

docker

A DOCKER JOURNEY


CHICHI

SINCE IT STARTED IN MARCH 2013...

- >200,000 pulls
- >23,500 github stars
- >1000 significant contributors
- >200 projects built on top of docker


THE CHALLENGE


Multiplicity of Stacks

 Static website
nginx 1.5 + modsecurity + openssl + bootstrap 2


 User DB
postgresql + pgv8 + v8

 Queue
Redis + redis-sentinel

 Analytics DB
hadoop + hive + thrift + OpenJDK

 Background workers
Python 3.0 + celery + pyredis + libcurl + ffmpeg + libopencv + nodejs + phantomjs

 Web frontend
Ruby + Rails + sass + Unicorn

 API endpoint
Python 2.7 + Flask + pyredis + celery + pycopg + postgresql-client

Do services and apps
interact
appropriately?



Multiplicity of
hardware
environments

 Development VM

 QA server

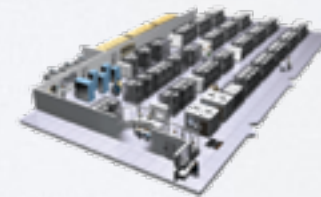
Customer Data Center



Public Cloud

Disaster recovery

Production Servers



Production Cluster









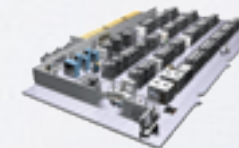
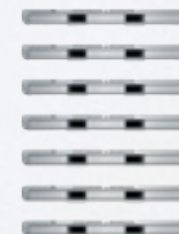
Contributor's laptop



Can I migrate
smoothly and
quickly?

THE MATRIX FROM HELL

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

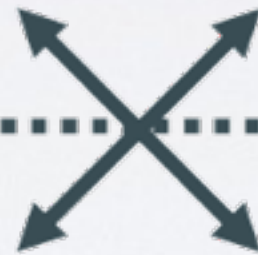


CARGO TRANSPORT PRE-1960

Multiplicity of Goods



Do I worry about
how goods interact
(e.g. coffee beans
next to spices)



Multiplicity of
methods for
transporting/storing



Can I transport quickly
and smoothly
(e.g. from boat to train
to truck)

