Microservices

Docker: Containers

Kubernetes: Orchestrators

Ibrahim Ates
@brhmates

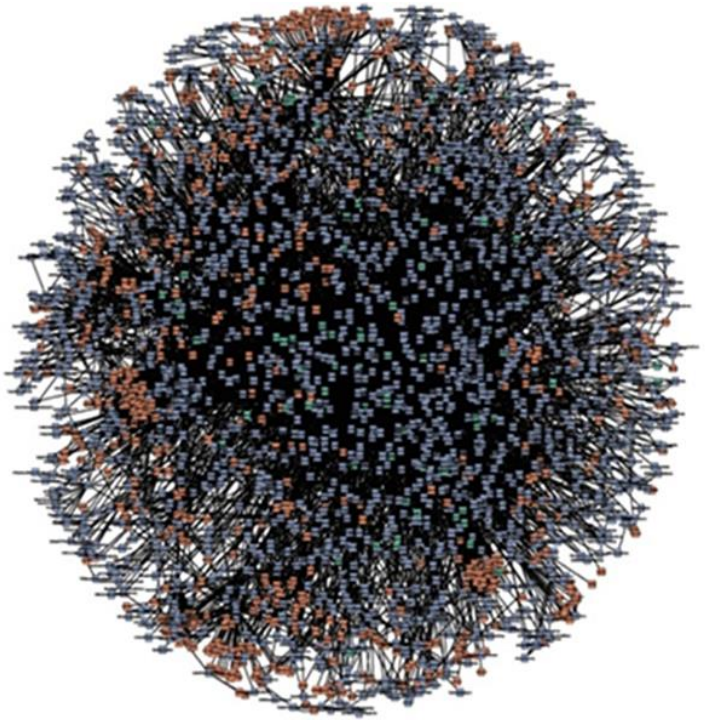
Image source: google

Contents

- What are Microservices?
- Why Using Microservices?
- Patterns / Monolithic vs Microservices Architecture
- Microservices are Perfect!?
- Important keys about microservices
- Docker: Container for Microservices
  - Where it came from?
  - What is Docker?
  - Why Docker?
  - Are there limitations to using Docker?
- Kubernetes : Orchestrator for Microservices
  - Where it came from?
  - What is Kubernetes?
  - Kubernetes Features

Image source: google

What are Microservices?



Amazon

Netflix

"Deathstar Diagrams"
Source: https://www.appcentrica.com/wp-content/uploads/2016/11/Microservices-Architecture-1.png

What are Microservices?

James Lewis and Martin Fowler

- the microservice architectural style is an approach to
- developing a single application* as a **suite of small services**,
- each **running in its own process** and communicating with lightweight mechanisms, often an HTTP resource API.
- These services are built around business capabilities and
- **independently deployable** by fully automated deployment machinery.
- There is a **bare minimum of centralized management**** of these services, which may be written in different programming languages and use different data storage technologies.

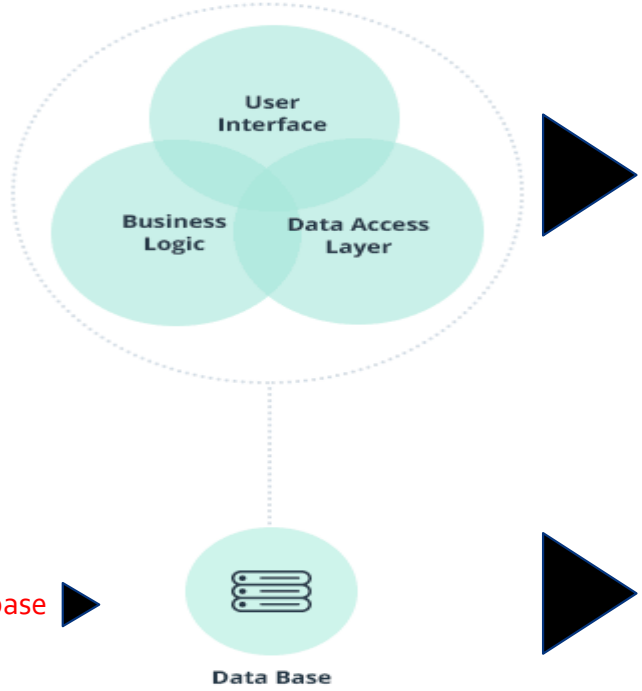
* Single Resposibilty Principle(SRP)

** Loosly coupled

Why using Microservices?

Traditional / monolithic architecture could not meet to requirements of the new World.

Traditional approach:Monolithic Architecture

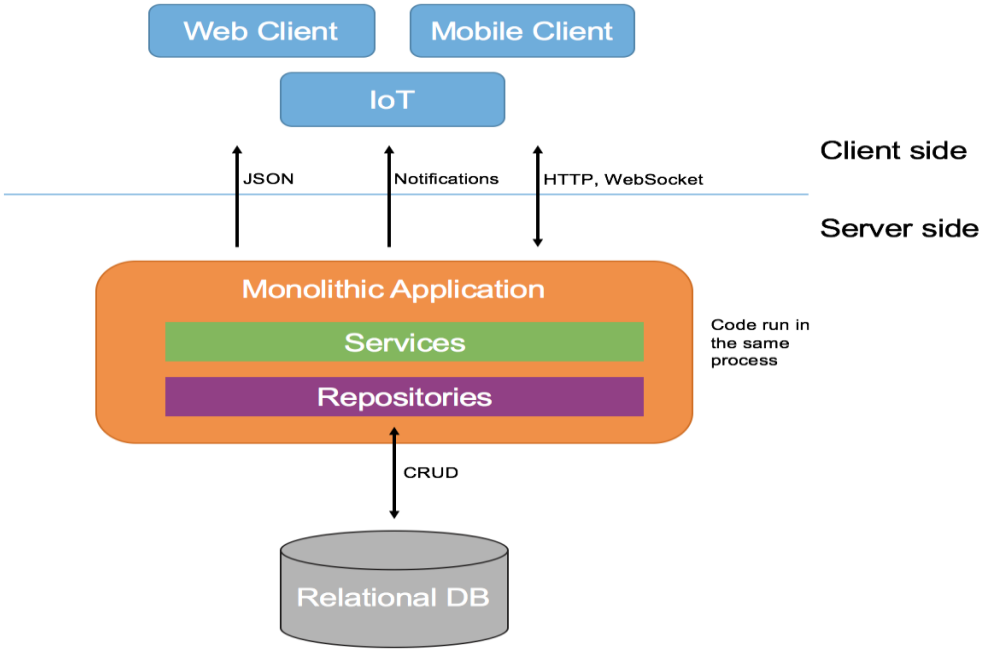


Monolithic tiers

- Presentation: JS / IOS / Android
- Application: Java / Python
- Database: MongoDB, PostgreSQL, Oracle

Image Source: https://www.intellias.com/the-decisive-role-of-microservices-in-modern-software-development/

