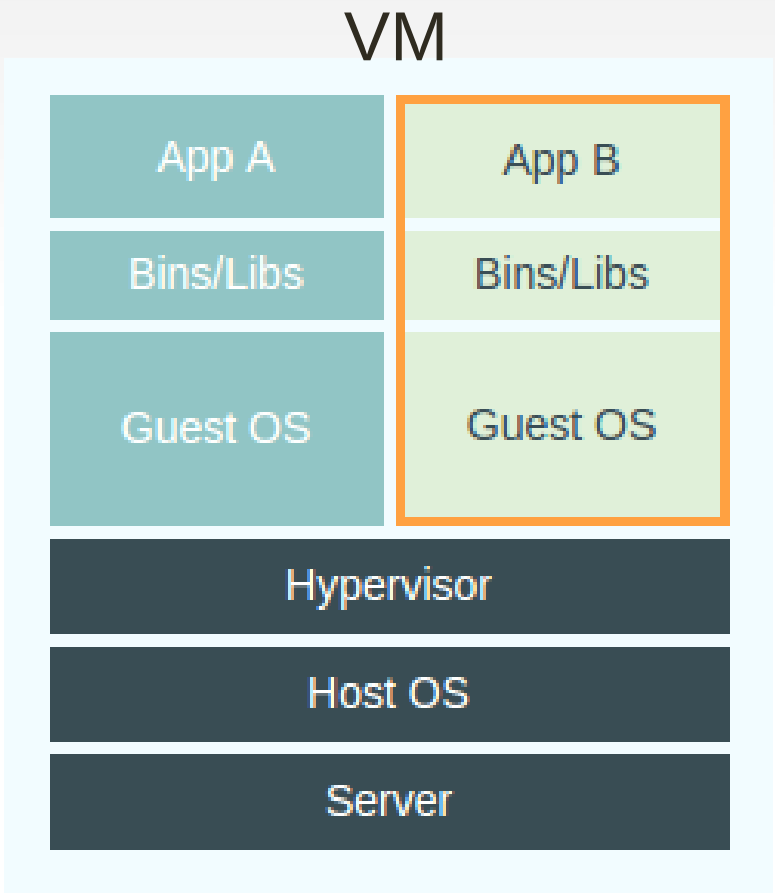
The solution #Docker

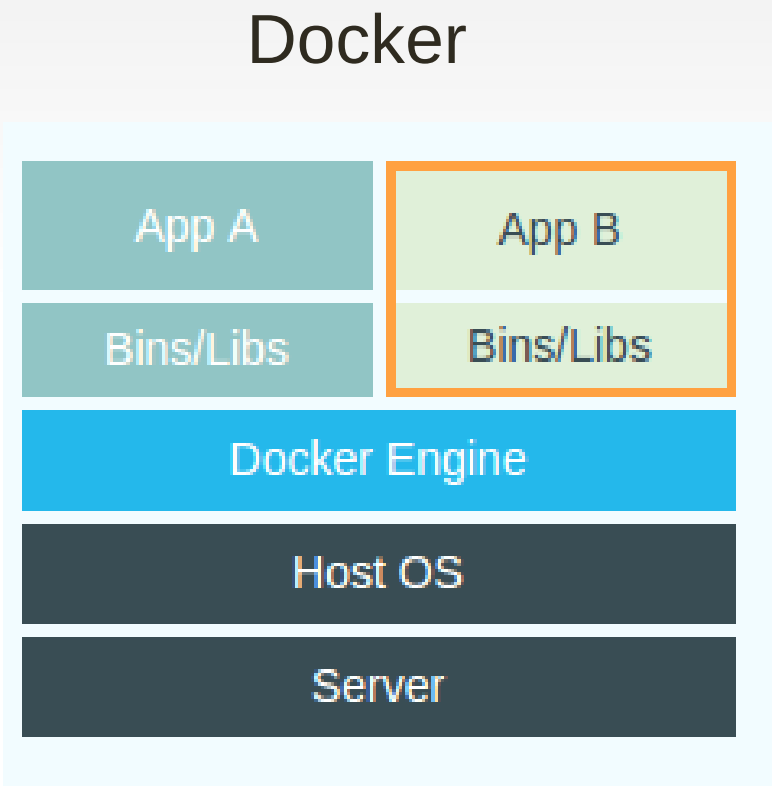
✓Docker is basically an open source tool for running isolated containers on Linux making the deployment of apps inside containers faster.

