_Containers! SQL Server + Docker + Kubernetes

William Lino Oliveira
Cofundador e Consultor

DataTuning_

_WhoAmI

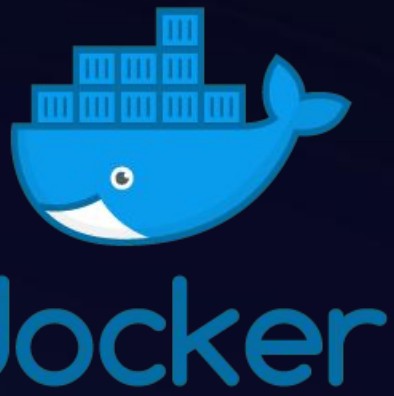
```yaml
---
apiVersion: v1
kind: Gordinho
metadata:
  name: William Lino Oliveira
  namespace: Will
spec:
  hobs:
    - Cinema\Series
    - Counter Strike\Fifa
    - Futebol
  jobs:
    - Co-Founder & Consultant - Data Tuning
    - Co-Founder & DevOps Engineer - Flapper
    - DBA - Grande Instituição Financeira
  socialMedia:
    - facebook - https://www.facebook.com/will.onil
    - twitter - @WillioniL
    - linkedin - https://www.linkedin.com/in/williamlinooliveira/
    - dev.io - @williamloliveira
  contatos:
    - e-mail: williamloliveira@hotmail.com
    - datatuning: william@datatuning.com.br
    - flapper: w.oliveira@flapper.com.br
    - telefone: (11) 972262055
```

DataTuning_

_Quem é a Data Tuning

- Consultoria em Bancos de Dados com foco em performance
- Site: https://datatuning.com.br
- Blog: https://blog.datatuning.com.br
- Twitter: @DataTuning
- Linkedin: https://www.linkedin.com/company/data-tuning/
- Youtube: https://www.youtube.com/channel/UCsd5MjHlc9dNDQV6NTf_IQQ
- GitHub: https://github.com/datatuning