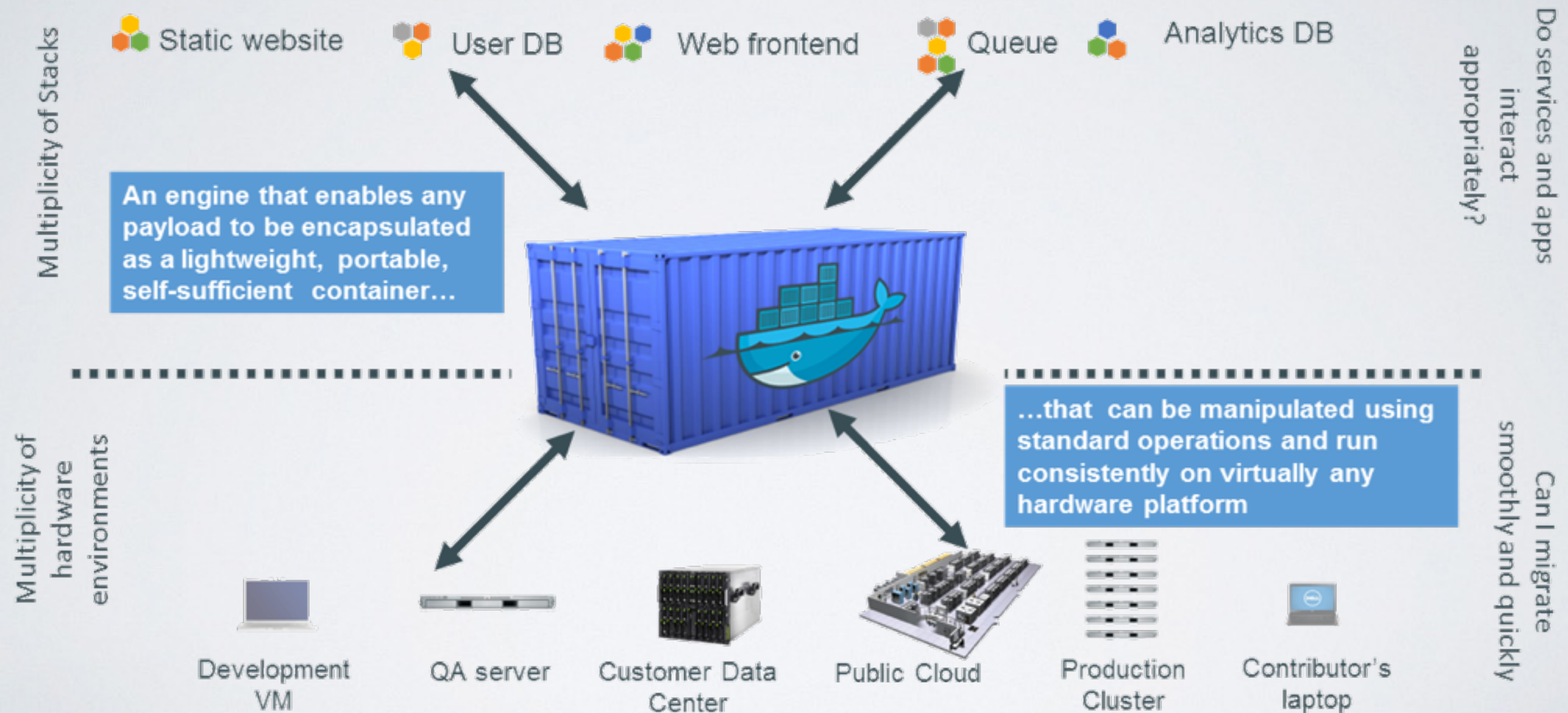
ALSO A MATRIX FROM HELL

	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
							

SOL:INTERMODAL SHIPPING CONTAINER



DOCKER IS A CONTAINER SYSTEM FOR CODE



DOCKER ELIMINATES THE MATRIX FROM HELL

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



WHAT DEVELOPERS WANT?

- Build once and run everywhere.
- A clean, safe, hygienic, portable runtime environment for your app.
- No worries about missing dependencies.
- Reduce/Eliminate concerns about compatibility

WHAT ADMINISTRATORS WANT?

- Configure once... run anything
- Make the entire lifecycle more efficient, consistent, and repeatable
- Increase the quality of code produced by developers.
- Eliminate inconsistencies between development, test, production, and customer environments.
- Because the containers are so lightweight, address significant performance, costs, deployment, and portability issues normally associated with VMs.

