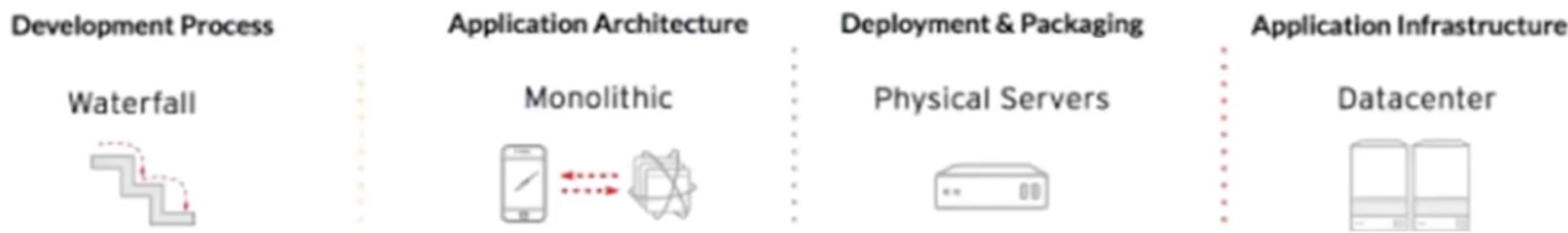


# Docker

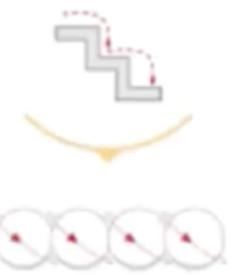
# OK... Why the buzz around containers?

- The software industry has changed.
- Before:
  - monolithic applications
  - long development cycles
  - slowly scaling up
- Now:
  - decoupled services
  - fast, iterative improvements
  - quickly scaling out



### Development Process

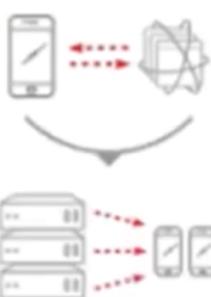
Waterfall



Agile

### Application Architecture

Monolithic



N-Tier



### Deployment & Packaging

Physical Servers



Virtual Servers



### Application Infrastructure

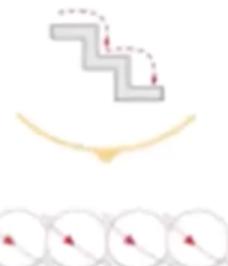
Datacenter



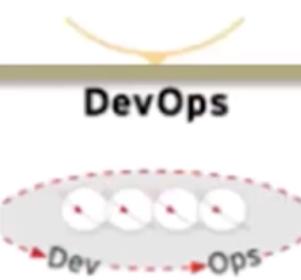
Hosted

## Development Process

Waterfall

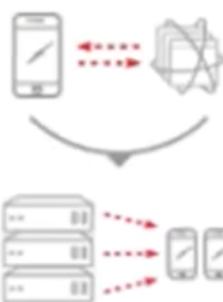


Agile



## Application Architecture

Monolithic



N-Tier



## Deployment & Packaging

Physical Servers



Virtual Servers



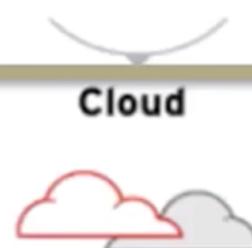
Containers

## Application Infrastructure

Datacenter



Hosted



Cloud

**TIME TO  
MARKET**

Slow

## Agile

- Automate testing
- Iterative process
- Flat structure of small teams
- Focus on time to market

## Agile + DevOps

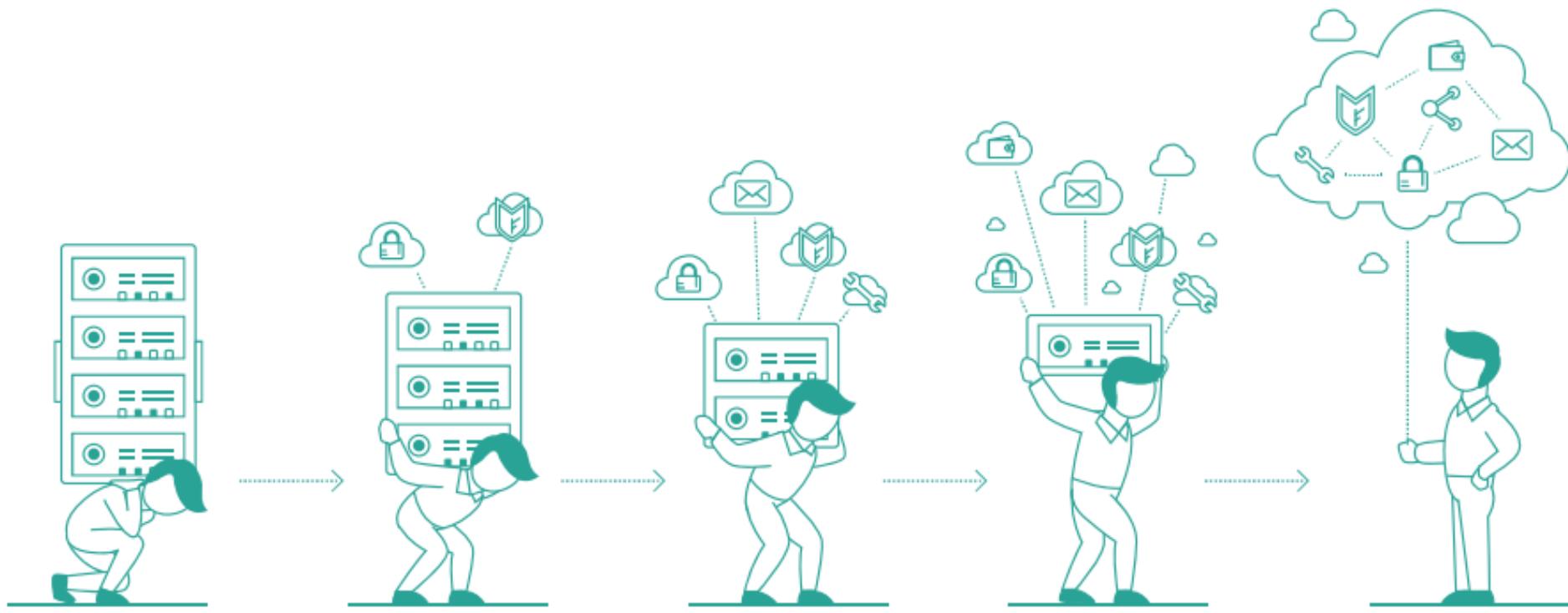
- Automate governance
- Microservices
- Product-centric teams
- Value stream management