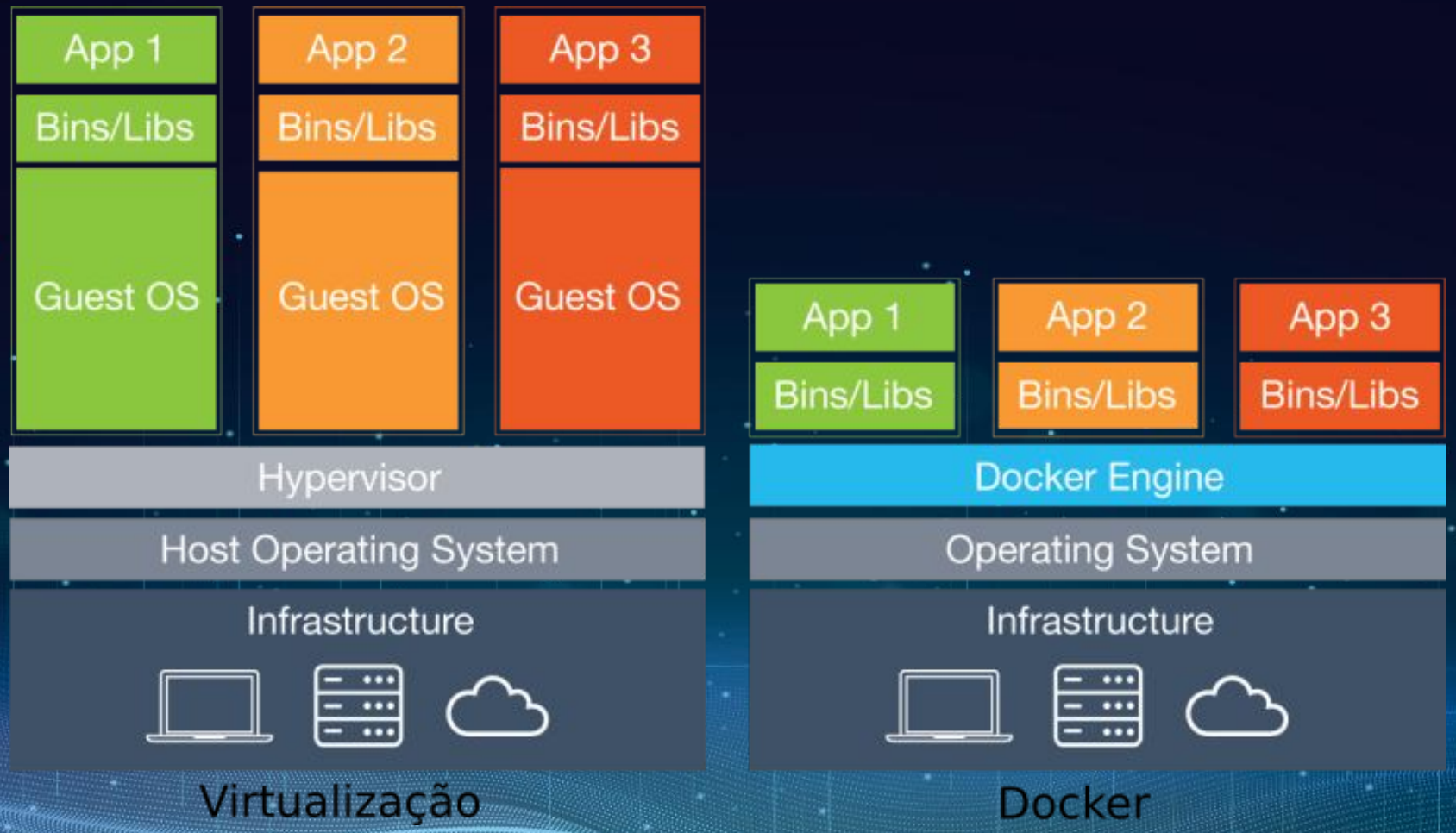
_Docke
r



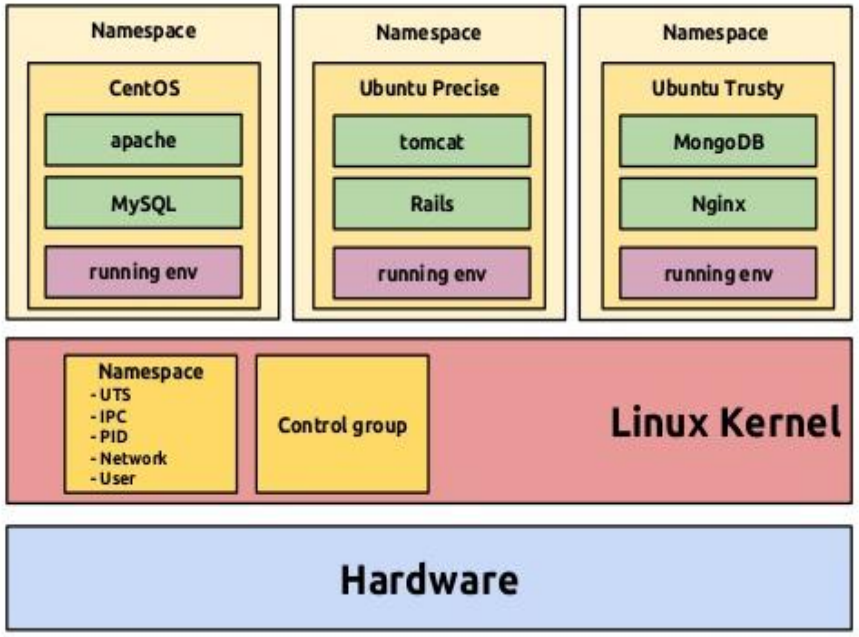
- Lançado em 2013
- Por Solomon Hykes (dotCloud -> Docker)
- Escrito em Go
- Revolucionou a forma de provisionar containers e fazer software