Monolithic Architecture Example



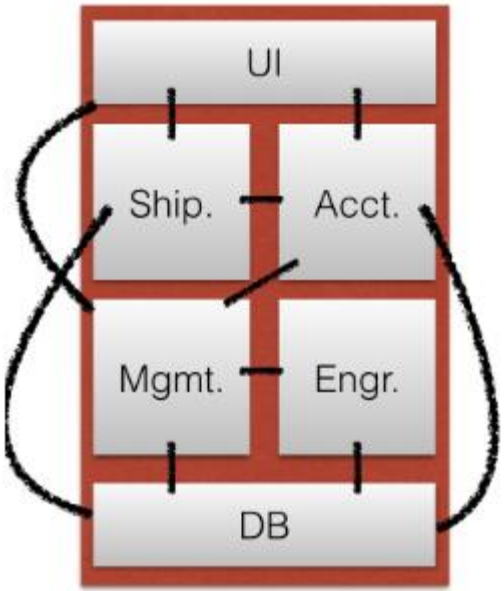
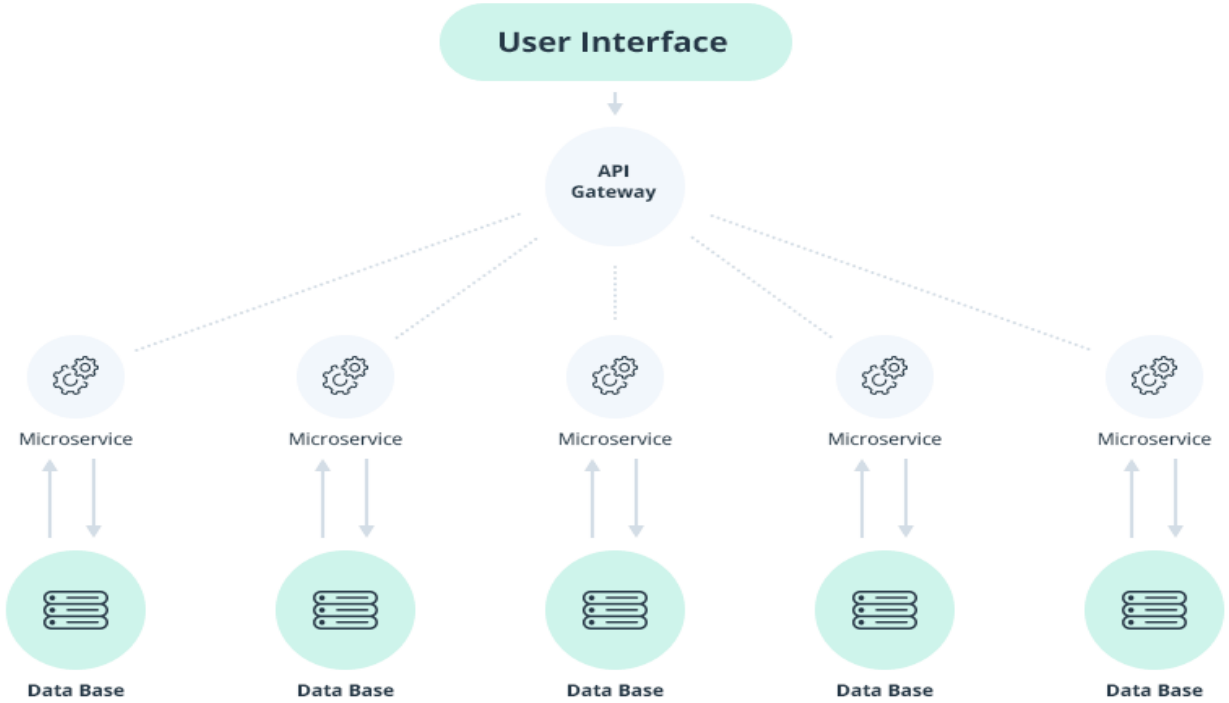


Image Source: https://learn.g2crowd.com/microservices

That is why we need to new approach.

New approach is Microservices



Source: https://www.intellias.com/the-decisive-role-of-microservices-in-modern-software-development/

Flexibility…

Reliability…

Development(Test, Maintenance) Speed…

Building Complex Application…

Scalability…

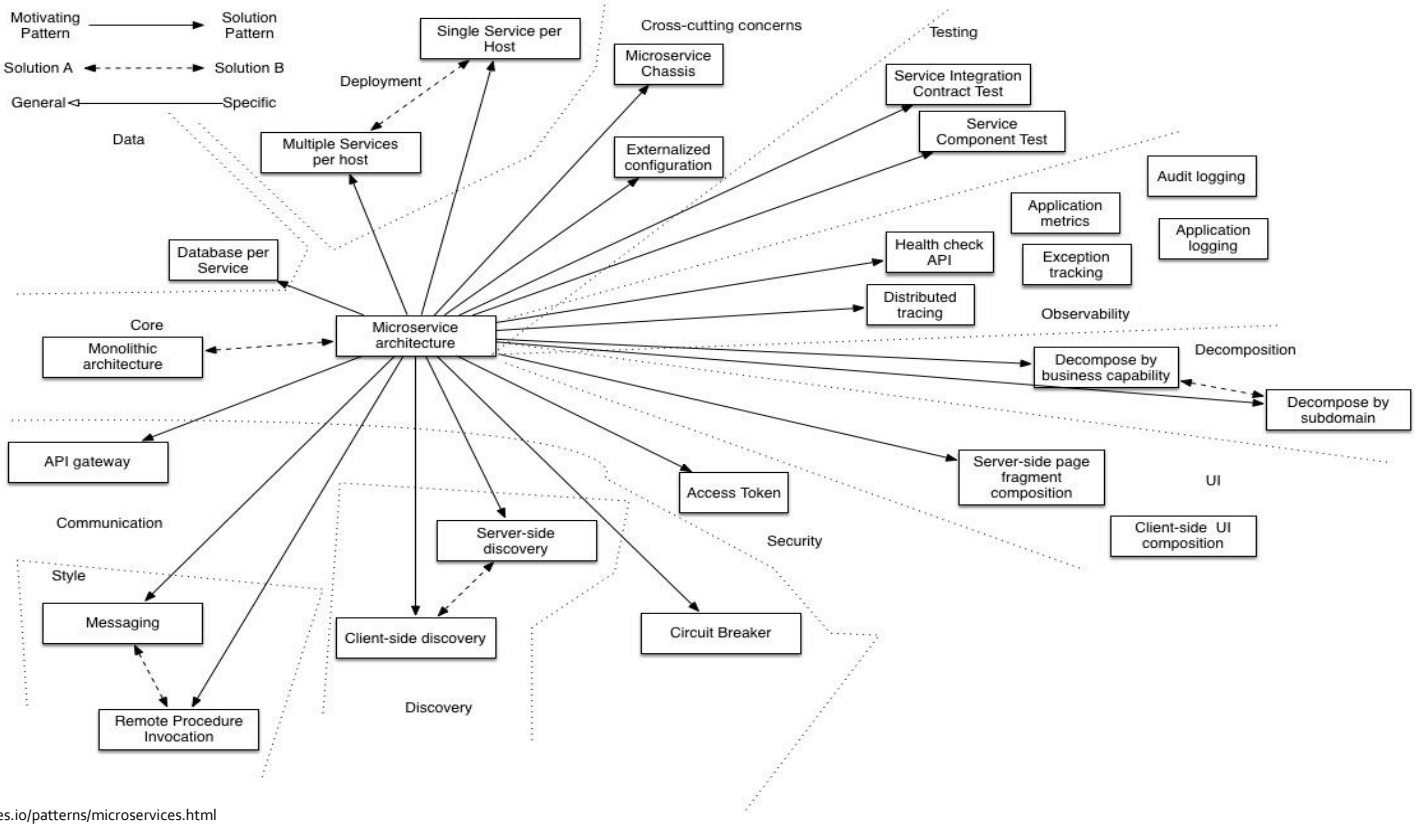
Continuous Deployment…

For new World Microservices are good solution, but unfortunately have drawbacks at the same time.

