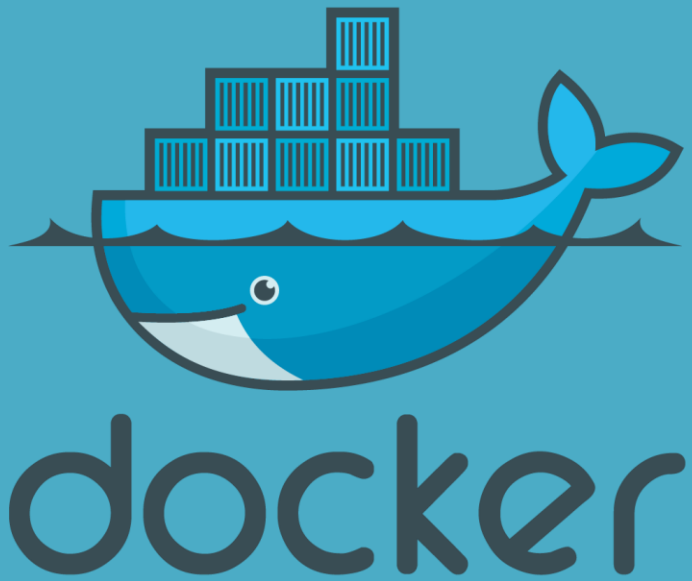


Docker

Vinicius Vielmo Cogo

Superclouds meeting, DI, FC/UL. March 11, 2015.

U LISBOA | UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS UNIVERSIDADE DE LISBOA

LASIGE Large-Scale Informatics Systems Laboratory

Navigators

Context

Developers and operations have their specificities.

http://www.enduserexperience.info/articles/share/597883/

Context

- **Developers** have been **automating software building**

- Make-based tools:
  - make, Rake, Snakemake

- Non-make-based tools:
  - Apache Ant, Buildr, Maven, MSBuild, Scons

- **Continuous integration** tools:
  - Jenkins, Magnum, Bamboo, BuildBot

Context

- Current demands:
  - **Continuous deliver** is required (up to production)
  - **Incident response** must be fast
  - **Operations** have to **automate** environment and application deployment
  - **Operations and developers** need to work **closer**

DevOps?

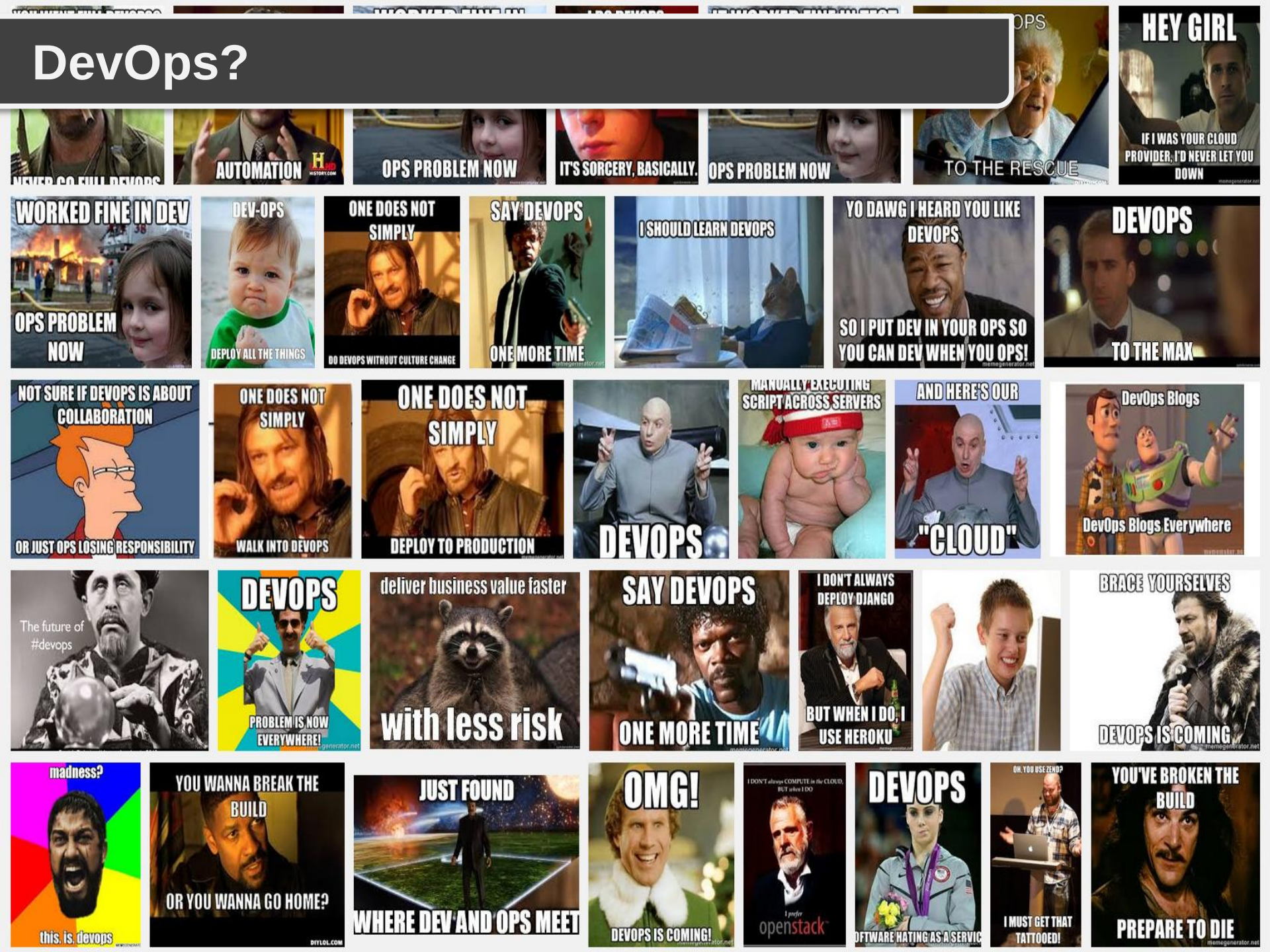
DevOps?

Developer

Operation**s**

http://blog.appdynamics.com/devops/part-dev-part-ops/

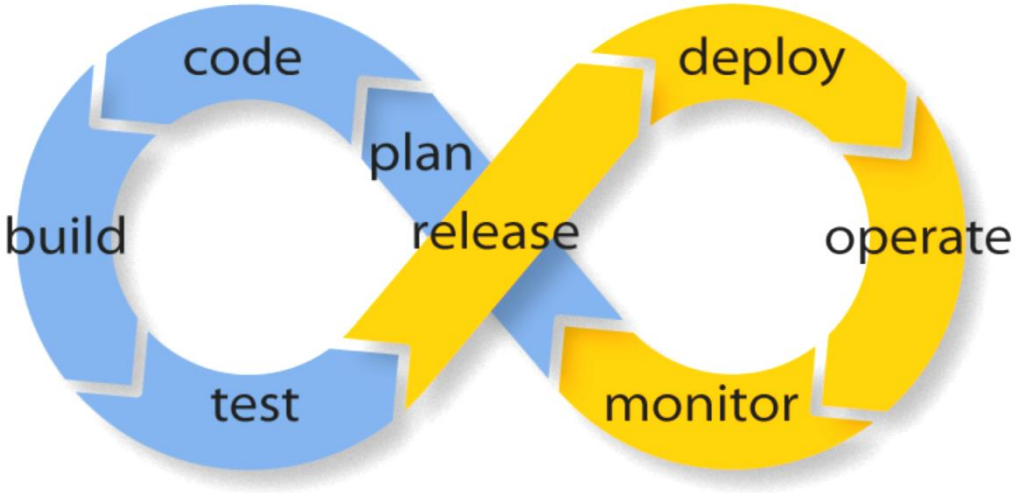
DevOps?