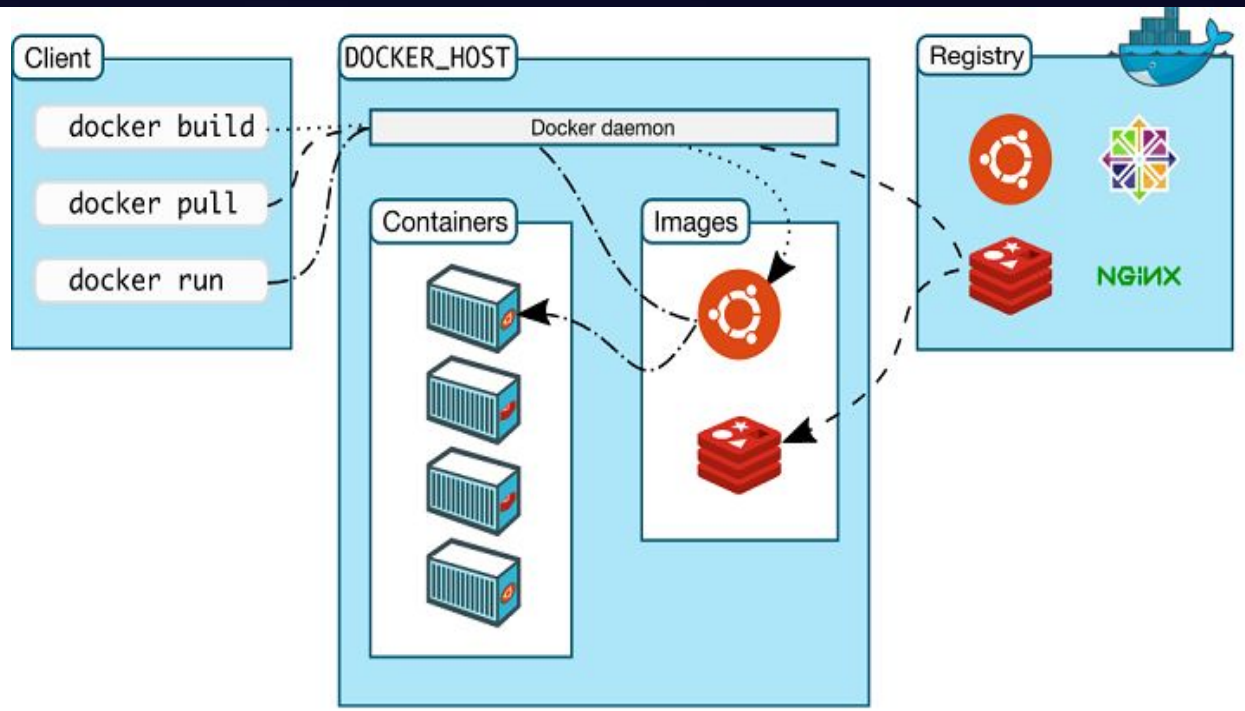
https://www.youtube.com/watch?v=wW9CAH9nSLs

_Docke
r


App 1
Bins/Libs
Guest OS
App 2
Bins/Libs
Guest OS
App 3
Bins/Libs
Guest OS
Hypervisor
Host Operating System
Infrastructure
Virtualização
App 1
Bins/Libs
App 2
Bins/Libs
App 3
Bins/Libs
Docker Engine
Operating System
Infrastructure
Docker

_Docke
r


Namespace
CentOS
apache
MySQL
running env
Namespace
Ubuntu Precise
tomcat
Rails
running env
Namespace
Ubuntu Trusty
MongoDB
Nginx
running env
Namespace
- UTS
- IPC
- PID
- Network
- User
Control group
Linux Kernel
Hardware