## Waterfall

- Sequential/linear stages
- Bureaucratic model
- Focus on stability & documentation

## DevOps

- Automate everything
- Continuous process
- Integrate Dev & Ops teams
- Focus on quality



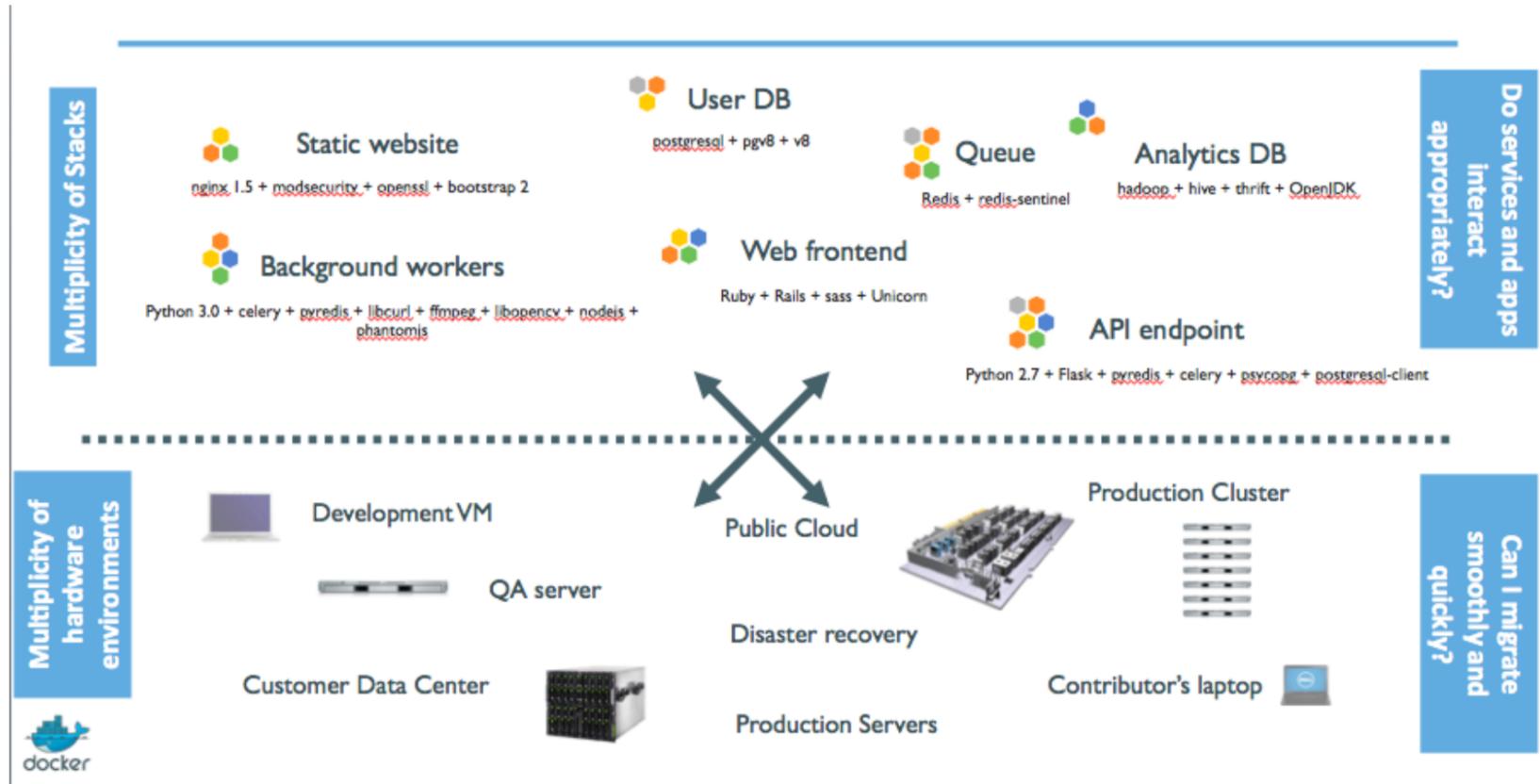
## About Docker

# Deployment becomes very complex

- Many different stacks.
- Many different targets.