AUTOMATION
OPS PROBLEM NOW
IT'S SORCERY, BASICALLY.
OPS PROBLEM NOW
TO THE RESCUE
HEY GIRL
IF I WAS YOUR CLOUD PROVIDER, I'D NEVER LET YOU DOWN
WORKED FINE IN DEV
OPS PROBLEM NOW
DEV-OPS
DEPLOY ALL THE THINGS
ONE DOES NOT SIMPLY
DO DEVOPS WITHOUT CULTURE CHANGE
SAY DEVOPS
ONE MORE TIME
I SHOULD LEARN DEVOPS
YO DAWG I HEARD YOU LIKE DEVOPS
SO I PUT DEV IN YOUR OPS SO YOU CAN DEV WHEN YOU OPS!
DEVOPS
TO THE MAX
NOT SURE IF DEVOPS IS ABOUT COLLABORATION
OR JUST OPS LOSING RESPONSIBILITY
ONE DOES NOT SIMPLY
WALK INTO DEVOPS
ONE DOES NOT SIMPLY
DEPLOY TO PRODUCTION
DEVOPS
SCRIPT ACROSS SERVERS
AND HERE'S OUR
"CLOUD"
DevOps Blogs
DevOps Blogs Everywhere
The future of #devops
DEVOPS
PROBLEM IS NOW EVERYWHERE!
deliver business value faster
with less risk
SAY DEVOPS
ONE MORE TIME
I DON'T ALWAYS DEPLOY DJANGO
BUT WHEN I DO, I USE HEROKU
BRACE YOURSELVES
DEVOPS IS COMING
madness?
this. is. devops
YOU WANNA BREAK THE BUILD
OR YOU WANNA GO HOME?
DIYLOL.COM
JUST FOUND
WHERE DEV AND OPS MEET
OMG!
DEVOPS IS COMING!
I DON'T always COMPUTE in the CLOUD, BUT when I DO
I prefer openstack
DEVOPS
OFTWARE HATING AS A SERVIC
OH, YOU USE ZEND?
I MUST GET THAT TATTOOED!
YOU'VE BROKEN THE BUILD
PREPARE TO DIE

DevOps

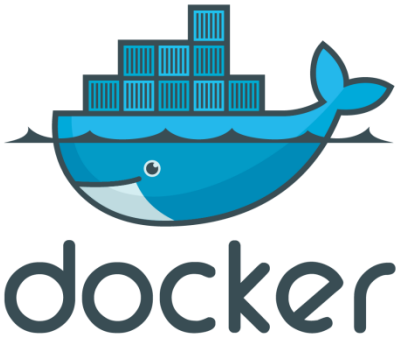
- Operations and development engineers participating **together** in the **entire service lifecycle**

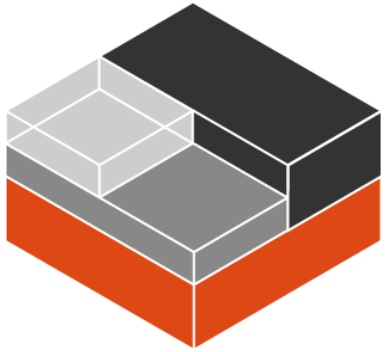
- From **design** through the **development** process to production **support**



- **Operations** staff making use many of the **same techniques as developers** for their systems work

(Some) Emerging DevOps solutions



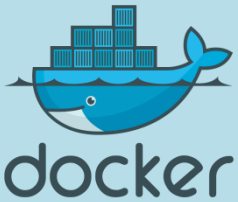


(Unofficial) Classification

- Configuration management:
  - Puppet, Chef, Ansible, SaltStack …
  - Describe environment + automatic configuration in execution time
- Containerization tools:
  - LXC, OpenVZ, Docker, Kubernetes, DxEnterprise …
  - Basic environment + libs and apps in a lightweight image
- Virtualization platforms:
  - Xen, KVM, VirtualBox, VMWare, Vagrant …
  - VMM + Guest OS image with the environment, libs and apps
- Infrastructure-as-a-Service:
  - AWS, Rackspace, Azure, OpenStack …
  - On-demand resources + VM image(s)