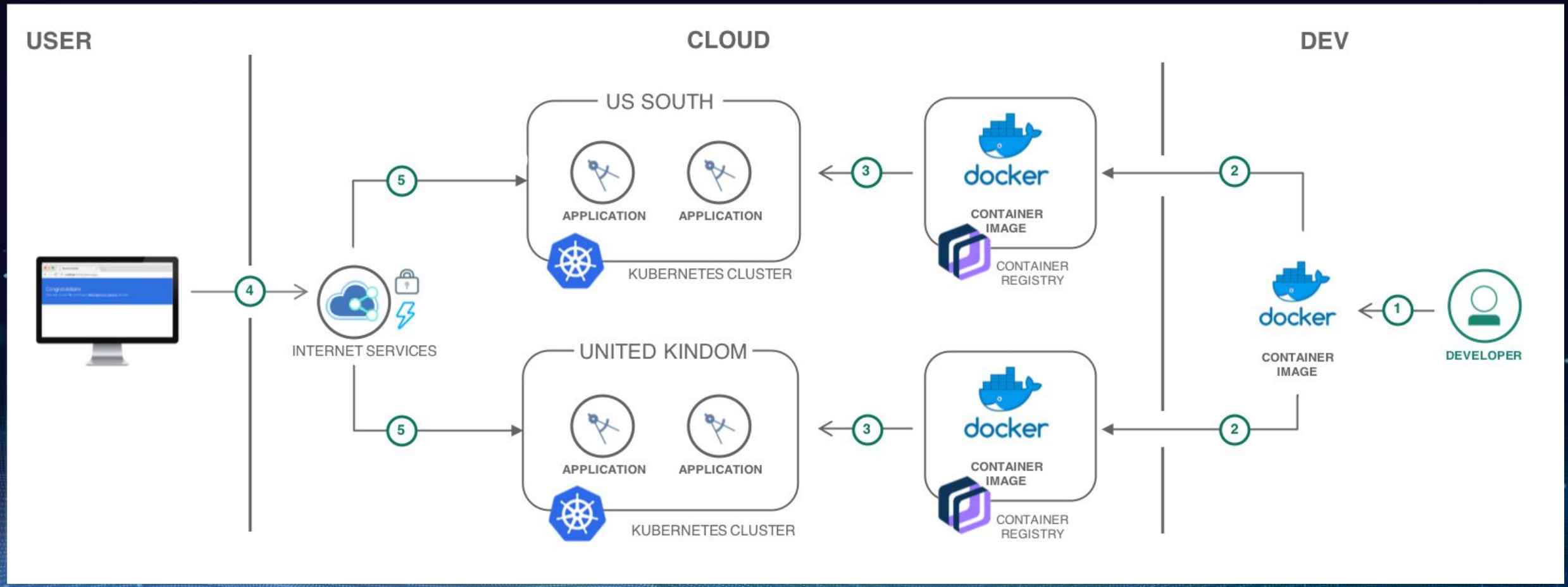

Client
docker build
docker pull
docker run
DOCKER_HOST
Docker daemon
Containers
Images
Registry
NGINX