## About Docker

# The deployment problem



## About Docker

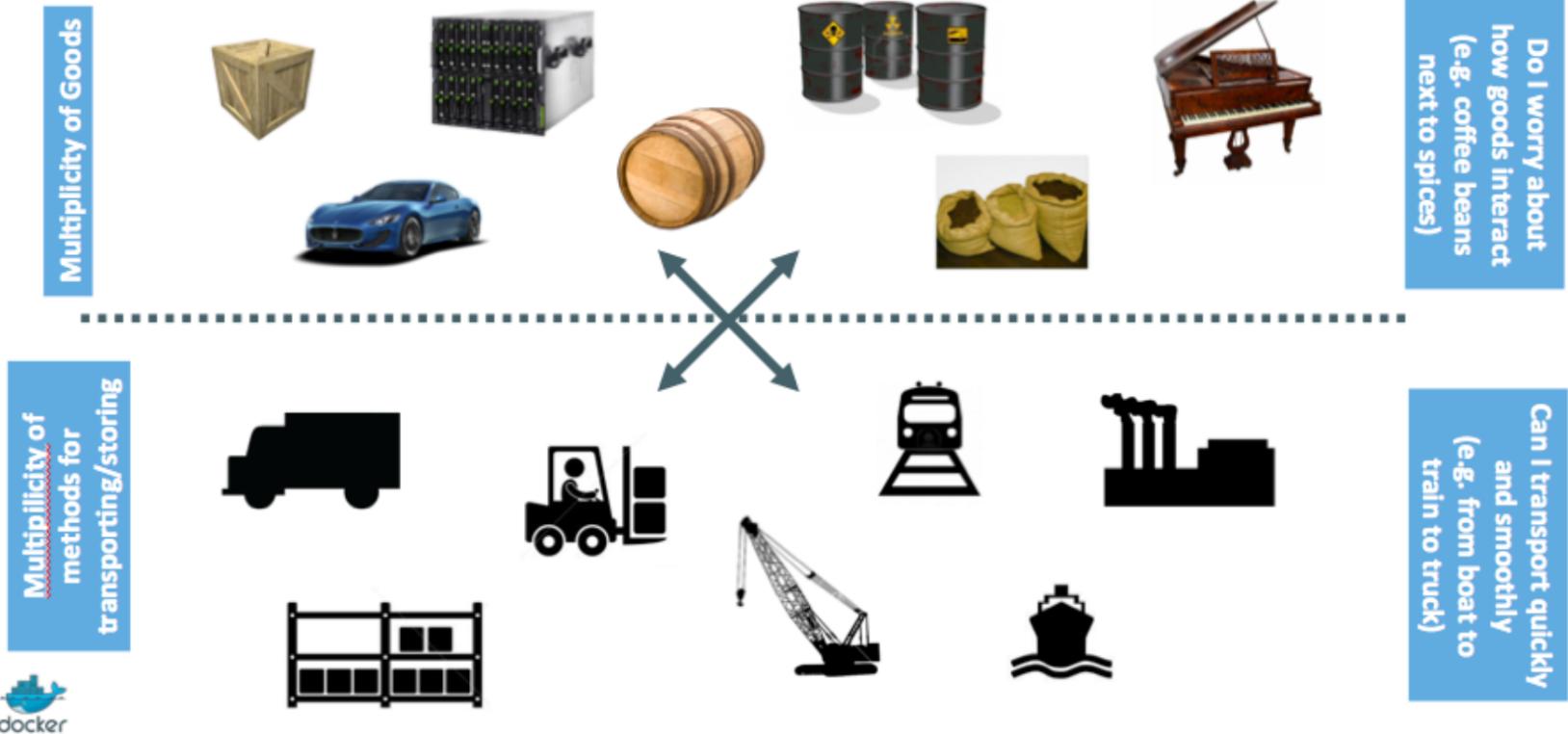
# The Matrix from Hell

 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
							



## About Docker

# An inspiration and some ancient history!



## About Docker

# Intermodal shipping containers



## About Docker

# This spawned a Shipping Container Ecosystem!

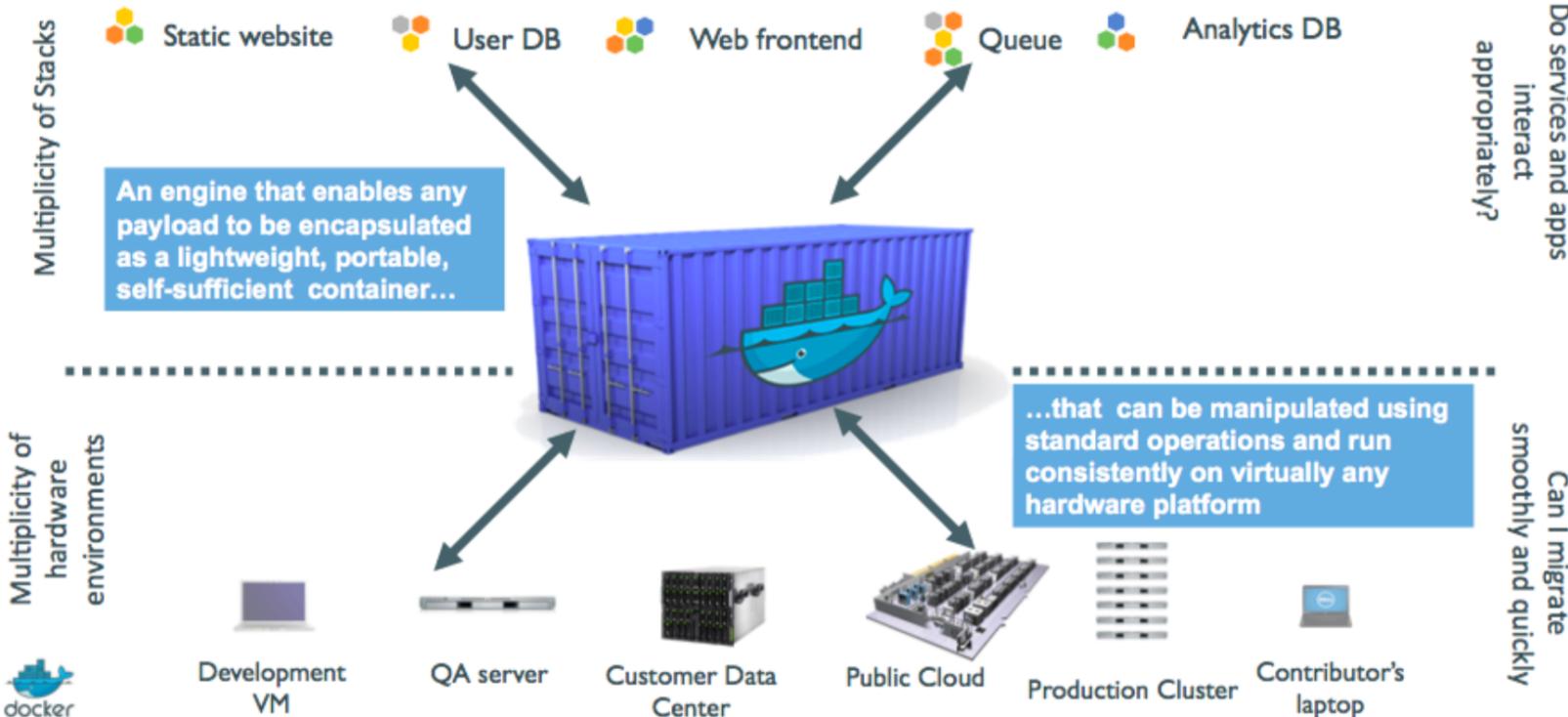


- 90% of all cargo now shipped in a standard container
- Order of magnitude reduction in cost and time to load and unload ships