Increased Memory Consumption…

Distributed System…

Microservices with Distributed System



Source:https://microservices.io/patterns/microservices.html

Important keys about microservices…

- Data-base
  // MongoDB, Apache Cassandra, Redis

- Data Synchronization
  // Apache Kafka, Akka

- Testing
  // Hoverfly, Pact, VCR

- Logging
  // Istio, Loggly

- Security / Autherization and Authentication
  // Kerberos, Open ID

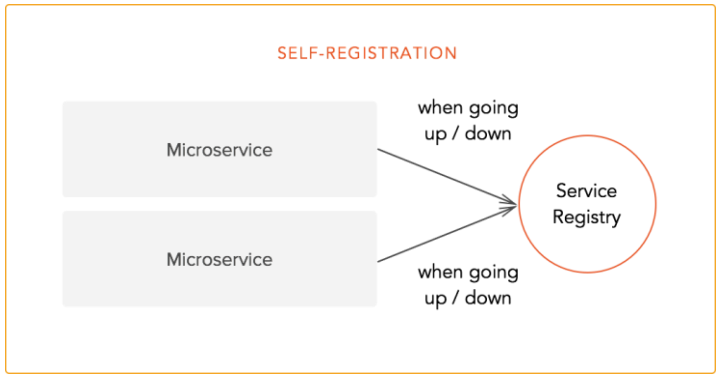
- Continuous Delivery
  // Jenkins, Asgard, Aminator

- Containers
  // Docker**, Rocker, Rkt

- Orchestrating
  // Kubernetes**, Mesos, Docker Swarm

- Distributed Services

  - Service Registry and Discovery
    // Eureka, Consul, Zookeeper



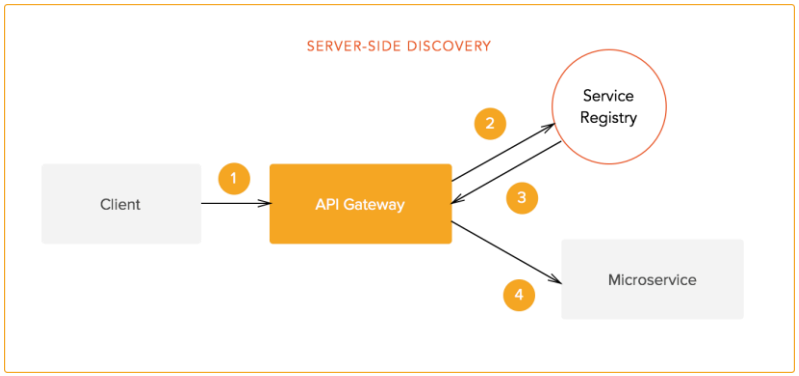


Image Source: https://autho.com/blog/an-introduction-to-microservices-part-3-the-service-registry/

- Distributed Services

  - Circuit Breaker: *Failure Managment*

    - Closed

    - Open

    - Half-Open

    // Hsytrix

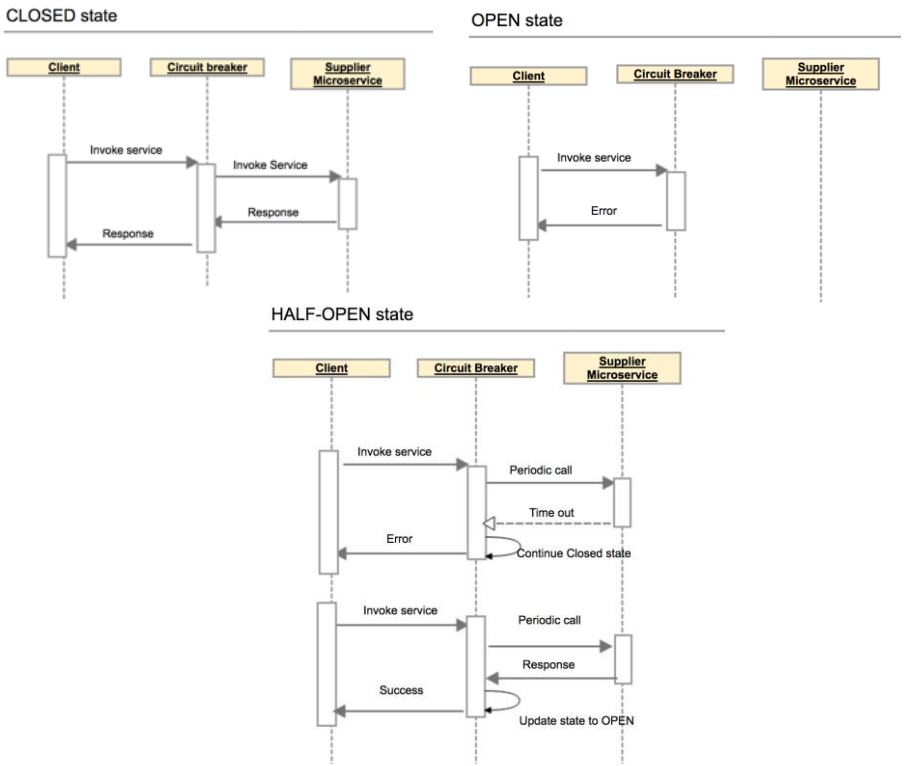
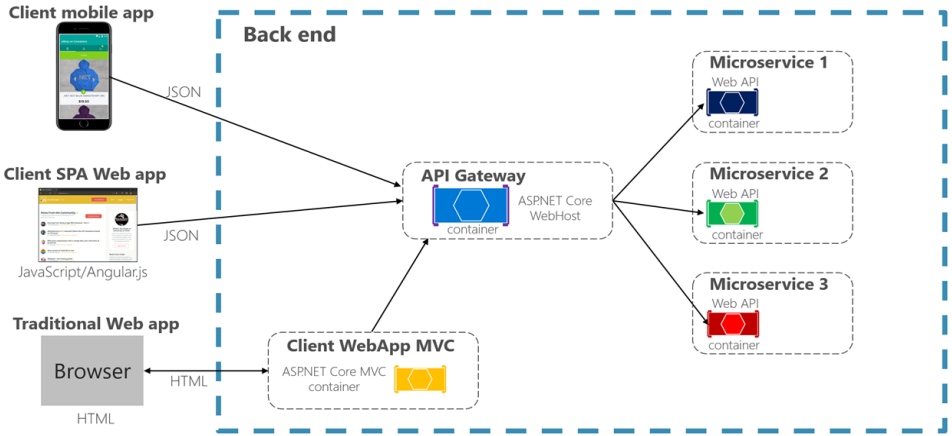