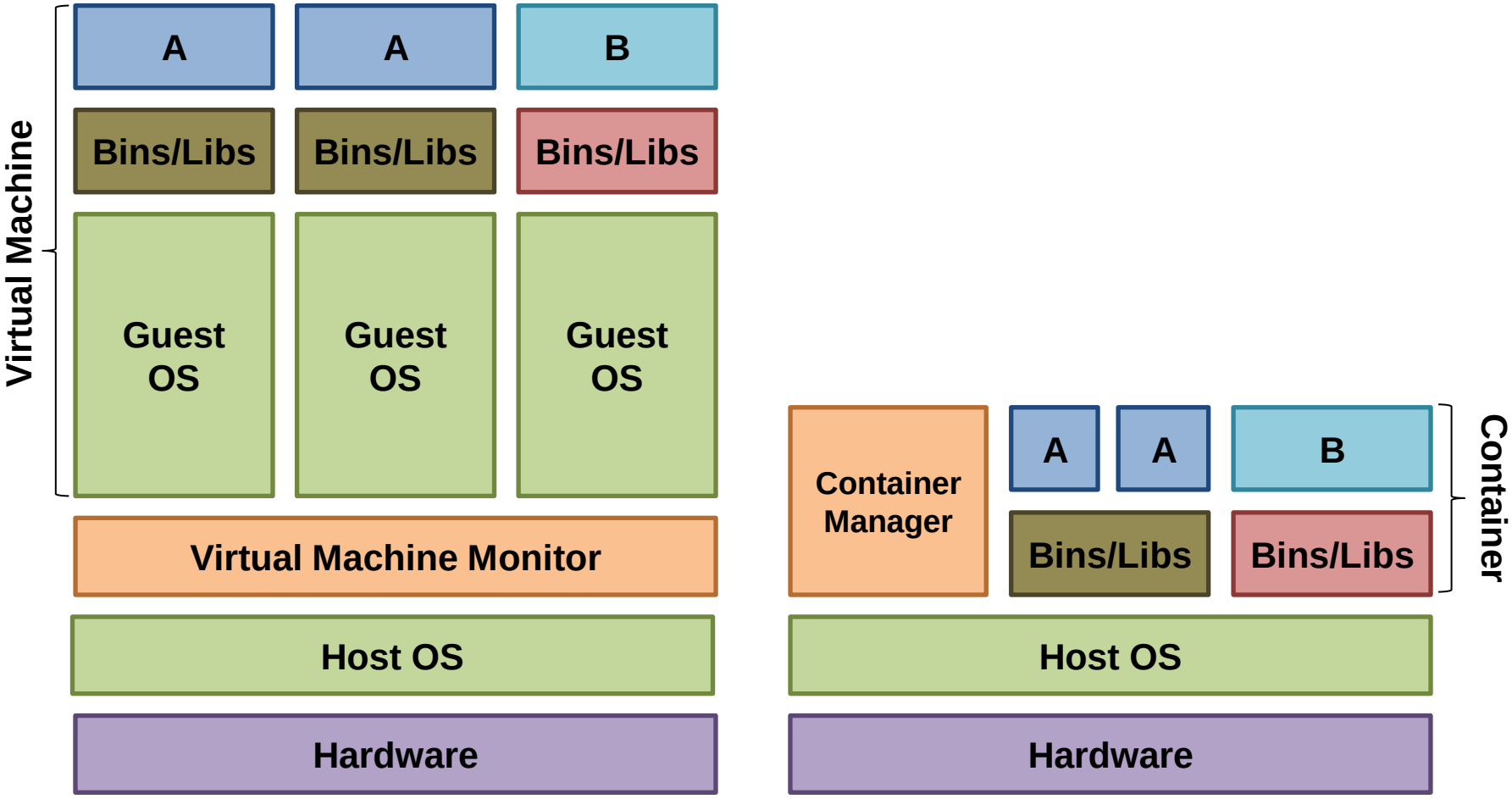
(Unofficial) Classification

- Configuration management:
  - Puppet, Chef, Ansible, SaltStack …
  - Describe environment + automatic configuration in execution time
- Containerization tools:
  - LXC, OpenVZ, **Docker**, Kubernetes, DxEnterprise …
  - Basic environment + libs and apps in a lightweight image



- Virtualization platforms:
  - Xen, KVM, VirtualBox, VMWare, Vagrant …
  - VMM + Guest OS image with the environment, libs and apps
- Infrastructure-as-a-Service:
  - AWS, Rackspace, Azure, OpenStack …
  - On-demand resources + VM image(s)

Virtual machines vs. containers

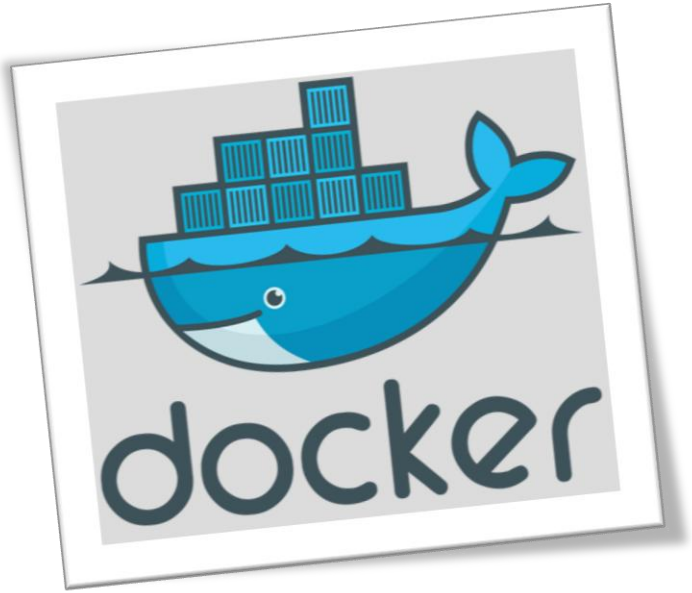


What is Docker?

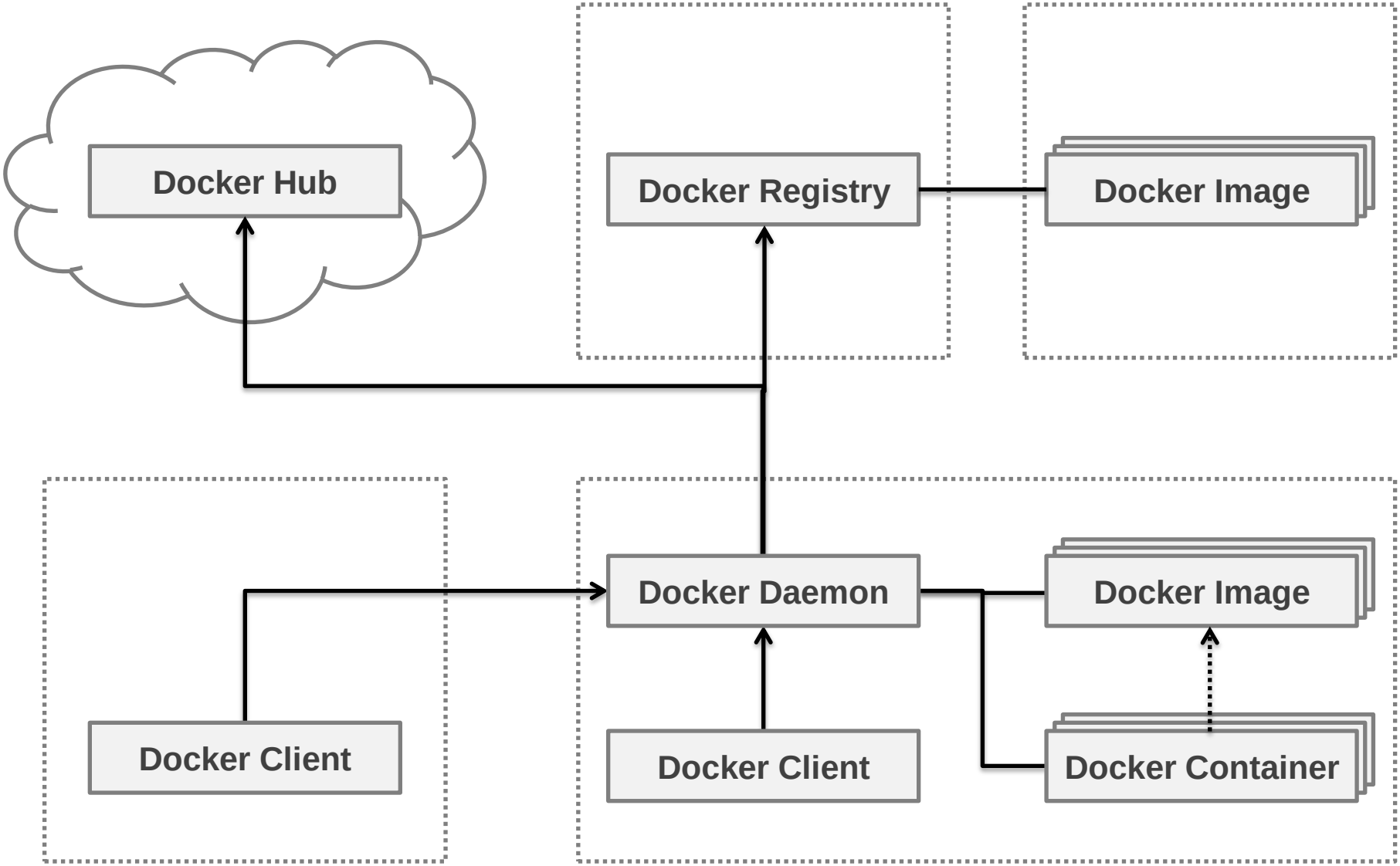
An open **platform** for developers and sysadmins to **build**, **ship**, and **run** distributed applications

IT can ship faster and run the same app, **unchanged**, on laptops, data center VMs, and any cloud