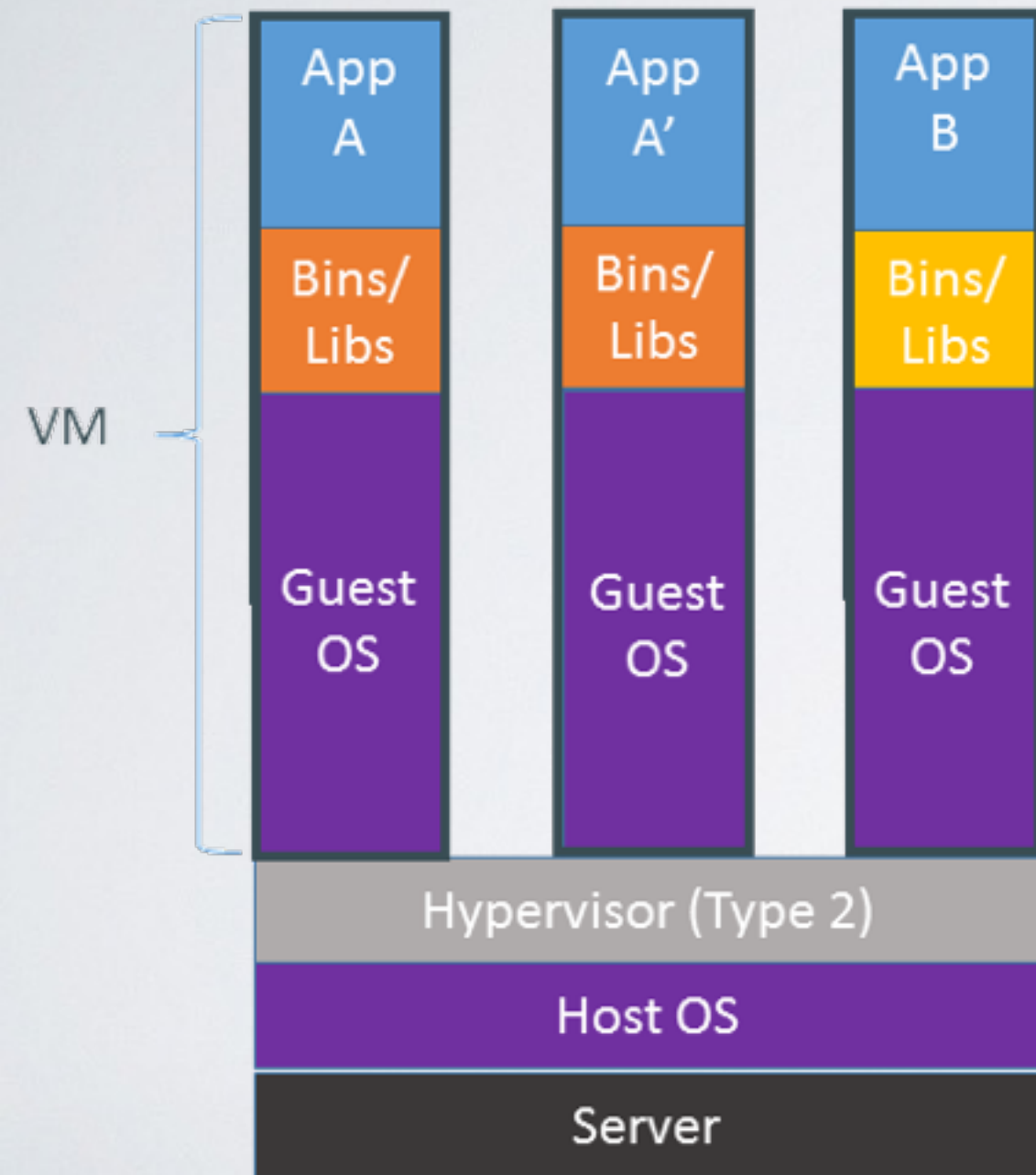
WHAT IT'S LIKE?

- High level: a lightweight VM
 - Own process space
 - Own network interface
 - Can run stuff as root
 - Can have its own /sbin/init (different from host)
 - <<machine container>>

WHAT IT'S LIKE?