Image Source: https://techblog.constantcontact.com/software-development/circuit-breakers-and-microservices/

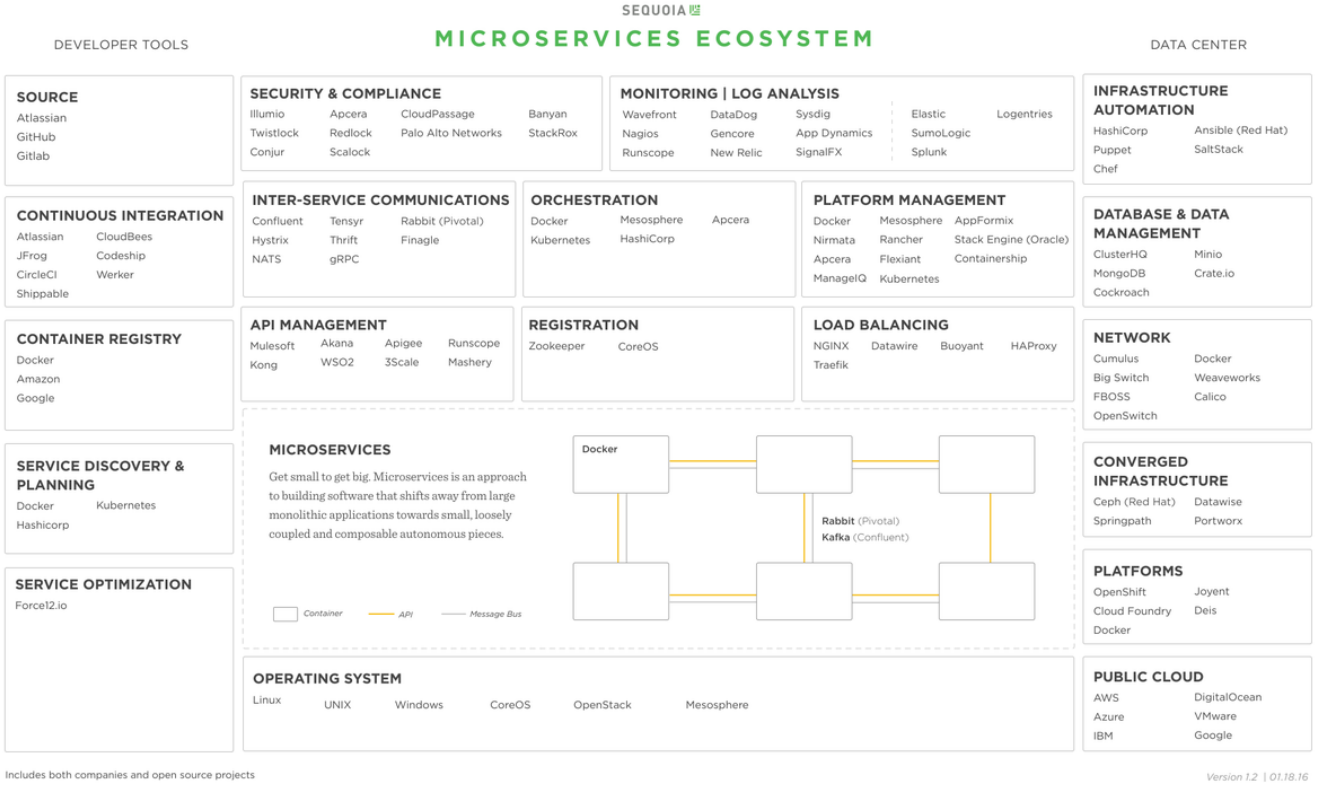
- Distributed Services

  - Gateway
    // Zuul, Netty, Feign



Source: Microsoft

Microservices Ecosystem



Architecture evolution for some tech companies

- eBay
  - Monolithic Perl -> Monolithic C++ -> Java -> Microservices
- Amazon
  - Monolithic C++ -> Java / Scala-> Microservices
- Twitter
  - Monolithic Rails -> JS / Rails / Scala -> Microservices

Docker : Container For Microservices...

Where it came from?

- The name Docker comes from British colloquialism meaning dock worker
- First version released 2013
- Written in Go/Golang

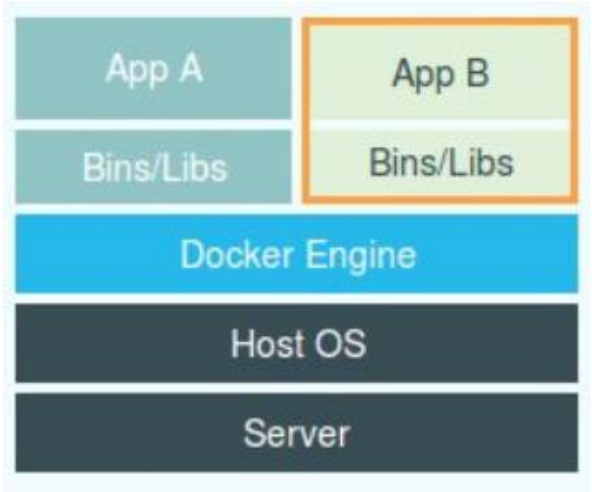
Source: https://towardsdatascience.com/how-docker-can-help-you-become-a-more-effective-data-scientist-7fc048ef91d5

What is Docker?

- « a software platform that allows you to build, test, and deploy applications quickly. Docker packages software into standardized units called containers** that have everything the software needs to run including libraries, system tools, code, and runtime.»

Amazon

- A lightweight way to virtualize applications
- Shortly write your code in any language and run anywhere - WORA

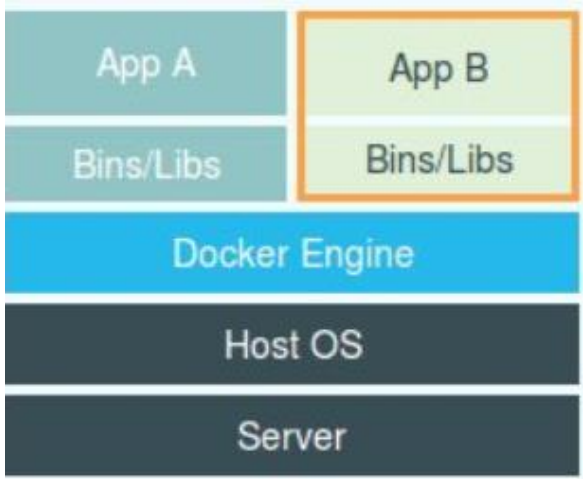


Source: https://rancher.com/playing-catch-docker-containers/

Is Docker a Virtual Machine?

- Virtual but not machine, it is a container what virtualized/containerized apps

Virtual Machines



Docker / Container

Image Source: http://leopard.in.ua/presentations/rubyc_2015/?full#DockerFigSlide

Docker : Container For Microservices…

Source:
http://leopard.in.ua/presentations/rubyc_2015/?full#DockerDisadvantagesSlide

Why Docker?