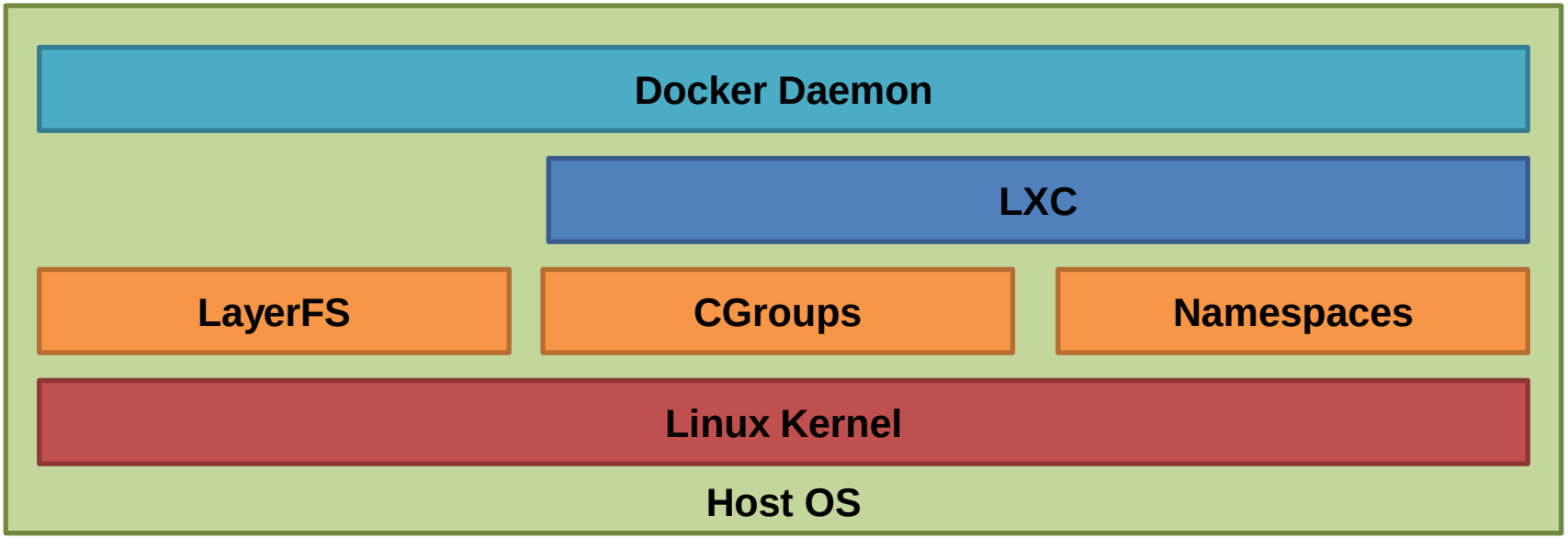
- Docker **Client**
  - Interaction tool (e.g., GUI, CLI …)
- Docker **Daemon**
  - Lightweight runtime and packaging tool
- Docker **Registry**
  - Private catalog of Docker images
- Docker **Hub**
  - Cloud service for sharing applications and automating workflows
- Docker **Image**
  - The libs+apps image volume
- Docker **Container**
  - A running Docker image