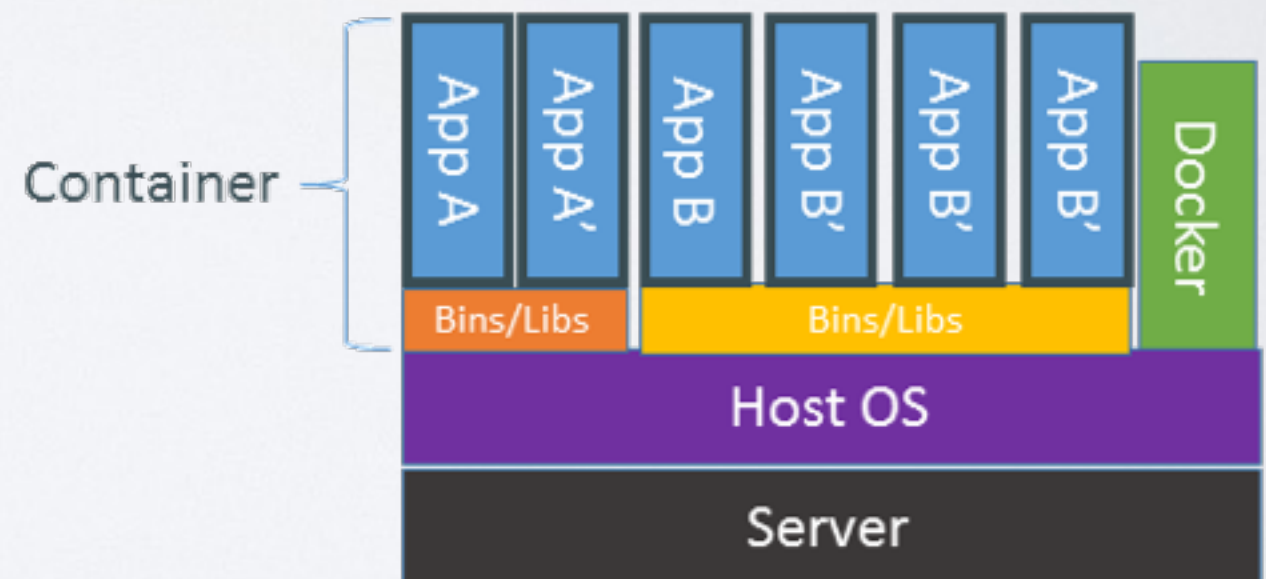
- Low level:chroot on steroids
 - Can also not have its own /sbin/init
 - Container = isolated processes
 - Share kernel with host
 - <<application container>>

VM & CONTAINER



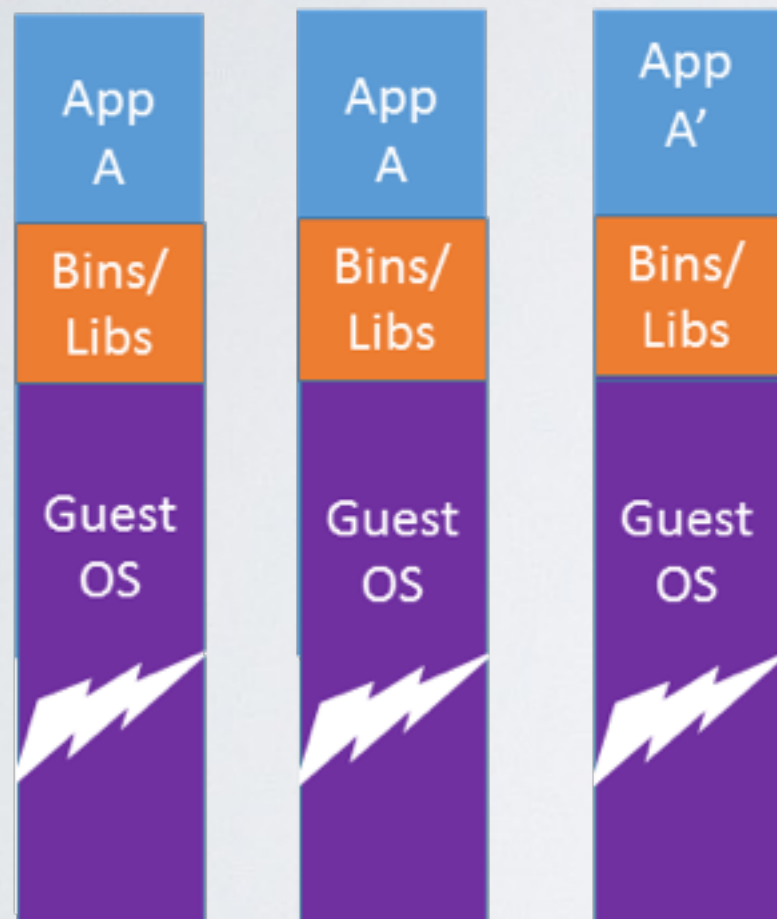
Containers are isolated, but share OS and, where appropriate, bins/libraries

...result is significantly faster deployment, much less overhead, easier migration, faster restart



WHY SO LIGHTWEIGHT?