✓Docker creates portable, self-sufficient containers from any application.

“The same container that the developer builds and tests on his PC, can run in production, on VMs, in the cloud and a lot more places”

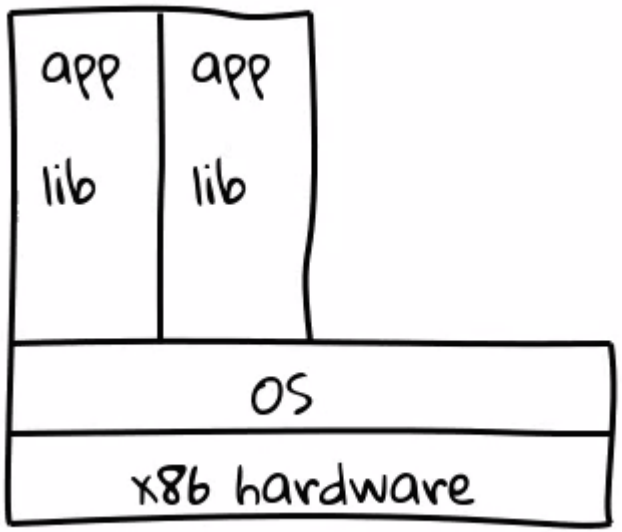
Docker vs Virtual Machines





More on Containers(LXc)

- ✓Simply containers provide OS level virtualization mechanism
- ✓Containers are group of processes in Linux box
- ✓Inside the box it looks like VM(But actually NOT a VM)
- ✓In outside the box it looks like group of processes