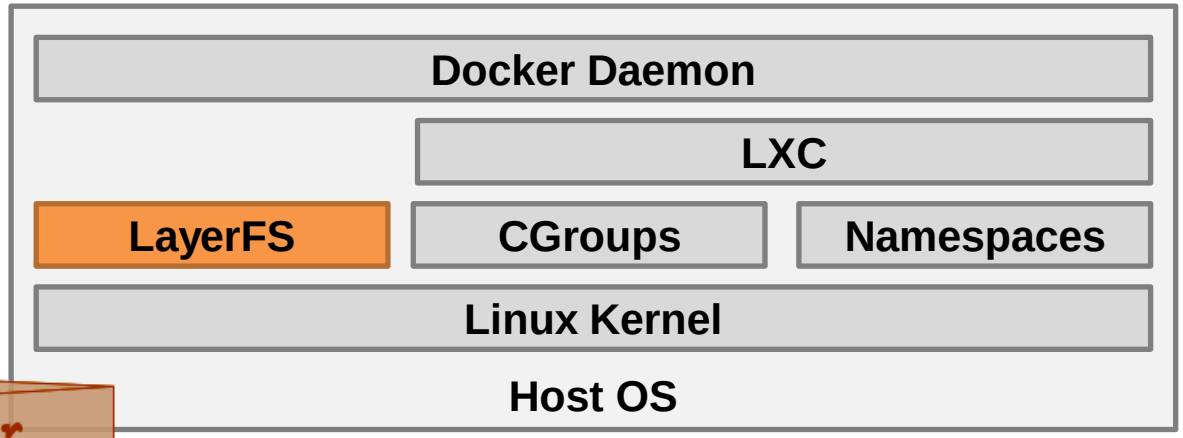
Docker infrastructure

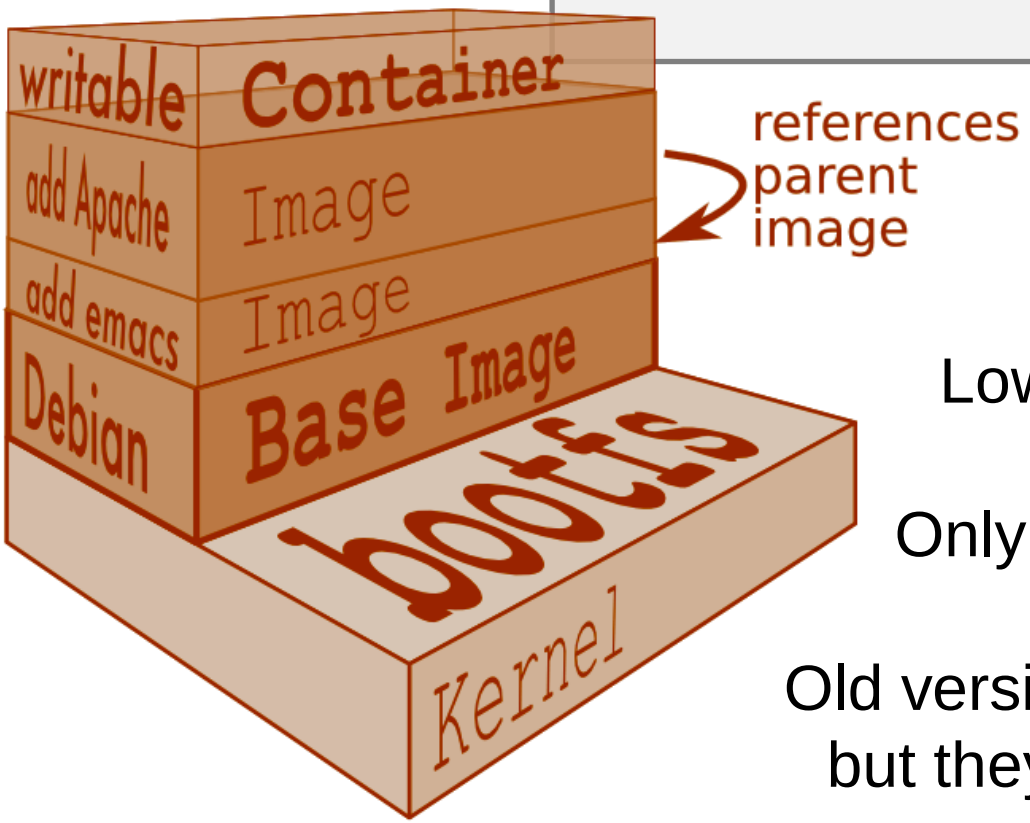


Docker architecture



Docker architecture - LayerFS





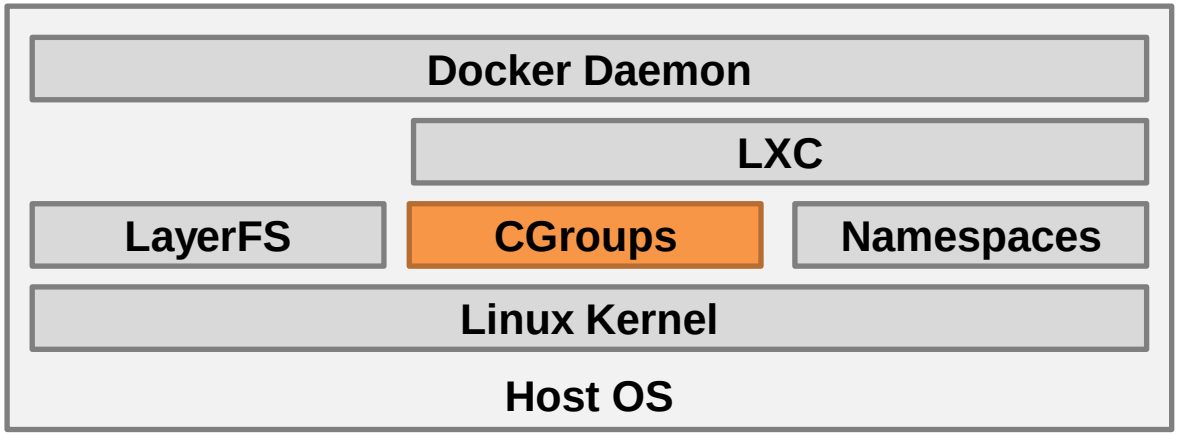
Use union mounts

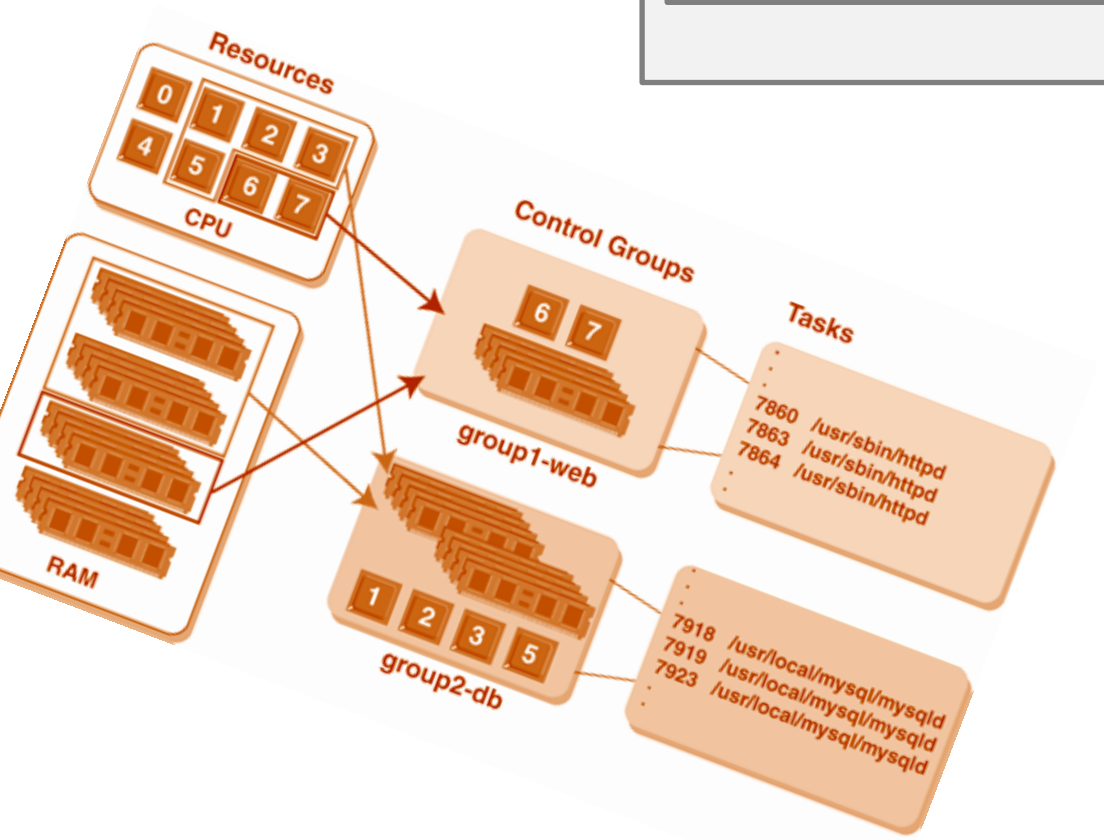
Lower layers are read-only

Only the top layer is writable

Old versions of files are invisible
but they are in the lower layers

Docker architecture - CGroups





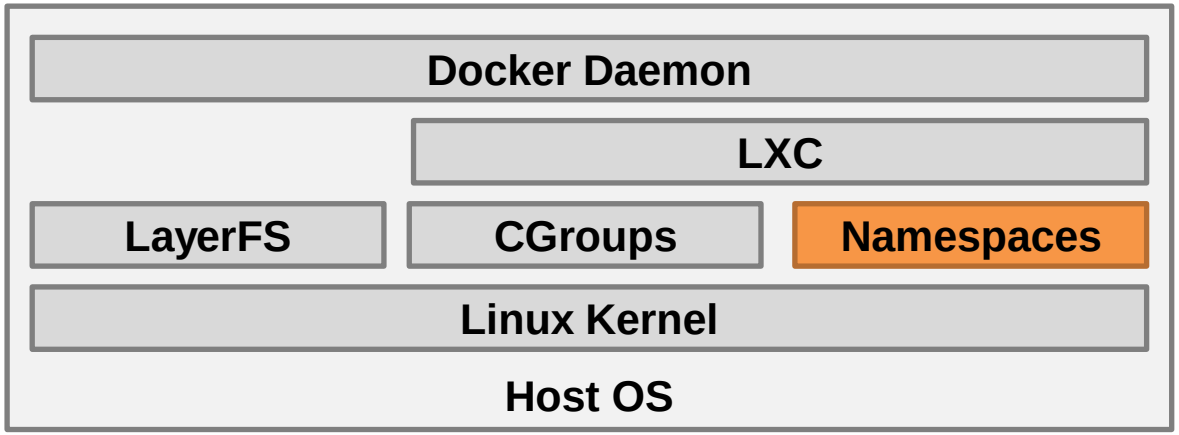
Resource management

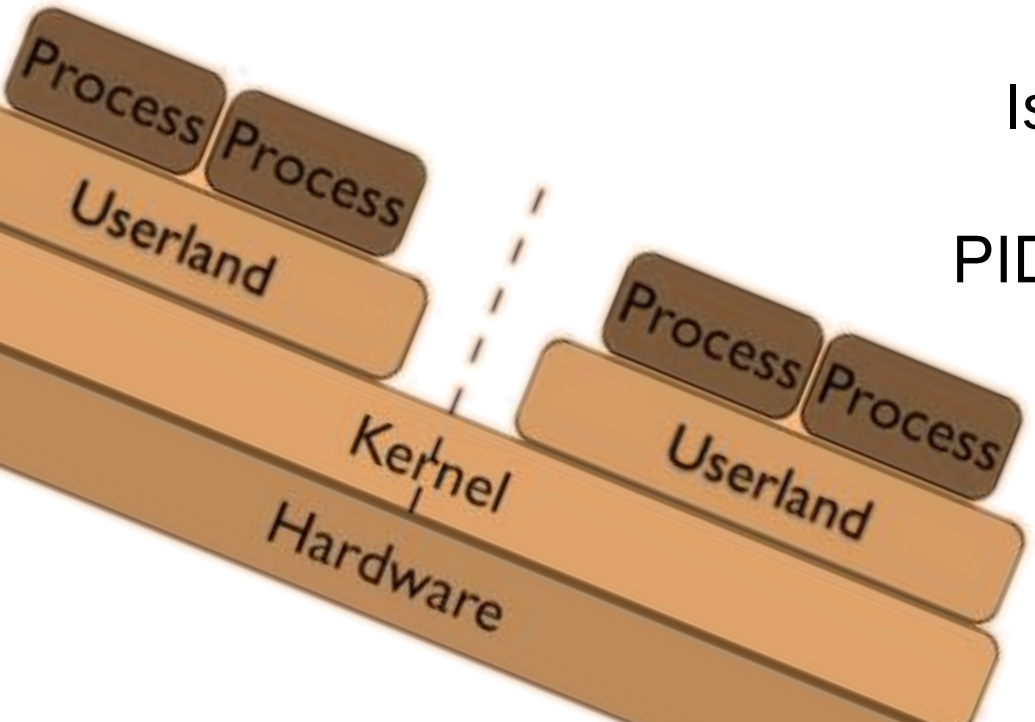
Groups of processes

Allocate resources for each group

CPU, RAM, Network, and Disk resources

Docker architecture - Namespaces



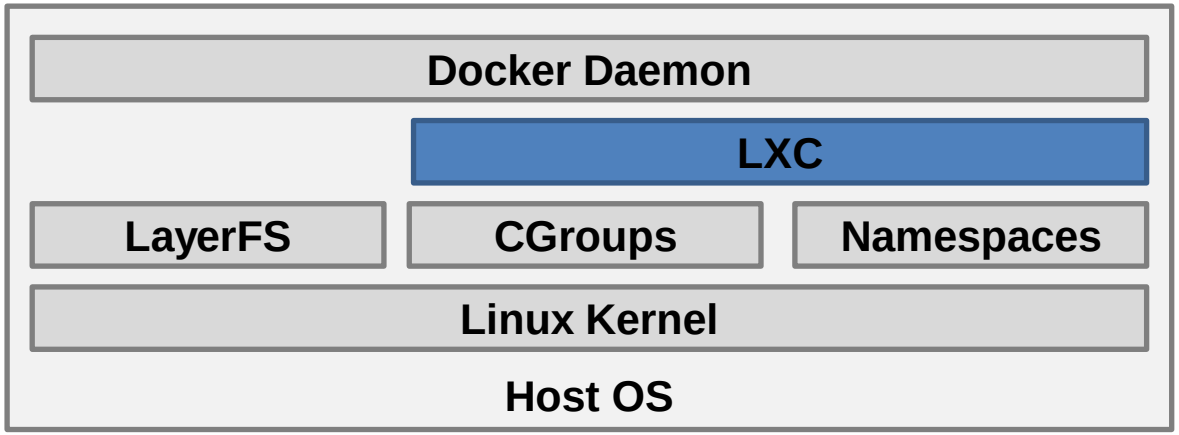


Isolate groups of processes

PIDs, NICs, mounts, users …

One group does not see the resources from others