- Efficiency

- Velocity(to market, deployment)…

- Portability…

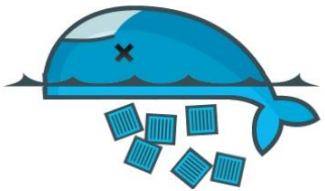
- Sharability…

Source: https://www.docker.com/why-docker

Any limitations to using Docker?

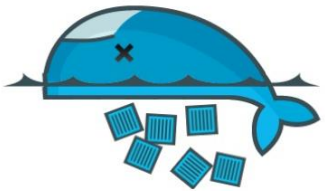
- Docker, by itself is very good when it manages single container

Docker Difficulties



- But when you start using more and more containers and containerized app get struggling with very difficulties.

Docker Difficulties



Containers are great but not enough at…

- Managment
- Orchestrating
- Scalling up
- Redundancy
- Scheduling
- Service Discovery
- Resiliency
- Configuration

That is why we need to Orchestrators.

Source: http://leopard.in.ua/presentations/rubyc_2015/?full#DockerDisadvantagesSlide

Kubernetes: Orchestrator for Microservices/Containers…

Where it came from?

- Kubernetes word from Greek ‘Helmsman’
  < the person who steers a ship

- Also the root of the words ‘governor’ and ‘cybernetic’

Kubernetes: History

- Born in Google and Informed by Google's experiences and internal systems(borg)
- Donated to Cloud Native Computing Foundation(CNCF) in 2014 (open source)
- Written in Go/Golang
- %100 Open Source
- https://github.com/kubernetes/kubernetes
- Often shortened to K8s

Source: wiki, pluralsight

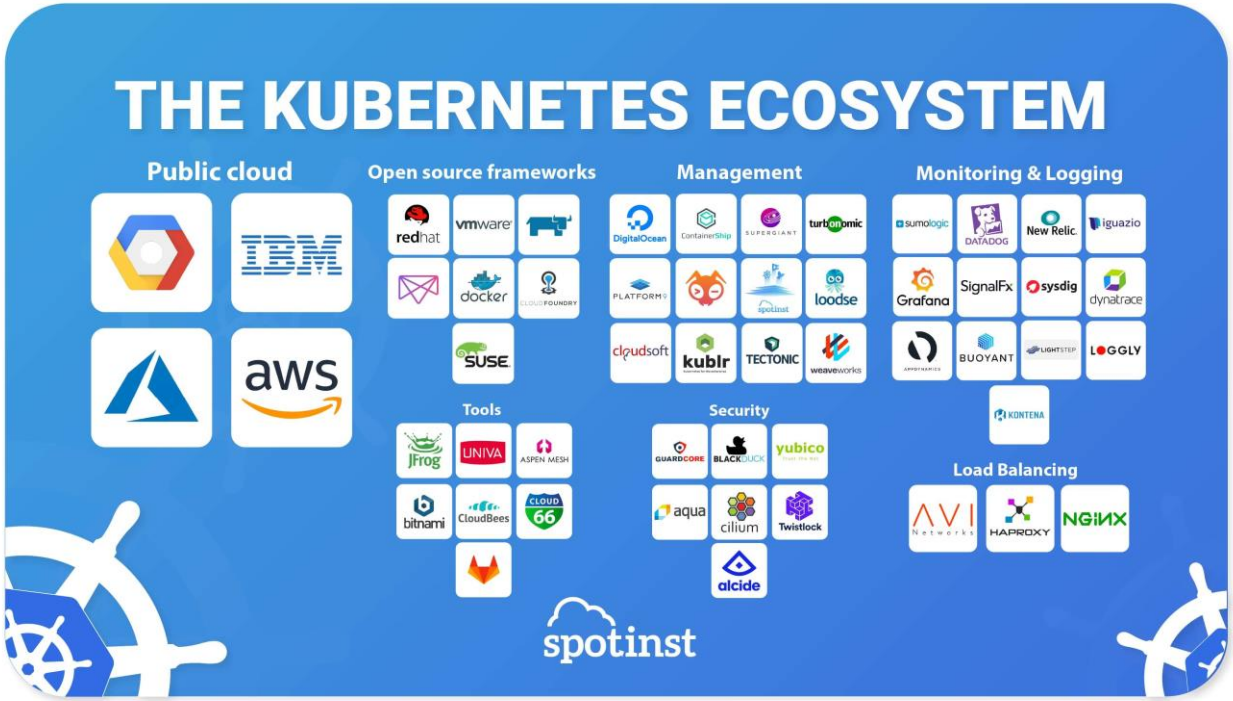
What is Kubernetes?

Kubernetes is

- An orchestrator for microservice apps run on containers
- A container scheduler
- Managment containerized cluster/applications and automating deployment tool
- Supports multiple cloud and multiple container runtimes