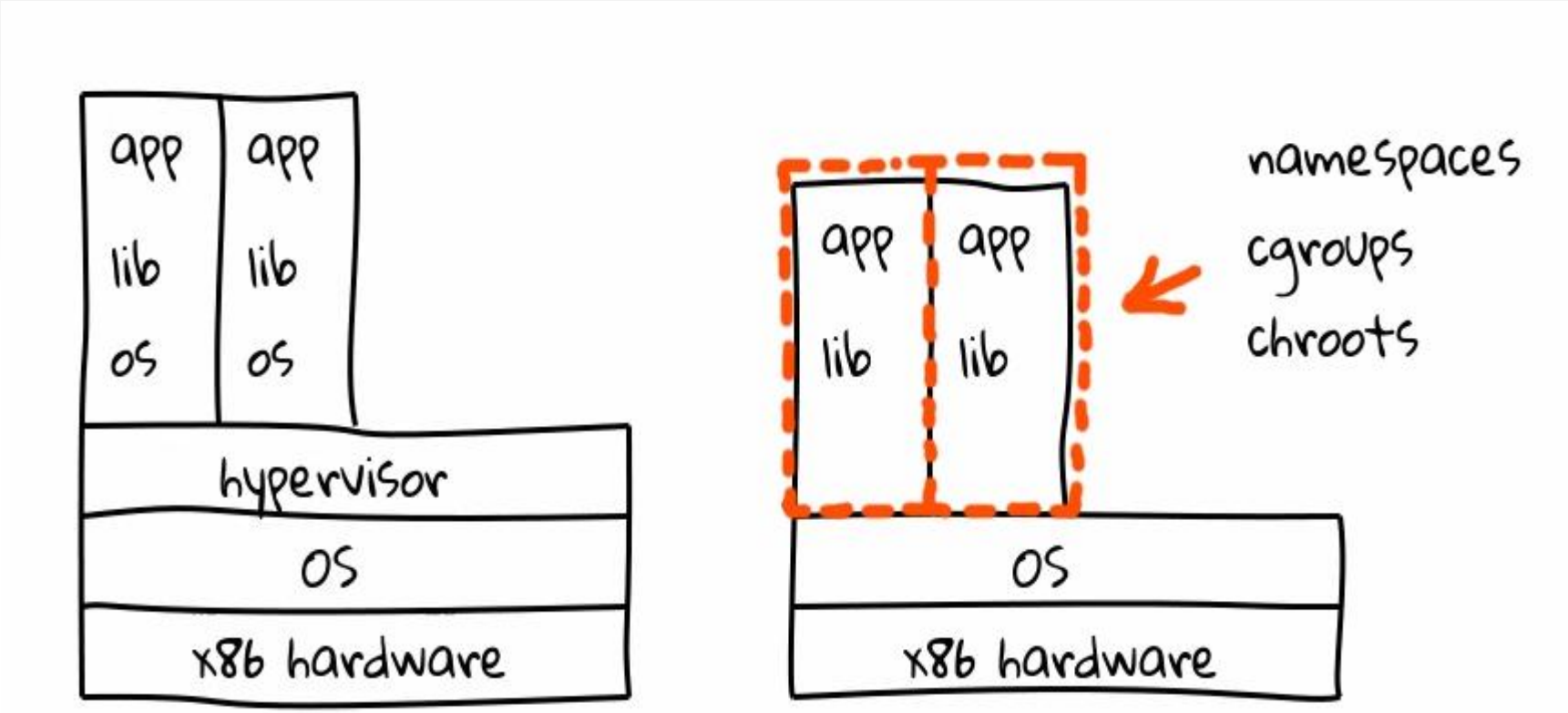


Containers vs VM

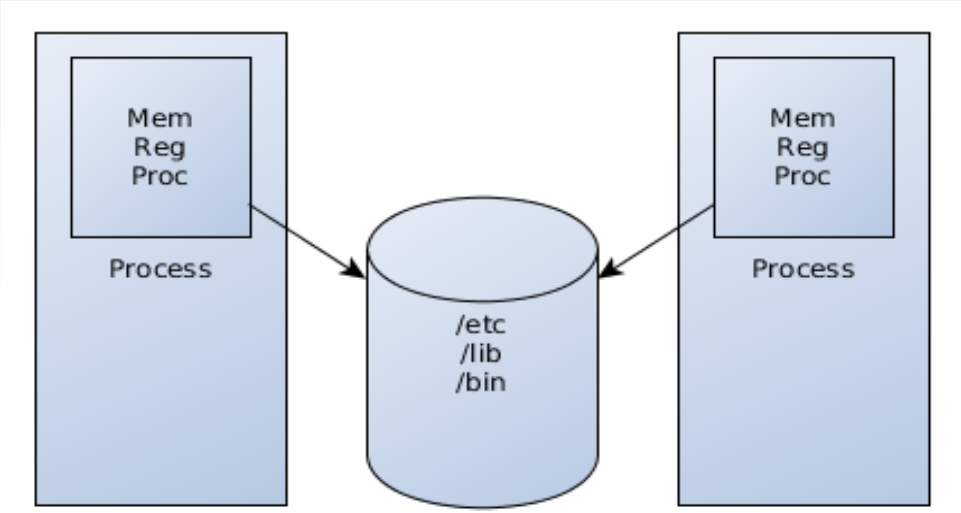
✓VMs consists with hypervisor which provides emulated hardware for virtual machine images
✓So VMs allows to creates many self contained systems via hypervisor
✓Containers are more light weight than VMs, since it shares kernal with host without hardware emulation(hypervisor)
✓Containers use kernal features such as kernel **namespaces**, and control groups(**cgroups**)
✓Kernel namespaces provide basic isolation and CGroups use for resource allocation