- Massive reduction in losses due to theft or damage
- Huge reduction in freight cost as percent of final goods (from >25% to <3%)
- massive globalization
- 5000 ships deliver 200M containers per year



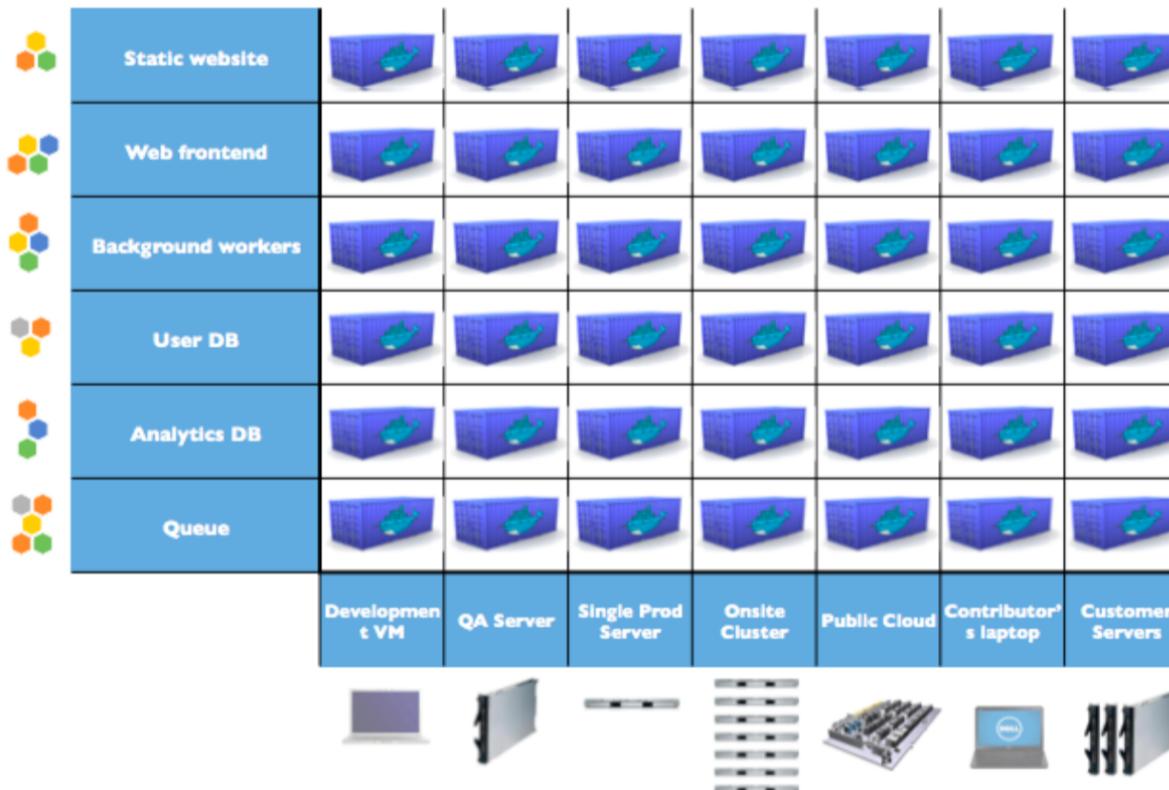
## About Docker

# A shipping container system for applications



## About Docker

# Eliminate the matrix from Hell

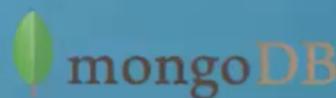


Web Server

Database

Messaging

Orchestration



Libraries

Dependencies

OS

Hardware Infrastructure

Web Server

Database

Messaging

Orchestration



mongoDB



The Matrix from Hell !!