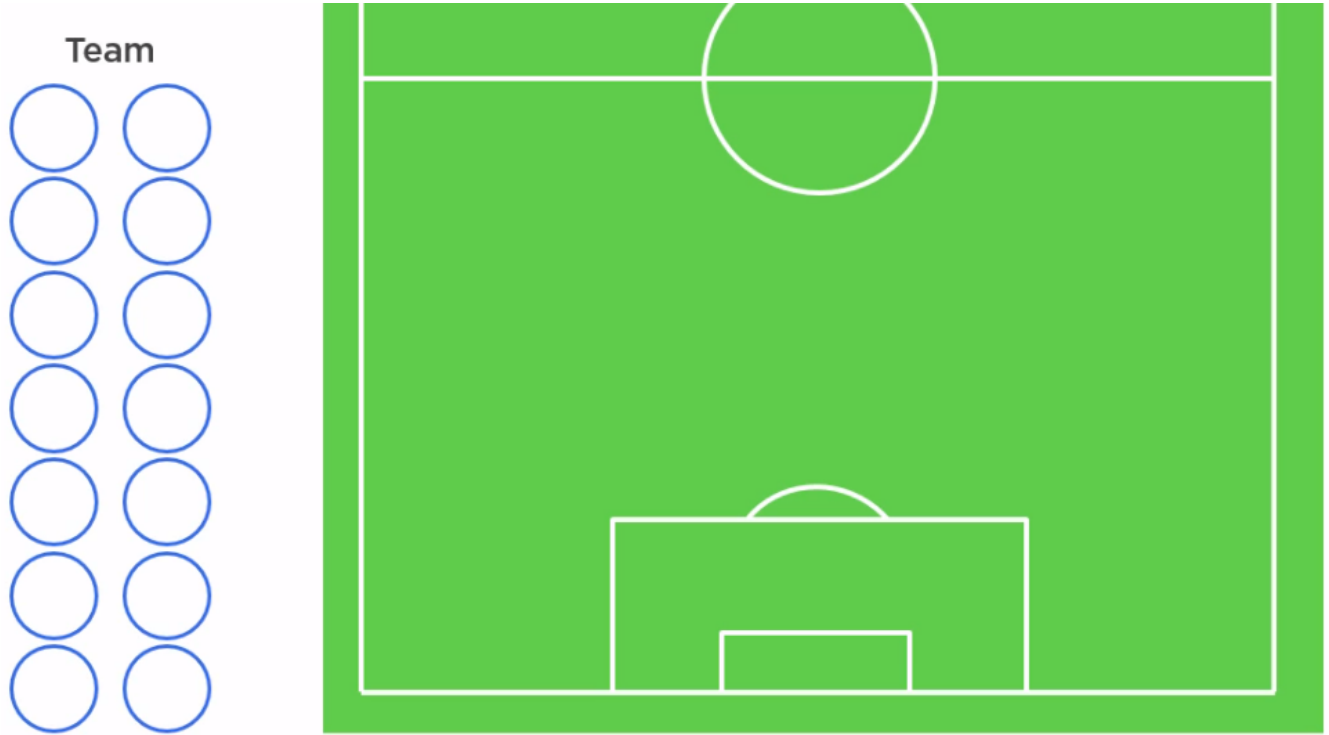
Source: https://kuberneteslab.com/

Kubernetes



Source: https://kuberneteslab.com/

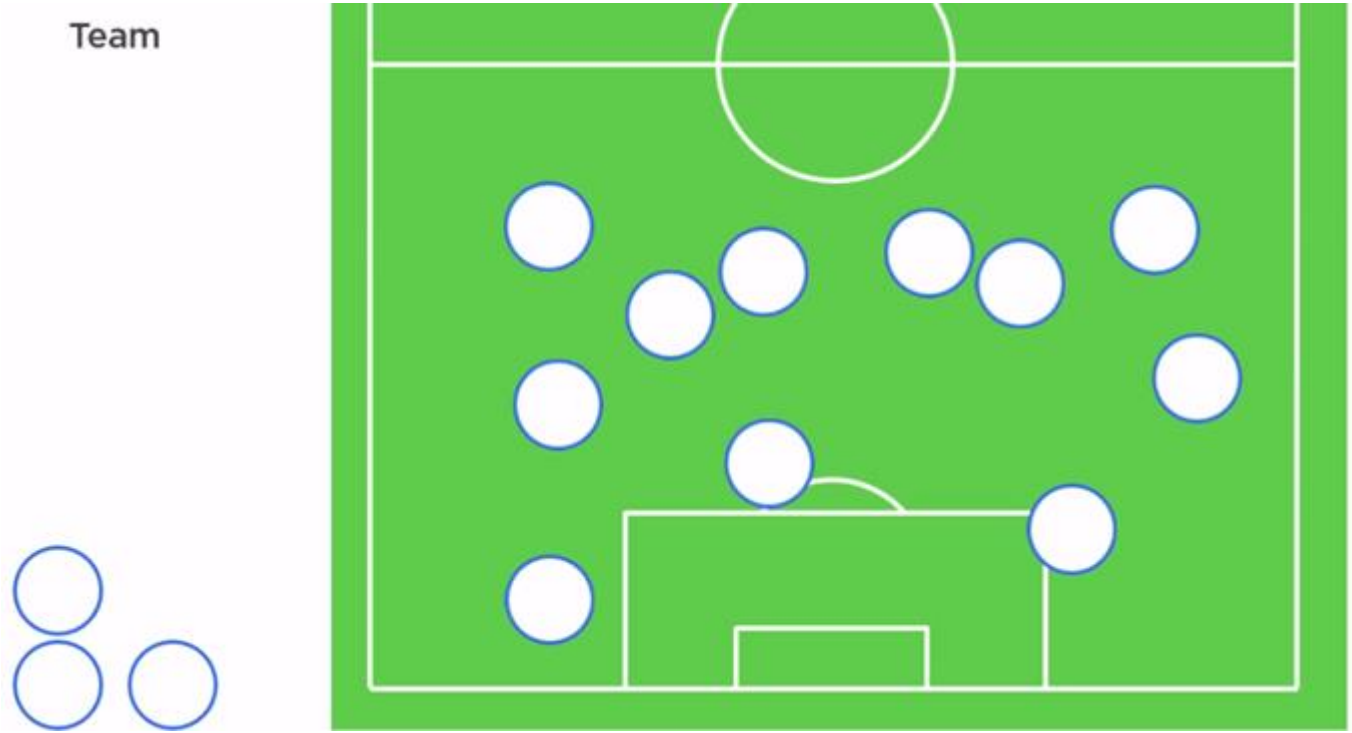
Think a team without a coach / manager



kubernetes

Image source: https://app.pluralsight.com/library/courses/getting-started-kubernetes/table-of-contents

How are they organize?



kubernetes

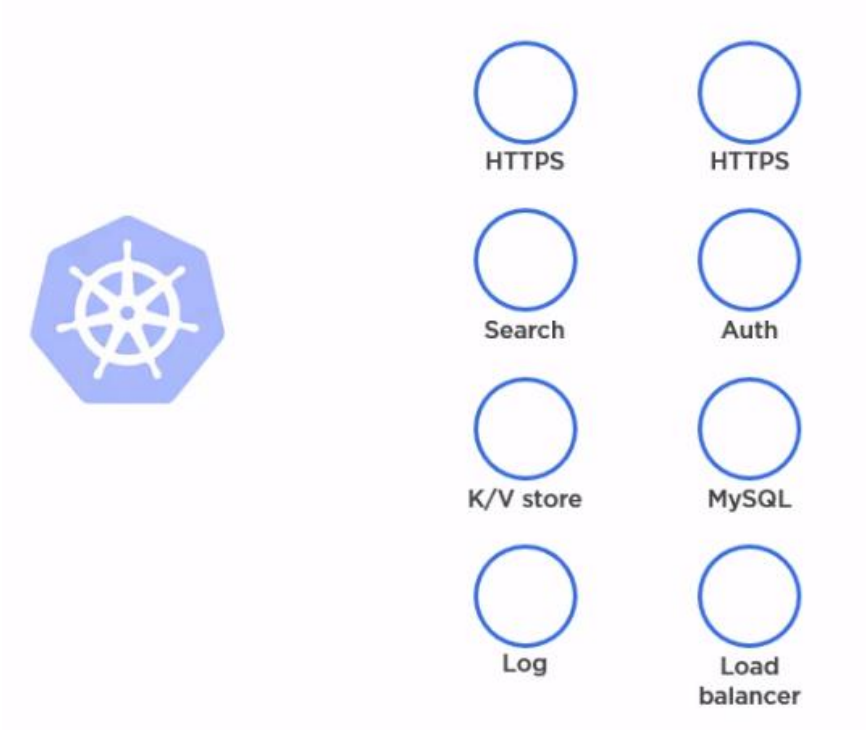
Image source: https://app.pluralsight.com/library/courses/getting-started-kubernetes/table-of-contents

But if they have…

kubernetes

Image source: https://app.pluralsight.com/library/courses/getting-started-kubernetes/table-of-contents