_Kubernetes

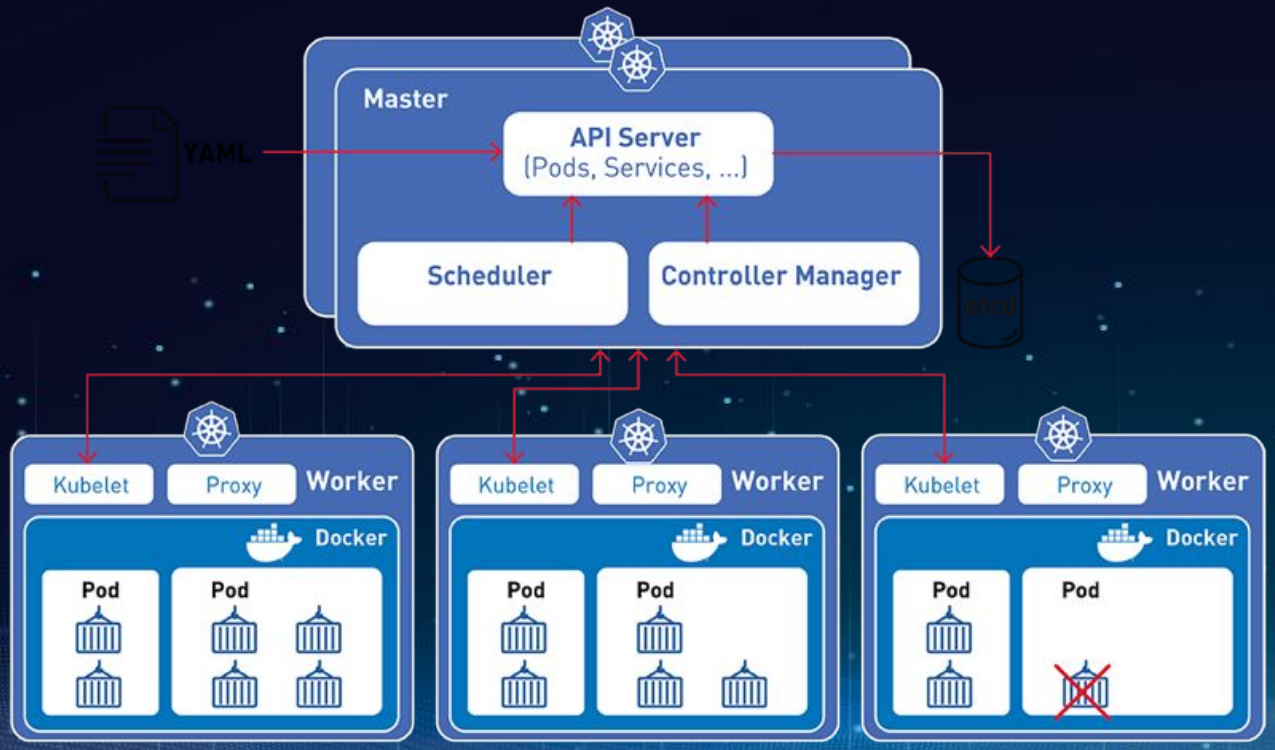
- Lançado em 2014
- Concebido pelo Google
- Mantido pela Cloud Native Computing Foundation
- Escrito em Go
- Orquestrador de container mais utilizado atualmente
- Segundo maior projeto do GitHub