Docker architecture - LXC



OS-level virtualization

Provide isolated environment for applications

LXC is one of Docker's execution drivers

Docker and the SuperCloud project

- Efficiency
- Performance
- Isolation
- Security
- Live migration and state transfer
- Diversity
- Nested virtualization

Efficiency

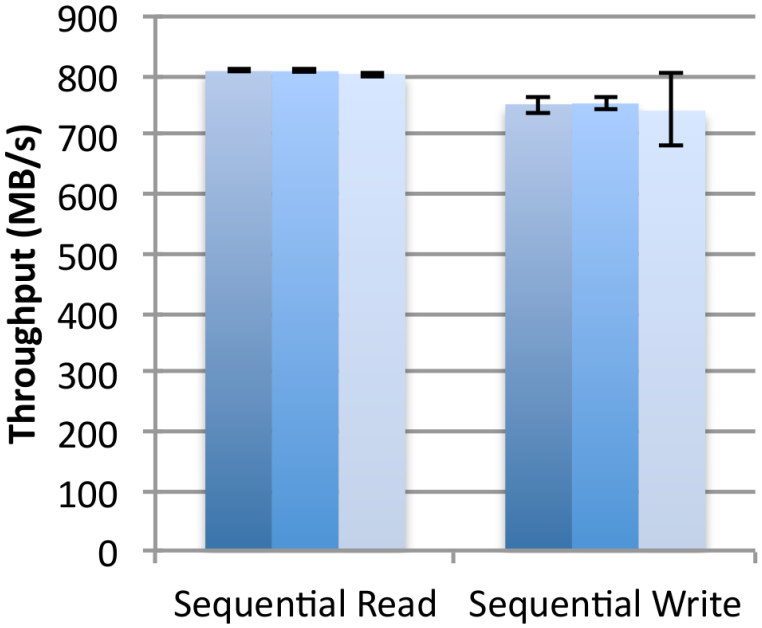
Property	Virtual Machine	Container
Unit size	Few GB	Few MB
Units per host	1–100	1–1000
Start time	30–45 sec	< 50 ms
Stop time	5–10 sec	< 50 ms

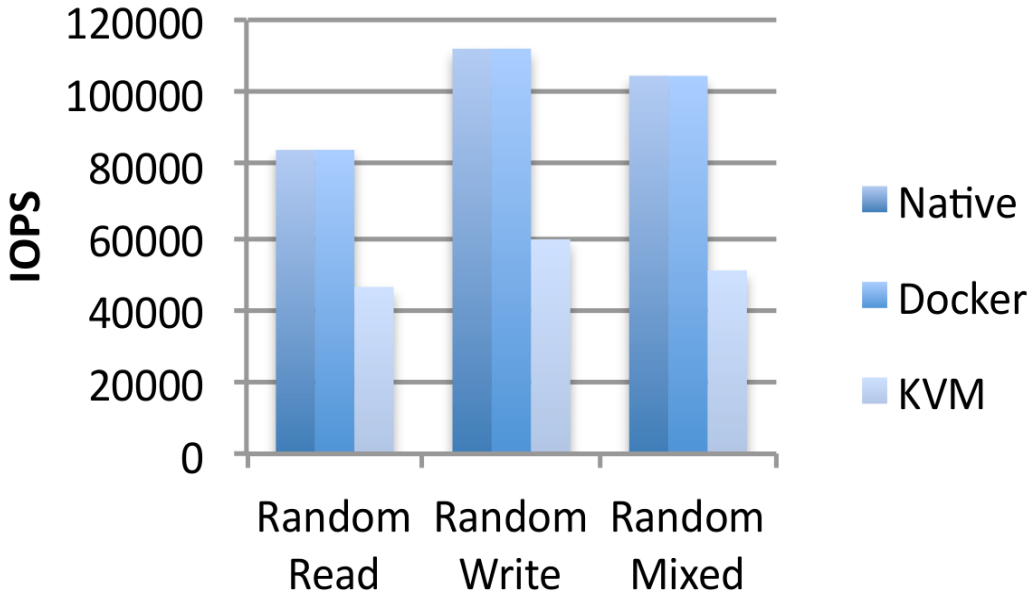
- The same Docker image could be efficiently deployed on anything from a fraction of a core to an entire machine

Performance

Felter, Wes, et al. "**An Updated Performance Comparison of Virtual Machines and Linux Containers**." IBM Tech Report (2014).