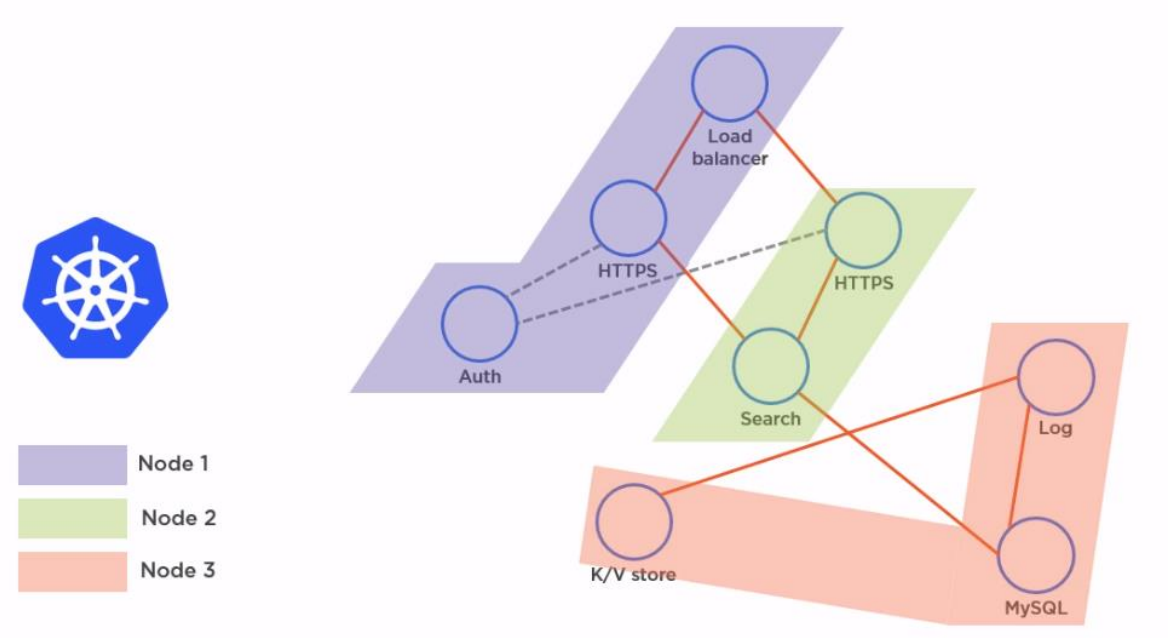
Kubernetes do exactly the same thing like manager.



kubernetes

Image source: https://app.pluralsight.com/library/courses/getting-started-kubernetes/table-of-contents

Managed, Organized, Orchestrated



kubernetes

Image source: https://app.pluralsight.com/library/courses/getting-started-kubernetes/table-of-contents

Kubernetes Features...

Service Discovery and Load Balancing…

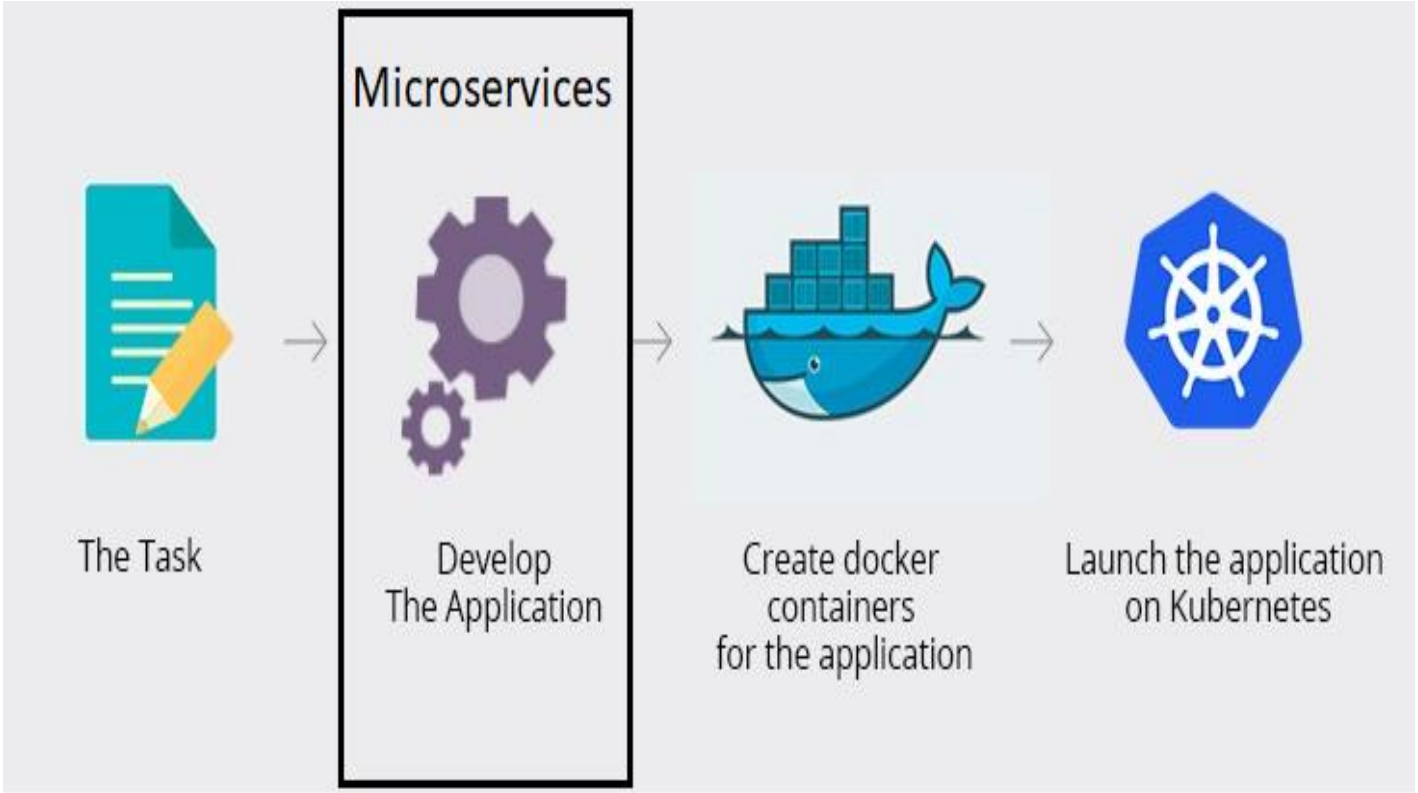
Storage Orchestration…

Automated rollouts and rollbacks...