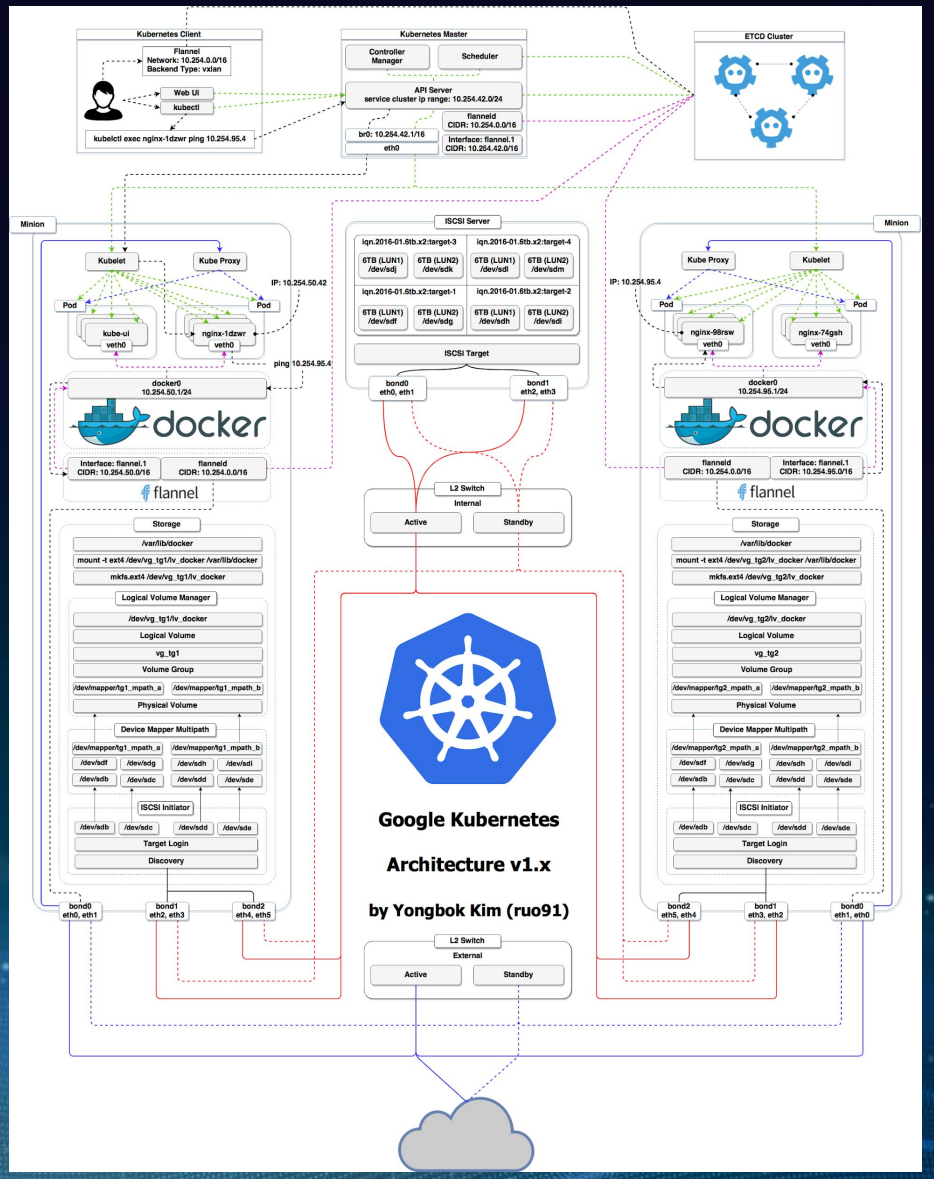
DataTuning_

_Kubernetes


USER
CLOUD
DEV
US SOUTH
APPLICATION
APPLICATION
KUBERNETES CLUSTER
UNITED KINDOM
APPLICATION
APPLICATION
KUBERNETES CLUSTER
INTERNET SERVICES
docker
CONTAINER IMAGE
CONTAINER REGISTRY
docker
CONTAINER IMAGE
CONTAINER REGISTRY
docker
CONTAINER IMAGE
DEVELOPER
1
2
3
4
5
5
2
3

_Kubernetes


YAML
Master
API Server
(Pods, Services, ...)
Scheduler
Controller Manager
etcd
Kubelet
Proxy
Worker
Docker
Pod
Pod
Kubelet
Proxy
Worker
Docker
Pod
Pod
Kubelet
Proxy
Worker
Docker
Pod
Pod


Kubernetes Client
Kubernetes Master
ETCD Cluster
Google Kubernetes
Architecture v1.x
by Yongbok Kim (ruo91)