Sequential/random read/writes

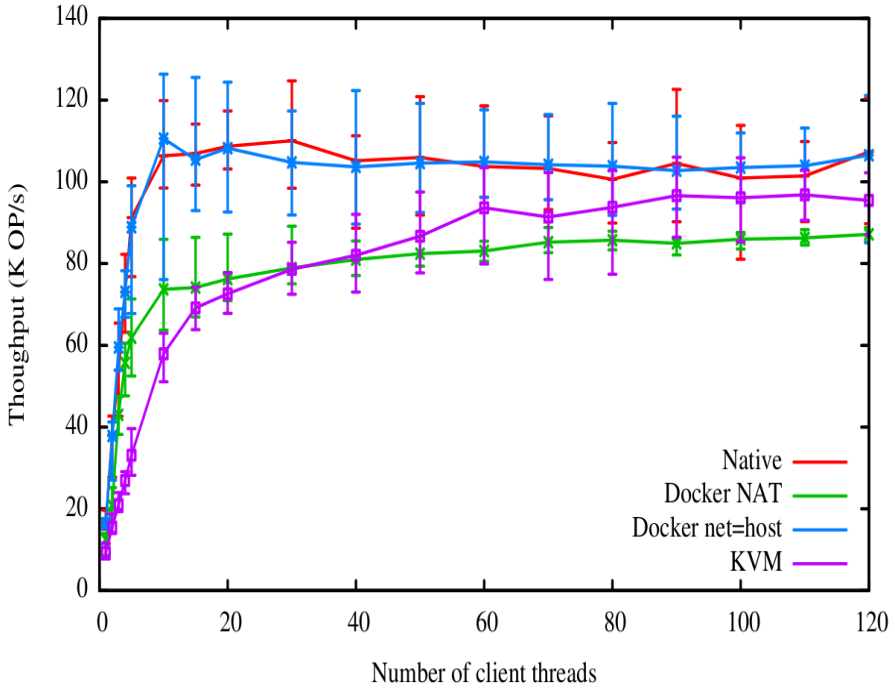


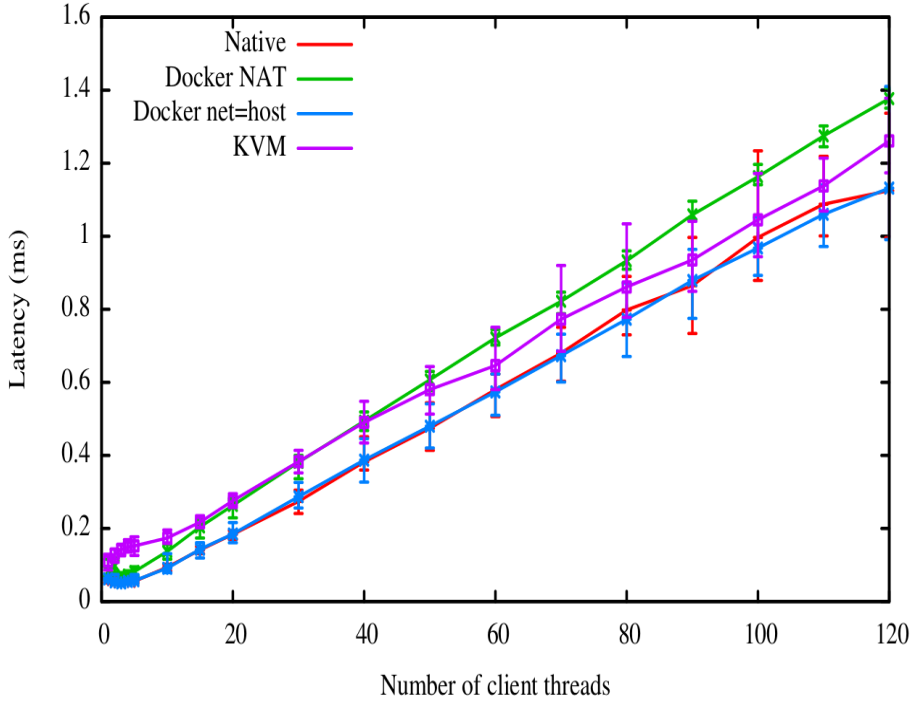


Performance

Felter, Wes, et al. "**An Updated Performance Comparison of Virtual Machines and Linux Containers**." IBM Tech Report (2014).

Throughput and latency in Redis NoSQL





Performance

Felter, Wes, et al. "**An Updated Performance Comparison of Virtual Machines and Linux Containers**." IBM Tech Report (2014).

- Docker equals or exceeds KVM performance in every case they tested
- Both KVM and Docker introduce negligible overhead for CPU and memory performance
- For I/O-intensive workloads, both forms of virtualization should be used carefully
- Deploying containers inside VMs imposes the performance overheads of VMs while giving **no** benefit compared to deploying containers directly on non-virtualized Linux

Isolation

- Docker uses for example LXC as execution driver
- LXC uses Kernel’s namespaces and cgroups
- Different users can run different containers in the same host
- Apps from one container do not see or affect resources from other containers

Security

- Security at **namespace** level = **LXC**
- Security at **resource** accounting and limiting = **cgroups**
- **Increased attack surface** in Docker daemon, which requires root privileges
- Docker daemon can **expose** a **REST API** over HTTP to remote clients
- Foreseen **secure improvements**:
  - Map root of a container to a non-root user in the host
  - Run unprivileged daemon
- Running a VM inside a container can create an extra layer of security since an attacker who can exploit QEMU would still be inside the container

Live migration and state transfer

- Docker **doesn’t** support **live migration**
- **Flocker** is a project trying to migrate active containers within a cluster (still with downtime)
- **CRIU** (Checkpoint/Restore In Userspace) is a project trying to perform live migration of LXC containers
- Three options from CRIU:
  - Stop old, copy, start new, kill old
  - Pre-dump old, stop old, copy diff, start new, kill old
  - Disk-less migration (transfer pages instead of disks)
- But it is not straightforward, and still is unstable