Libraries

Dependencies

OS

Hardware Infrastructure

**why do you need Docker ?**

# **why do you need Docker ?**

1. compatibility/dependency

# **why do you need Docker ?**

1. compatibility/dependency
2. Long setup time

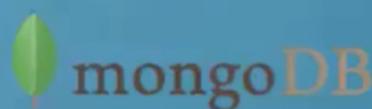
# **why do you need Docker ?**

1. compatibility/dependency
2. Long setup time
3. Different Dev/Test/Prod environments

Web Server



Database



Messaging



Orchestration



?

OS

Hardware Infrastructure

Web Server

Database

Messaging

Orchestration



Docker

OS

Hardware Infrastructure

Container

Web Server



Container

Database



Container

Messaging



Container

Orchestration



Libs

Deps

Libs

Deps

Libs

Deps

Libs

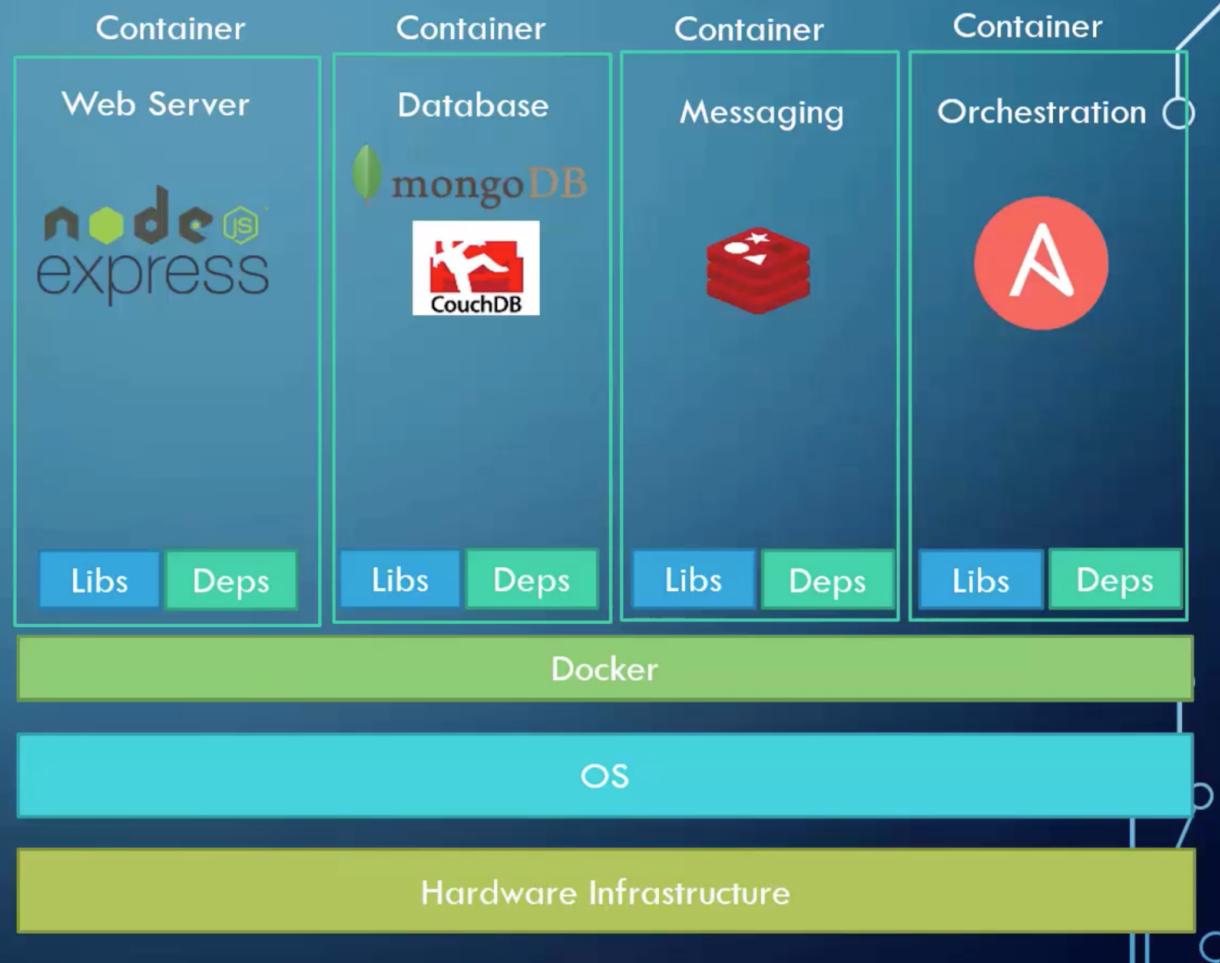
Deps

Docker

OS

Hardware Infrastructure

- Containerize Applications
- Run each service with its own dependencies in separate containers