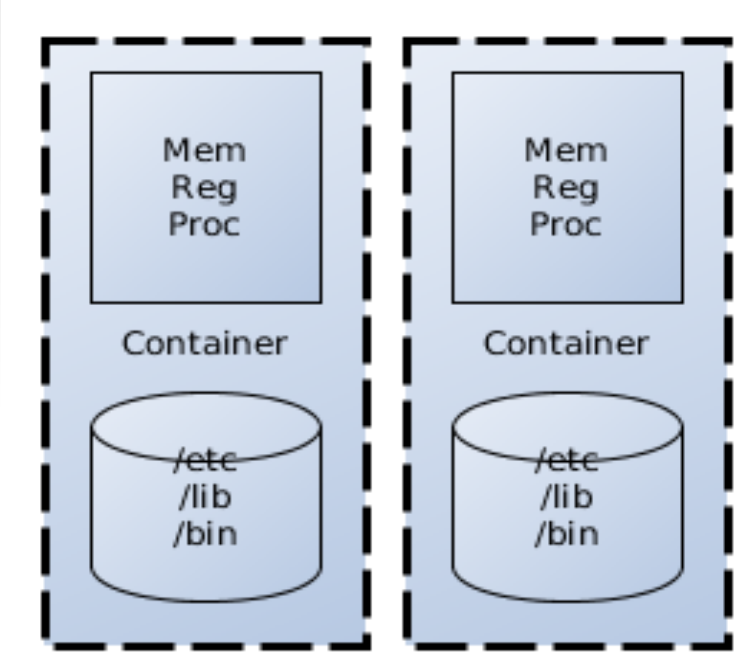
“Basically containers provide same functionality which provides by VMs, with out any hypervisor overhead”