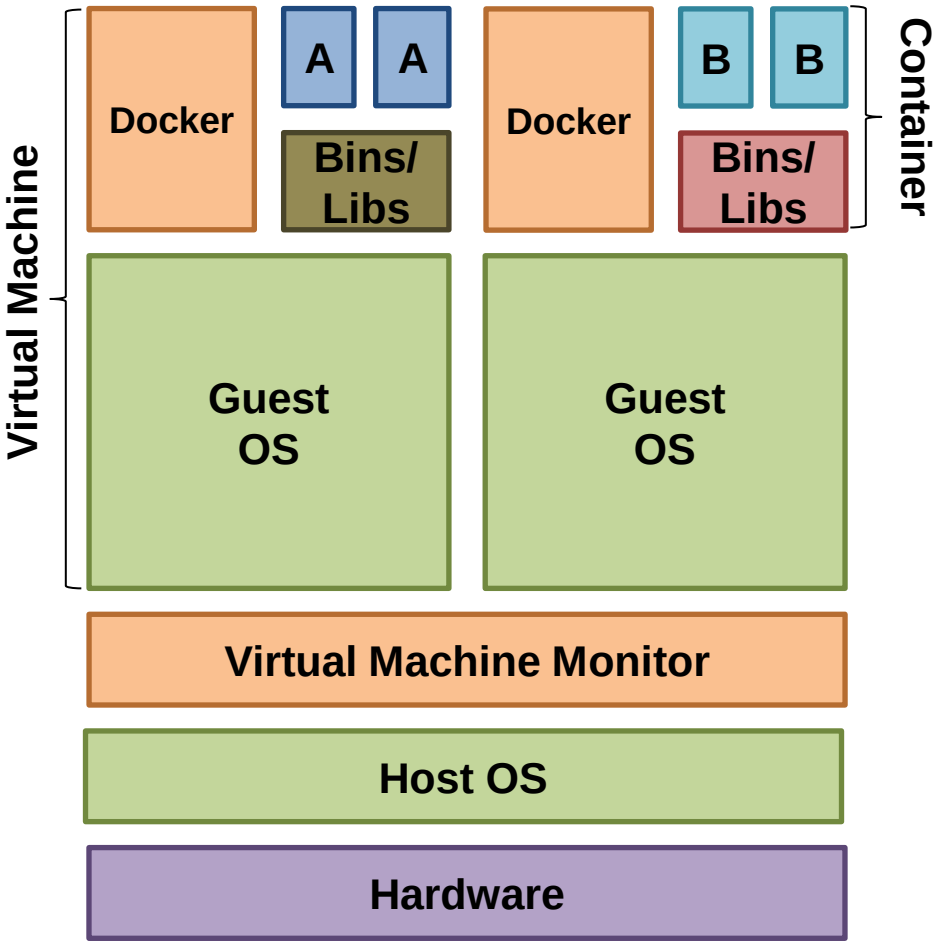
Diversity

- Multiple platforms supporting Docker:
  - OS-X and Windows **don't** support Docker containers by default
  - **Boot2Docker** is a "tiny" VirtualBox VM to run Docker in OS-X and Windows

- Docker supporting multiple platforms:
  - Docker containers run **only Linux** containers (Ubuntu, Debian, Fedora,RedHat …)
  - It means, OS-X or Windows apps don't run on Docker
  - DxEnterprise is a competitor for running Windows apps

Nested virtualization

Nested virtualization – Scenario 01



Docking Dockers

Docker within a VM

Original container size

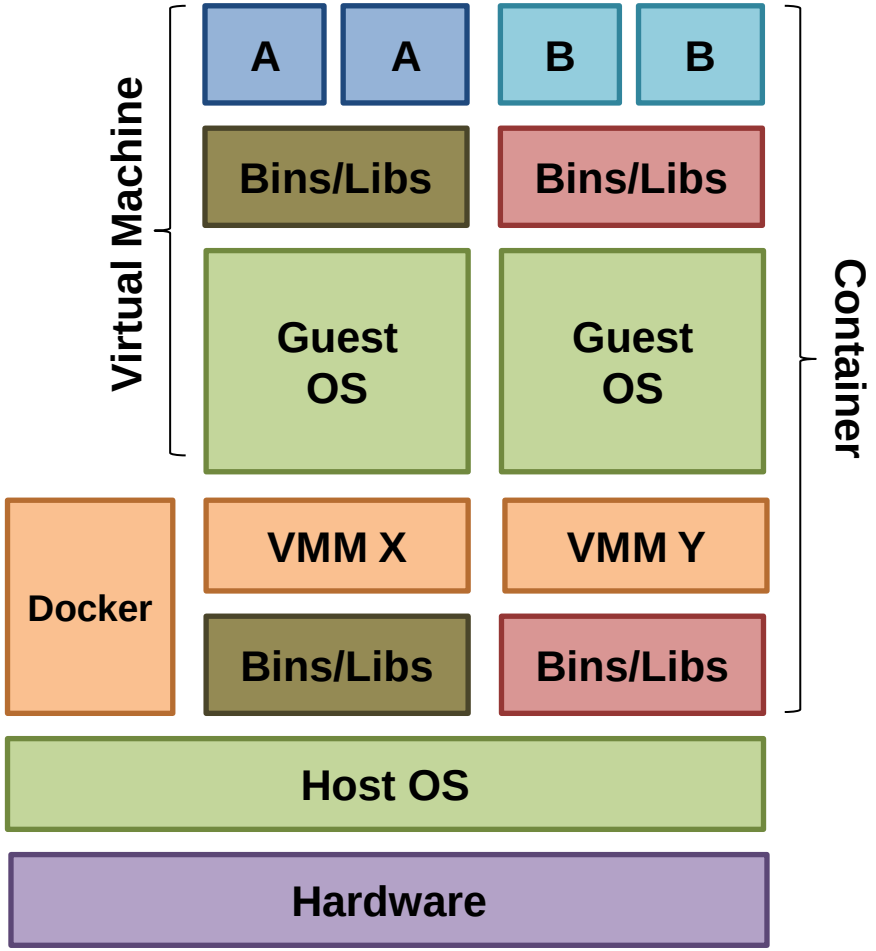
VMs are used to grow

Works in current clouds!

Only Linux containers

No diversity

Nested virtualization – Scenario 2



Docking VMMs

VMM within a container

Container size grows (a lot!)

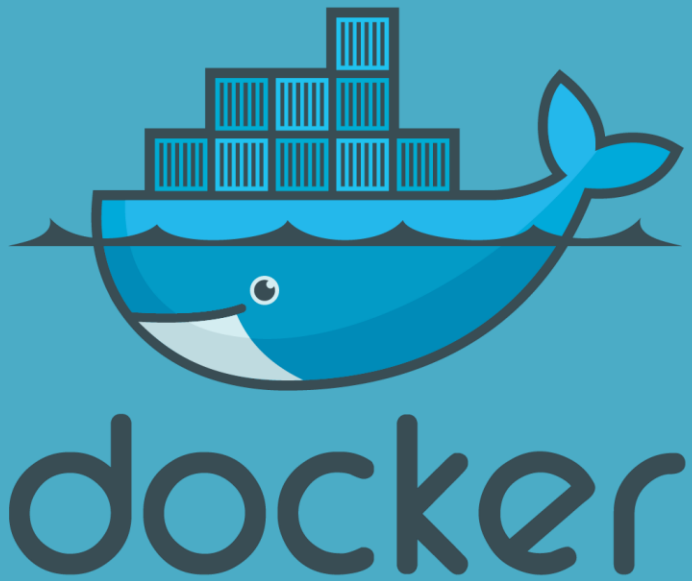
Doesn't work in clouds!

OS and apps diversity

Secure VMMs (within the namespace)

Conclusions

- Both VMs and containers are mature technology
- Containers do not replace virtual machines
- Docker does not support live migration
  (Flocker does, with downtime)
- Docker within VMs (scenario 01) seems to be more appropriate for SuperCloud
- Docker probably will cause less overhead in nested virtualization
- However, we probably would need to code or hack Docker to have all desired features



Thank you!

LISBOA | UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS UNIVERSIDADE DE LISBOA

LASIGE Large-Scale Informatics Systems Laboratory

Navigators