# CONTAINERS



Processes

Network

Mounts



Processes

Network

Mounts



Processes

Network

Mounts



Processes

Network

Mounts

OS Kernel



Software

Software

Software

OS Kernel

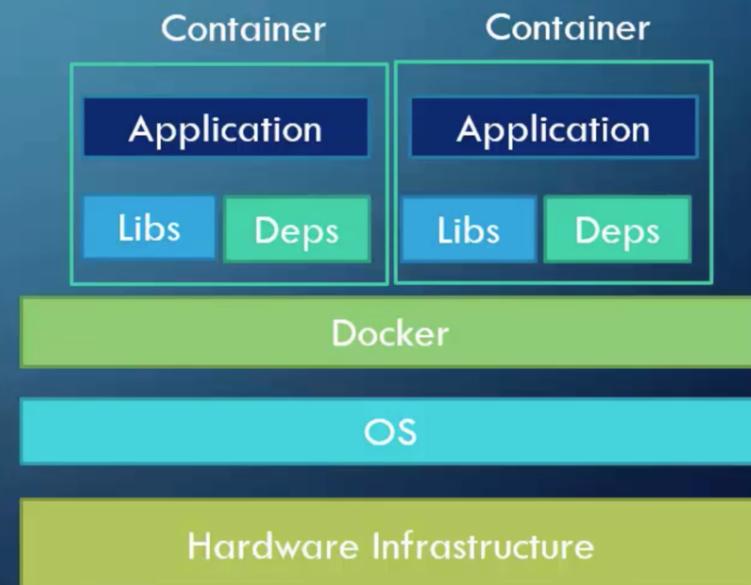
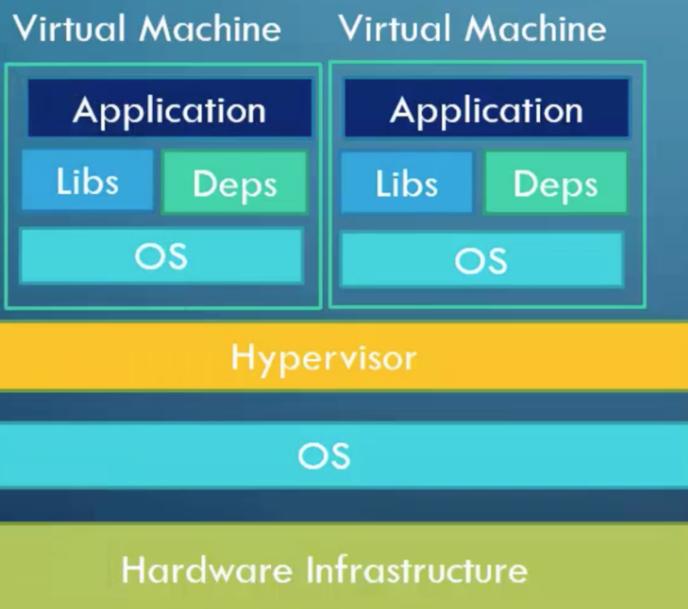
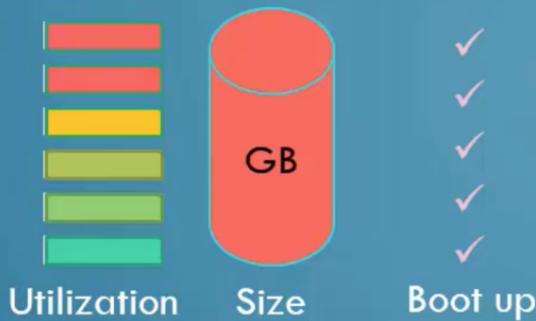


Docker

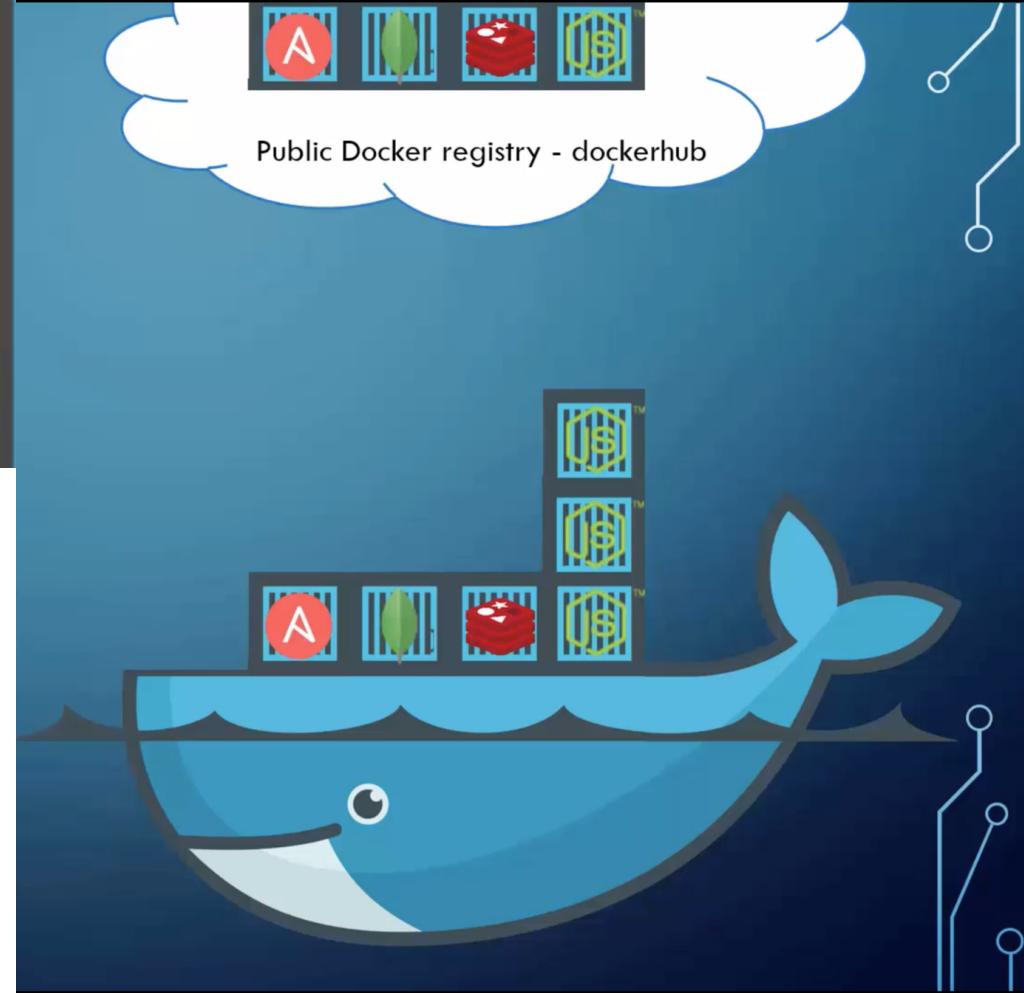


OS - Ubuntu

# CONTAINERS VS VIRTUAL MACHINES



```
docker run ansible  
docker run mongodb  
docker run redis  
docker run nodejs  
docker run nodejs
```