Containers vs VM



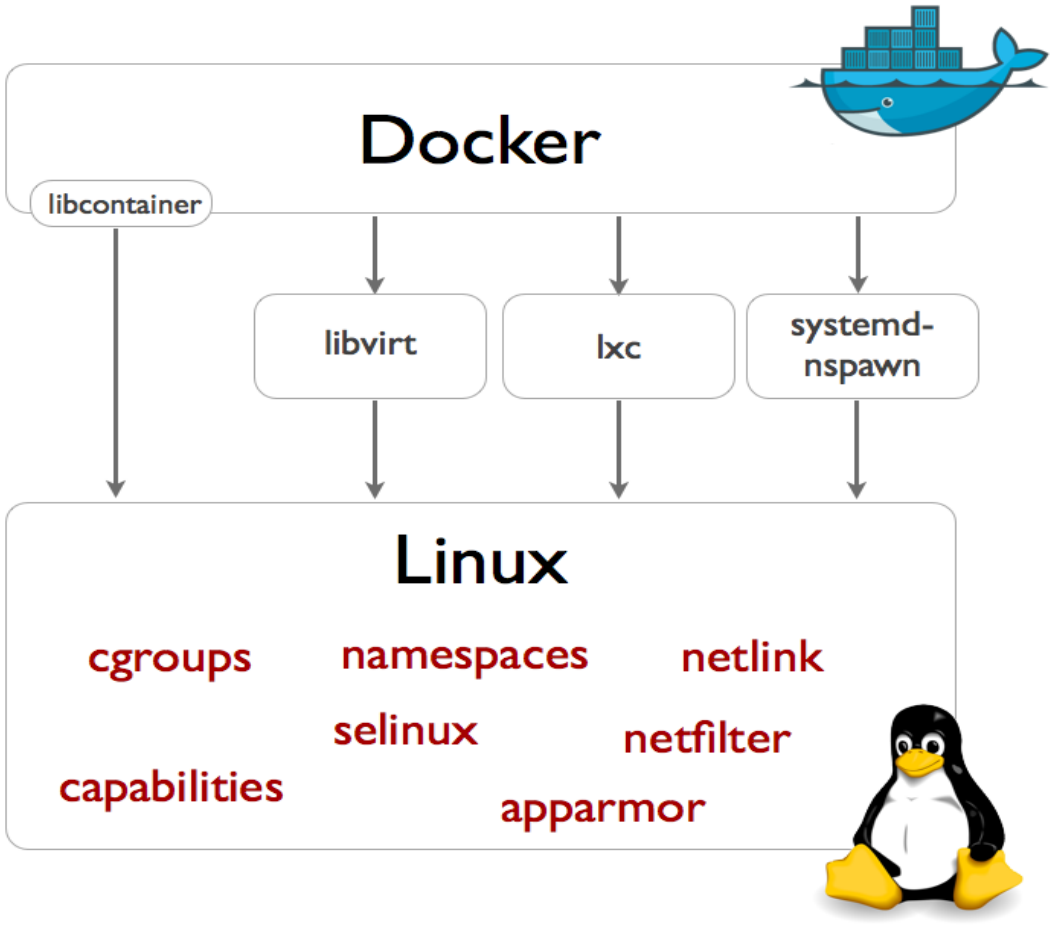
Process vs Container



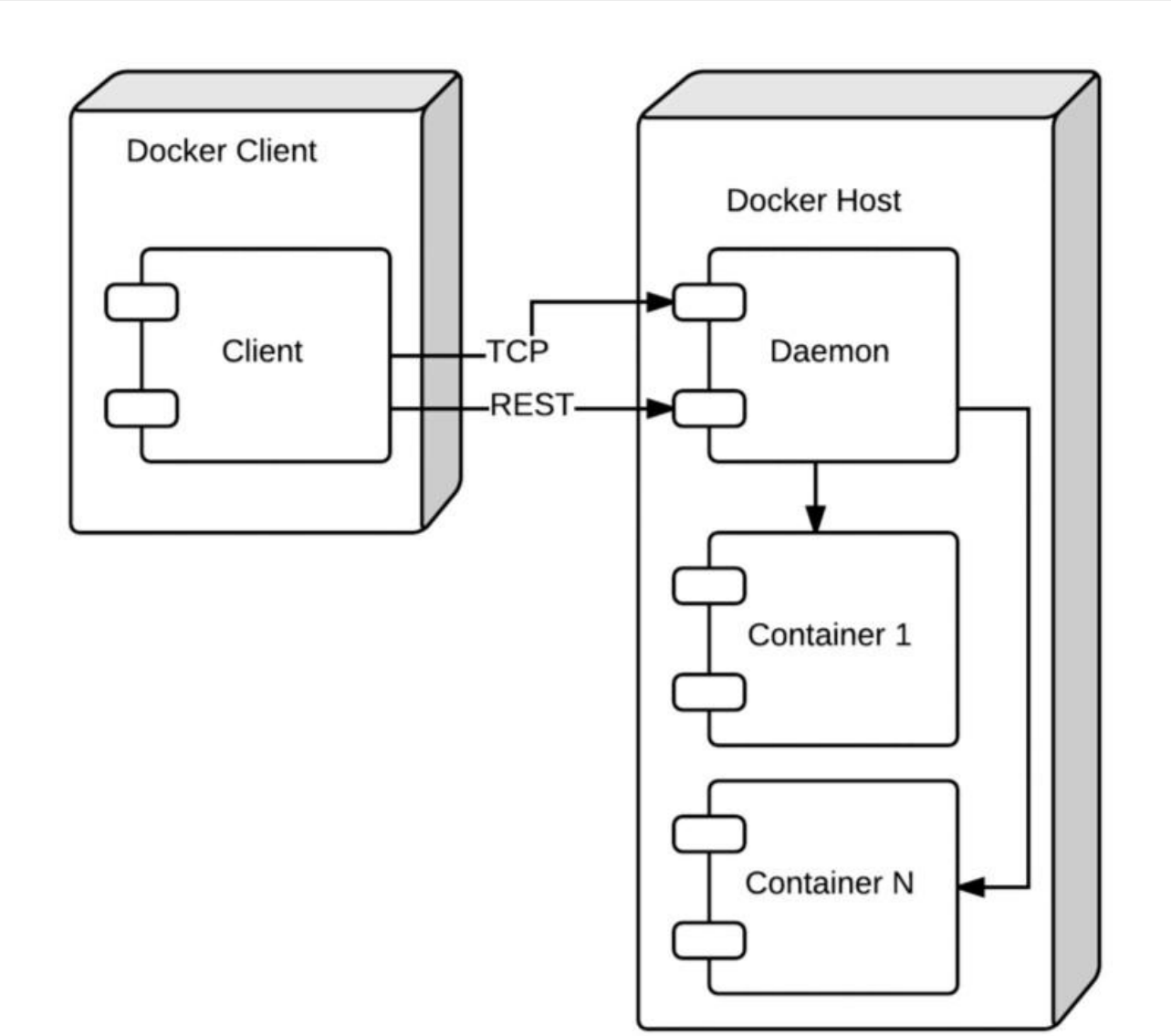


- Linux kernel features
  - Kernal namespaces
  - Control groups (cgroups)

Docker and LXc



The Docker Architecture



Setting up Docker

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

- Ubuntu 14.04
  - $ sudo apt-get update
  - $ sudo apt-get install docker.io
- Windows
  - Using Boot2Docker*
- Mac Osx
  - Using Boot2Docker*
- *Boot2Docker (~24Mb lightweight linux distribution)