VMs



VMs

Every app, every copy of an app, and every slight modification of the app requires a new virtual server

Containers

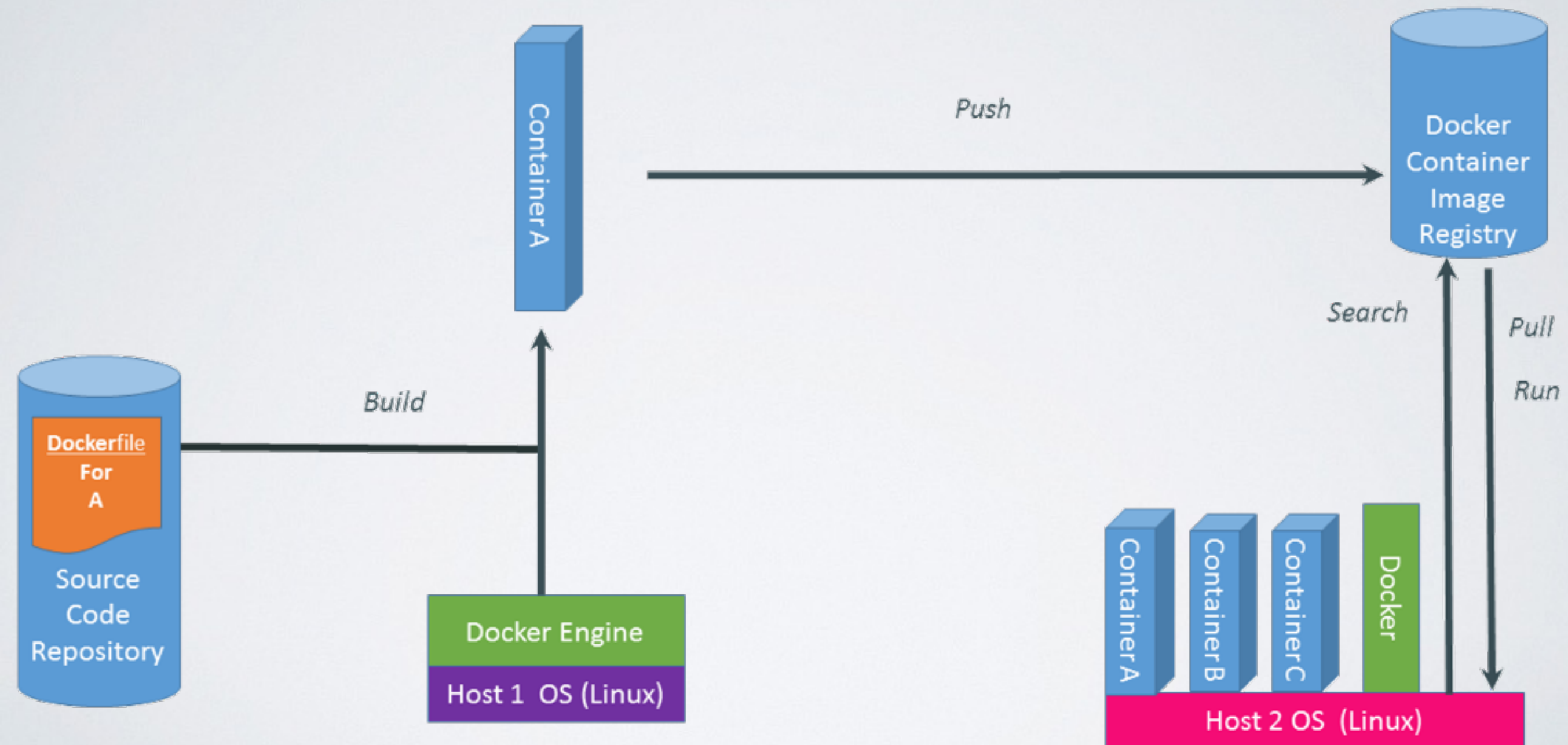


Original App
(No OS to take up space, resources, or require restart)