# CONTAINER VS IMAGE



Docker Image

Package  
Template  
Plan



Docker Container #1



Docker Container #2

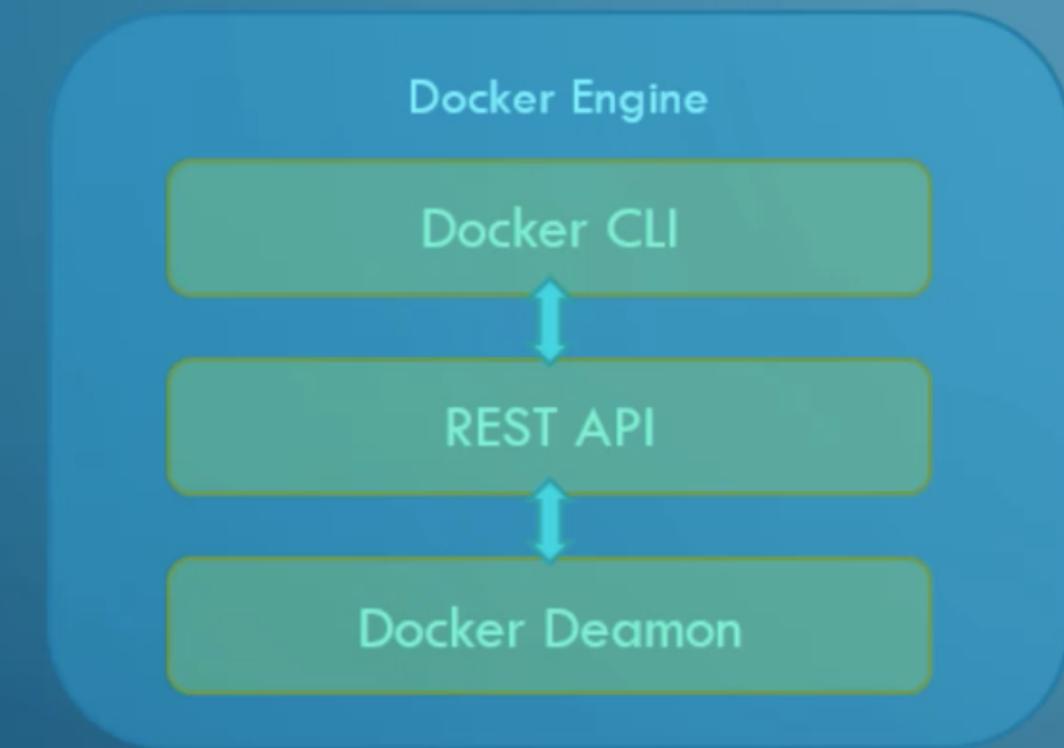


Docker Container #3

# Docker - Instalation

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

# ❖ DOCKER ENGINE



# CONTAINERIZATION

Process ID

Unix Timesharing

Network