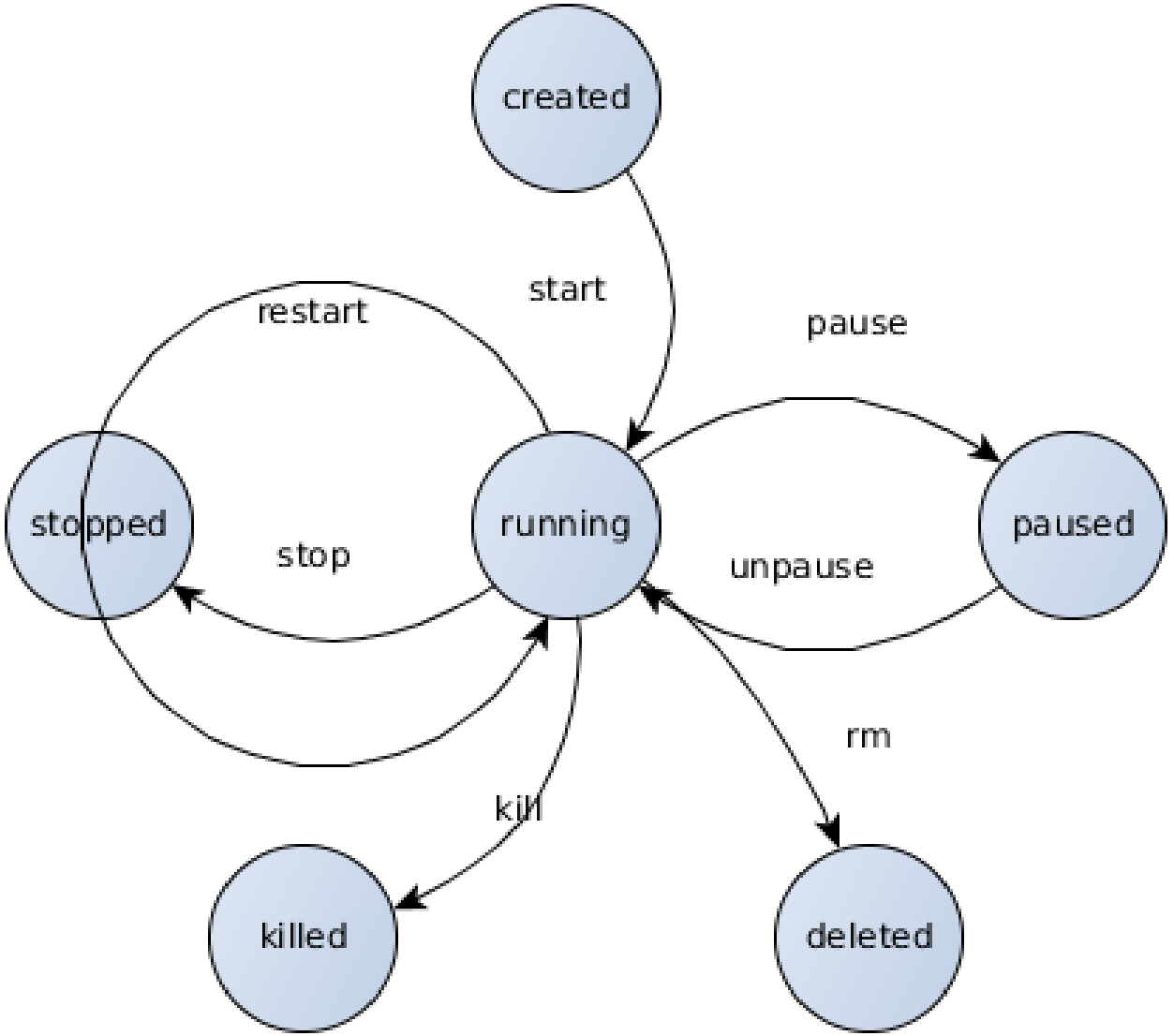
Docker CLI Commands

- search
- run
- ps
- start
- stop
- restart
- rm
- rmi
- kill
- commit
- pull

Docker CLI Examples

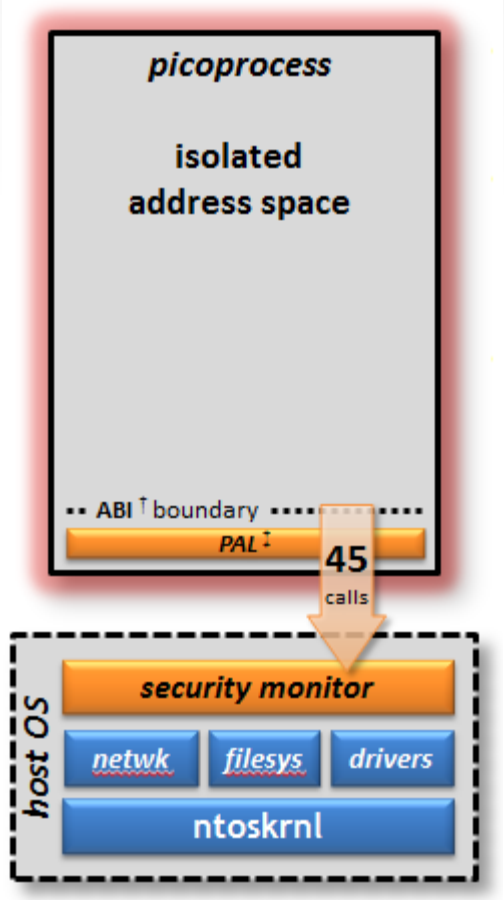
- Creating a simple container from an image
  - `docker run -i -t ubuntu:14.04 /bin/bash`
- Listing containers
  - `docker ps`
- Stop container
  - `docker stop <container hash>`
- Remove container
  - `docker rm <container hash>`

Container Lifecycle



MS Native Alternatives

- Drawbridge - Still in research level
  - picoprocess = container



Use Cases

- Easy application deployment
- Continuous Integration
- Continuous Delivery

Demo

https://github.com/kdkanishka/DockerTechTalk

Q&A