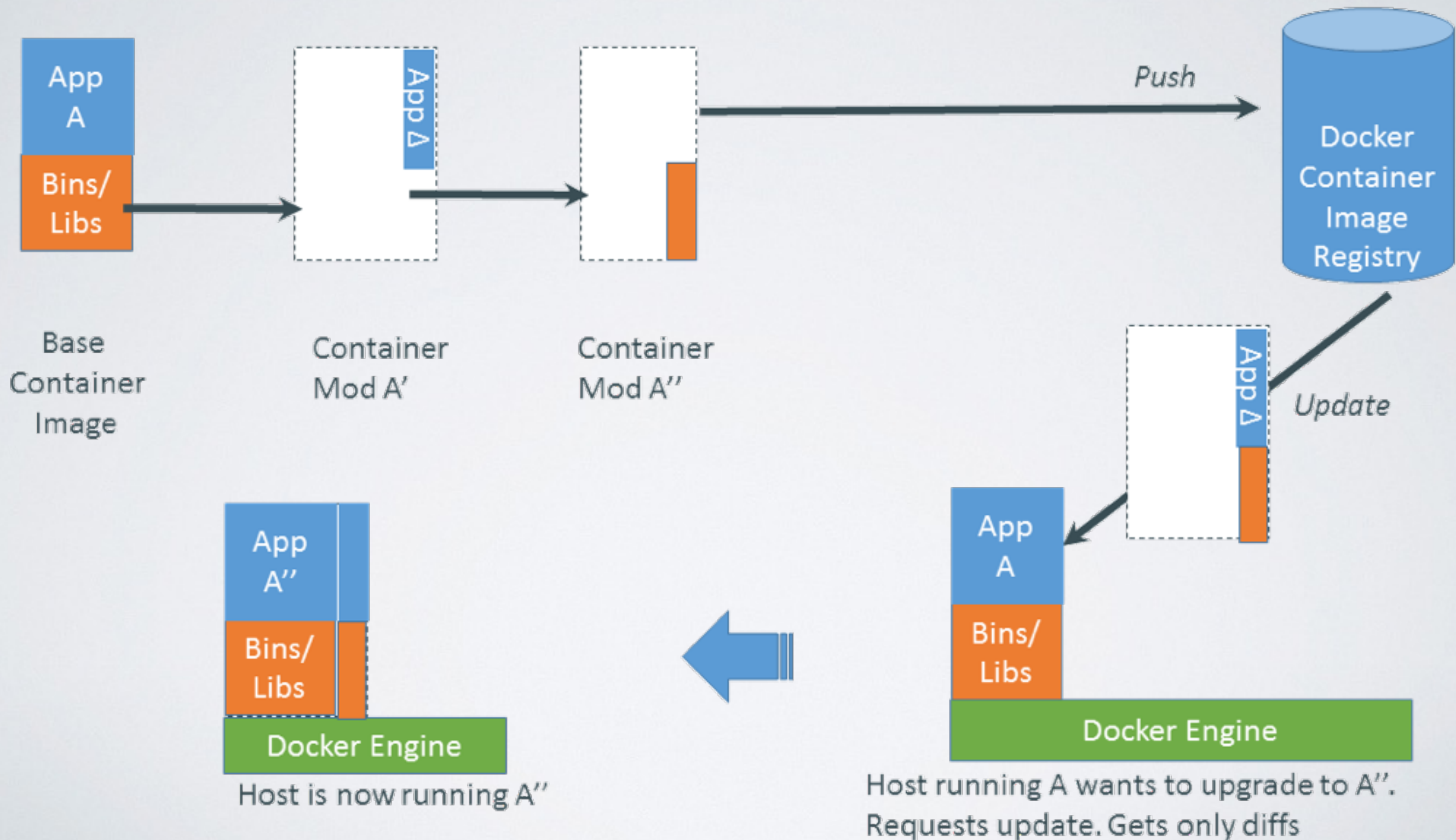
Copy of App
No OS. Can Share bins/libs

Modified App
Copy on write capabilities allow us to only save the diffs Between container A and container A'