Mount

Namespace

InterProcess

# • NAMESPACE - PID

Linux System

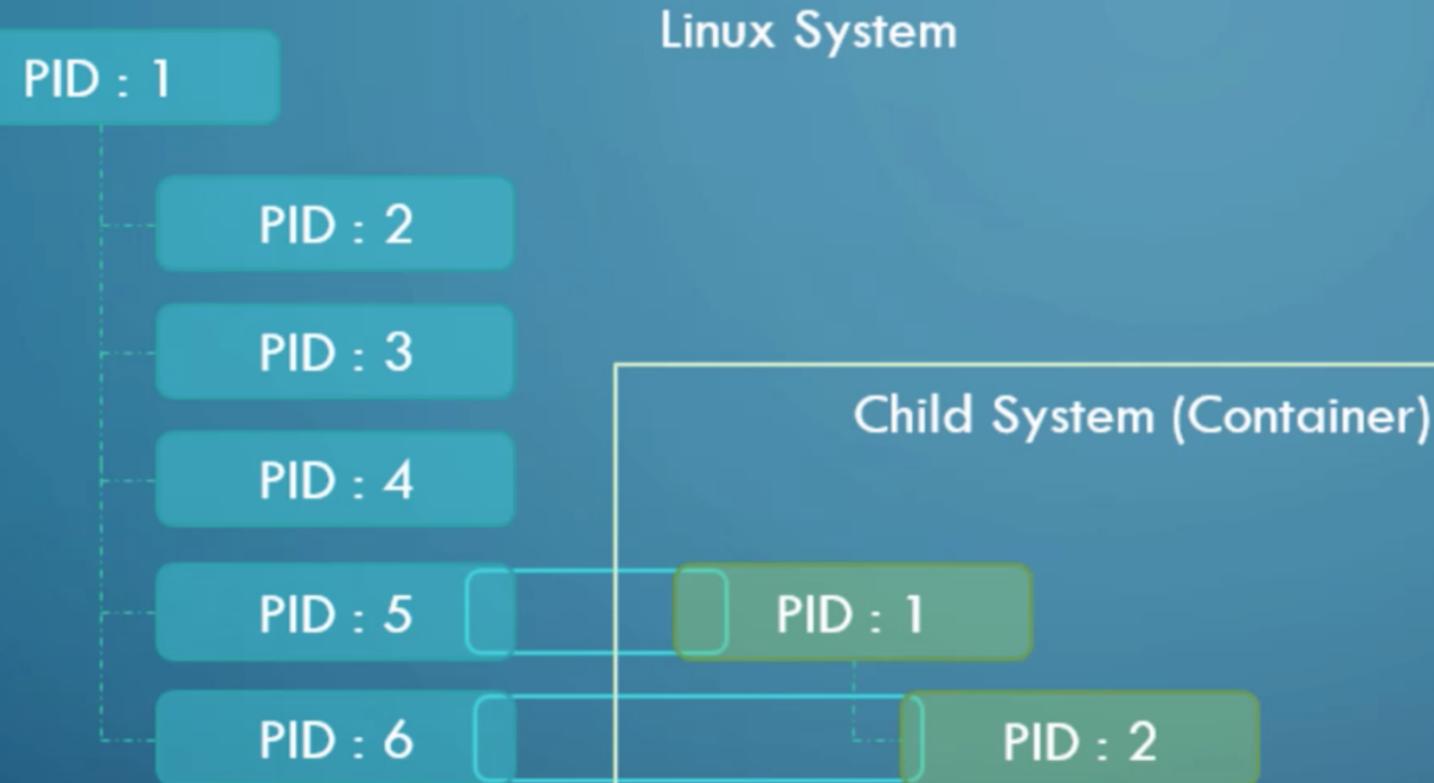
PID : 1

PID : 2

PID : 3

PID : 4

# ◦ NAMESPACE - PID



# cgroups

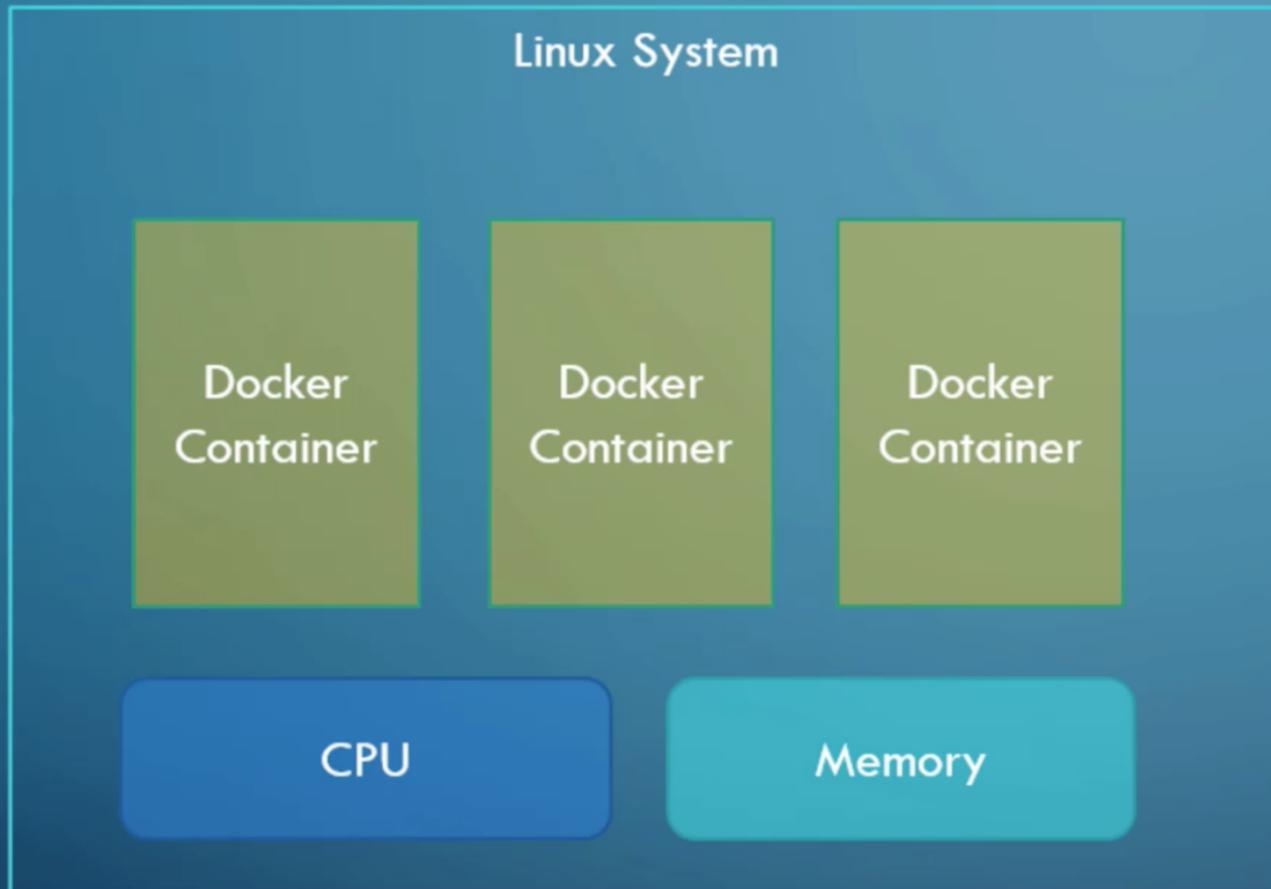
Linux System



CPU

Memory

# CGROUPS



```
docker run --cpus=.5 ubuntu
```

```
docker run --memory=100m ubuntu
```