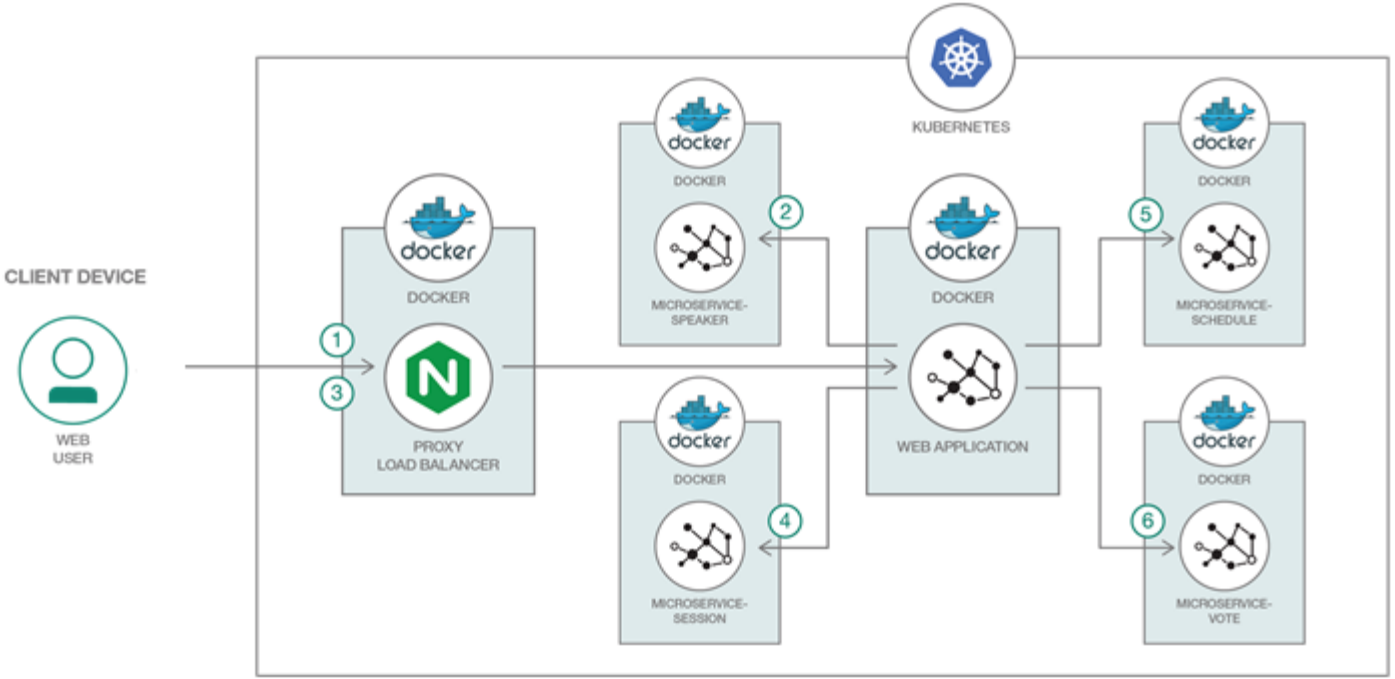
Batch Execution…

Cost Savings…

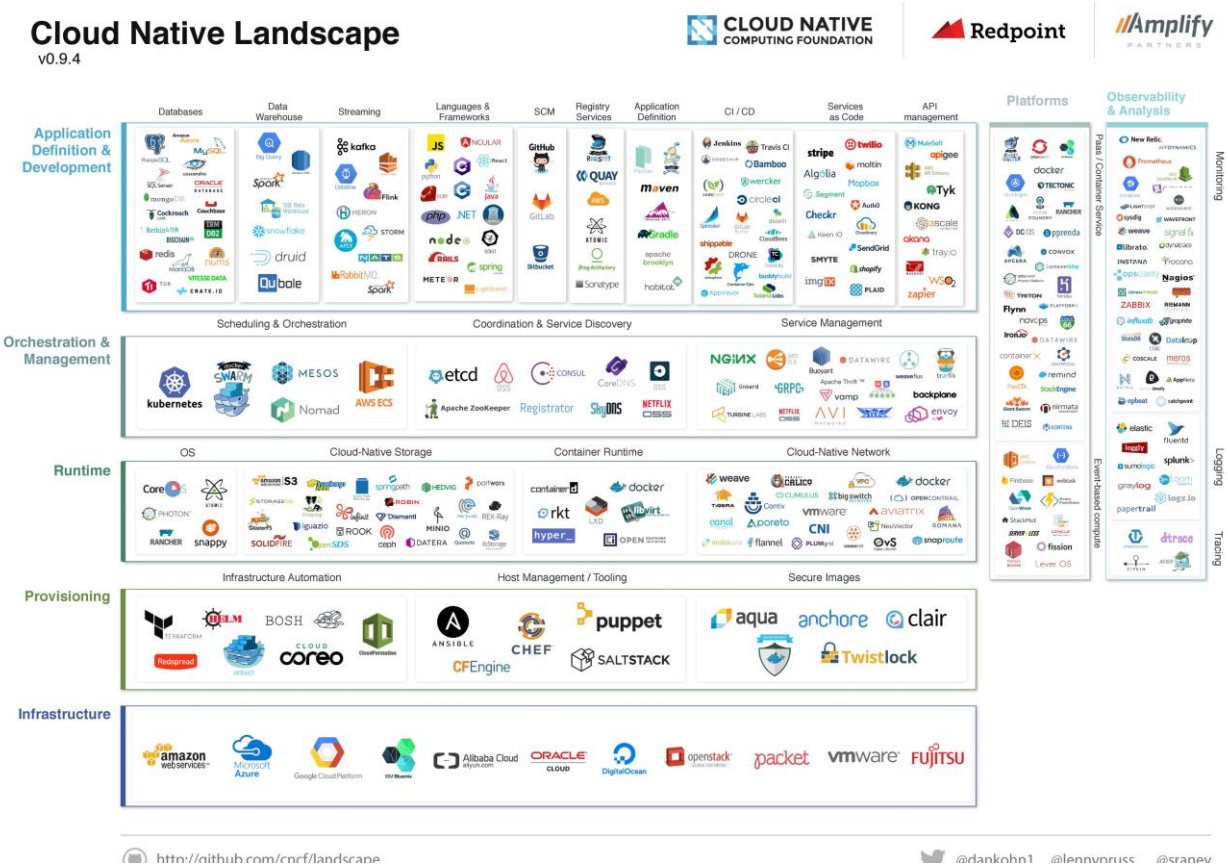
Microservices > Docker > Kubernetes Flow



Microservices > Docker > Kubernetes Flow Example



Source: Google


Cloud Native Landscape
v0.9.4
CLOUD NATIVE COMPUTING FOUNDATION
Redpoint
Amplify PARTNERS
Databases
Data Warehouse
Streaming
Languages & Frameworks
SCM
Registry Services
Application Definition
CI / CD
Services as Code
API management
Platforms
Observability & Analysis
Application Definition & Development
Scheduling & Orchestration
Coordination & Service Discovery
Service Management
Orchestration & Management
kubernetes
MESOS
Nomad
AWS ECS
etcd
CONSUL
Apache ZooKeeper
Registrator
NGINX
gRPC
vamp
backplane
envoy
OS
Cloud-Native Storage
Container Runtime
Cloud-Native Network
Runtime
docker
rkt
hyper_
weave
CNI
flannel
Infrastructure Automation
Host Management / Tooling
Secure Images
Provisioning
BOSH
ANSIBLE
CHEF
puppet
SALTSTACK
CFEngine
aqua
anchore
clair
Twistlock
Infrastructure
amazon web services
Microsoft Azure
Google Cloud Platform
Alibaba Cloud
ORACLE CLOUD
DigitalOcean
openstack
packet
vmware
FUJITSU
PaaS / Container Service
Event-based compute
Monitoring
Logging
Tracing
http://github.com/cncf/landscape
@dankohn1
@lennypruss
@sranev

Any questions?