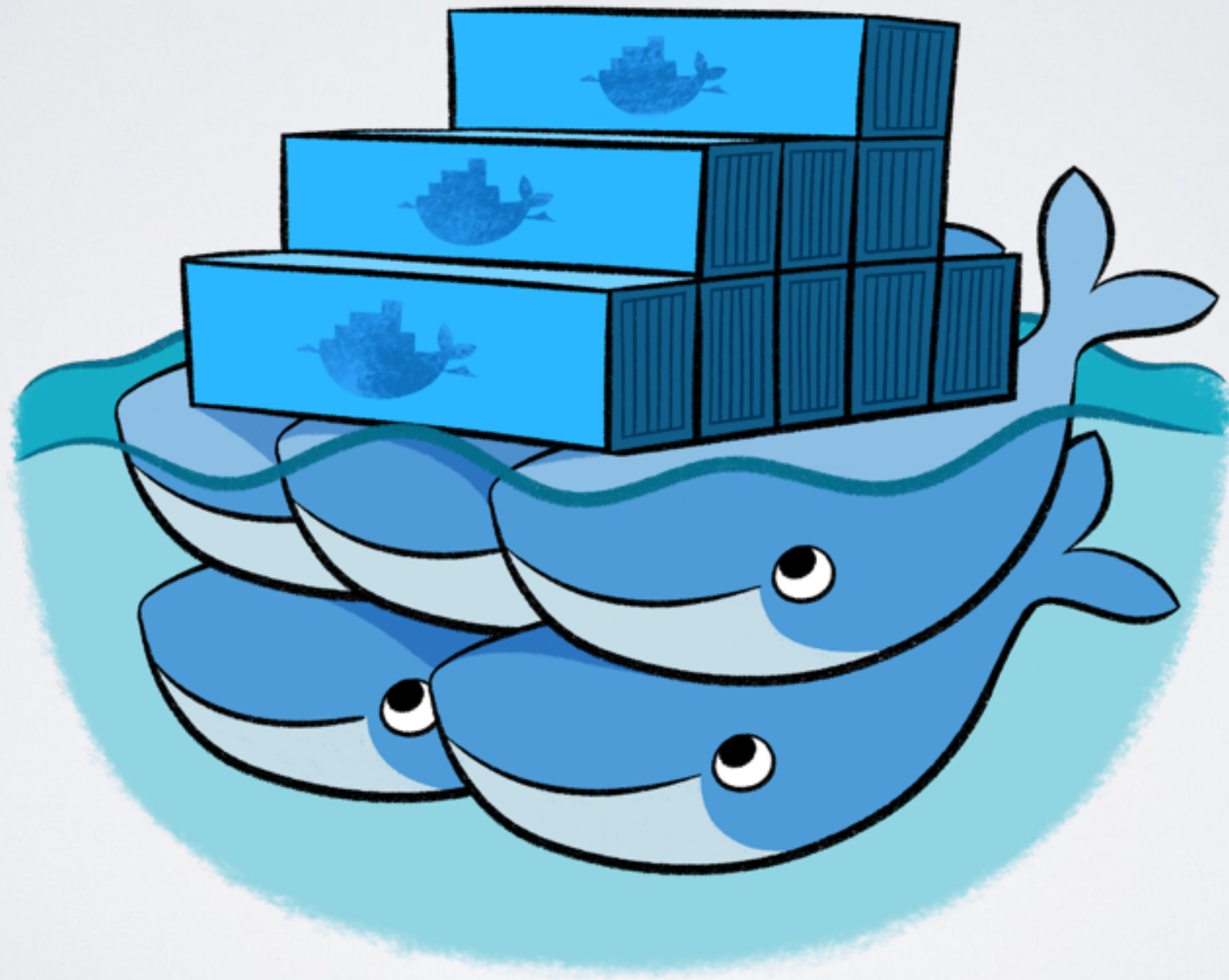
BASICS OF A DOCKER SYSTEM



CHANGES AND UPDATES



DEMO



GET YOUR HANDS DIRTY

- Write a dockerfile for a simple node server
- Build it & Run it
- Connection to a MySQL Container
- Push to hub.docker.com