_Kubernetes - Cluster

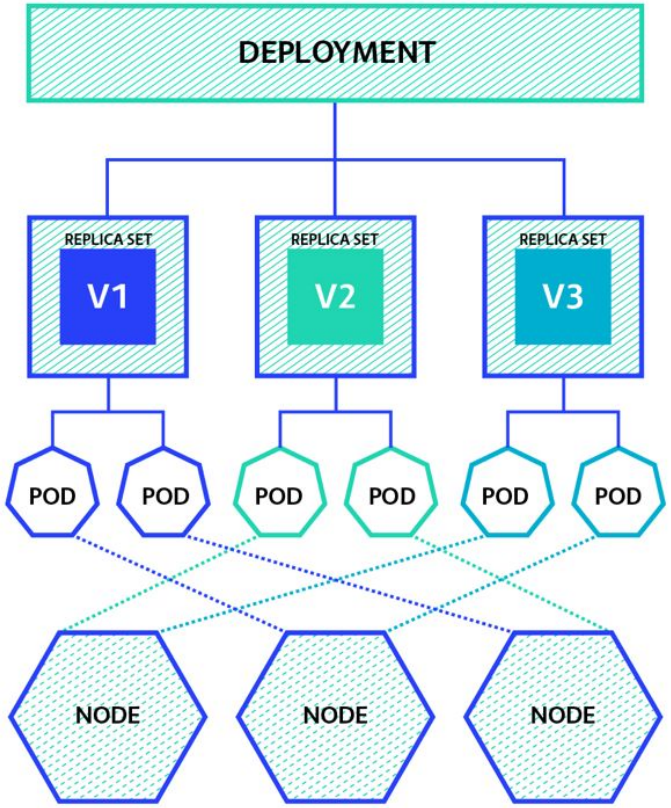
Componentes

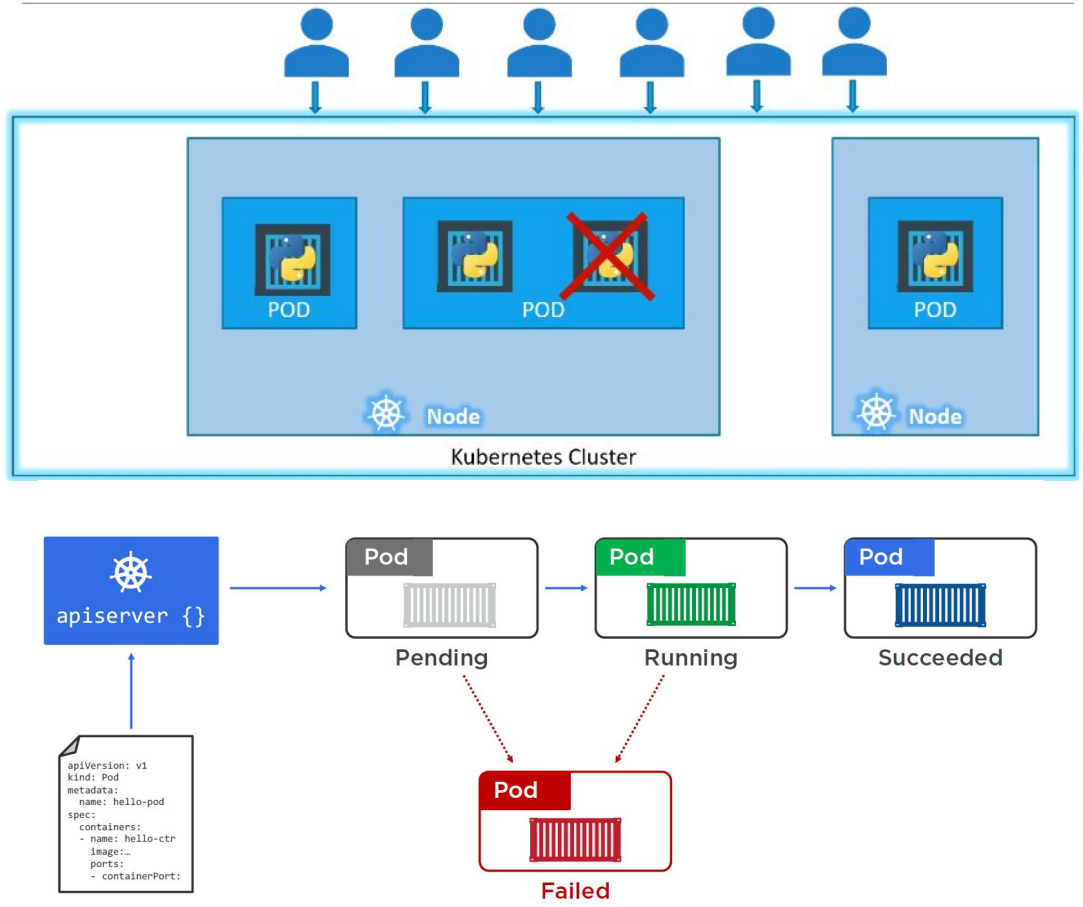
- **Master Node**: gerencia o cluster K8S e seus nós;
- **Worker Node**: onde os pods são executados;
- **API Server**: interface entre o usuário\aplicações comunicam com o K8S
- **Controller Manager**:
  - Responsável pela Orquestração;
  - Gerencia as mudanças;
  - Mantem o ambiente no estado configurado;
  - Controla o numero de replicas;
  - Gerencia os Endpoints.
- **Scheduler**:
  - Escuta o API Server em busca de novos Pods;
  - Aplica Pods para os Workes executarem.

Ferramentas

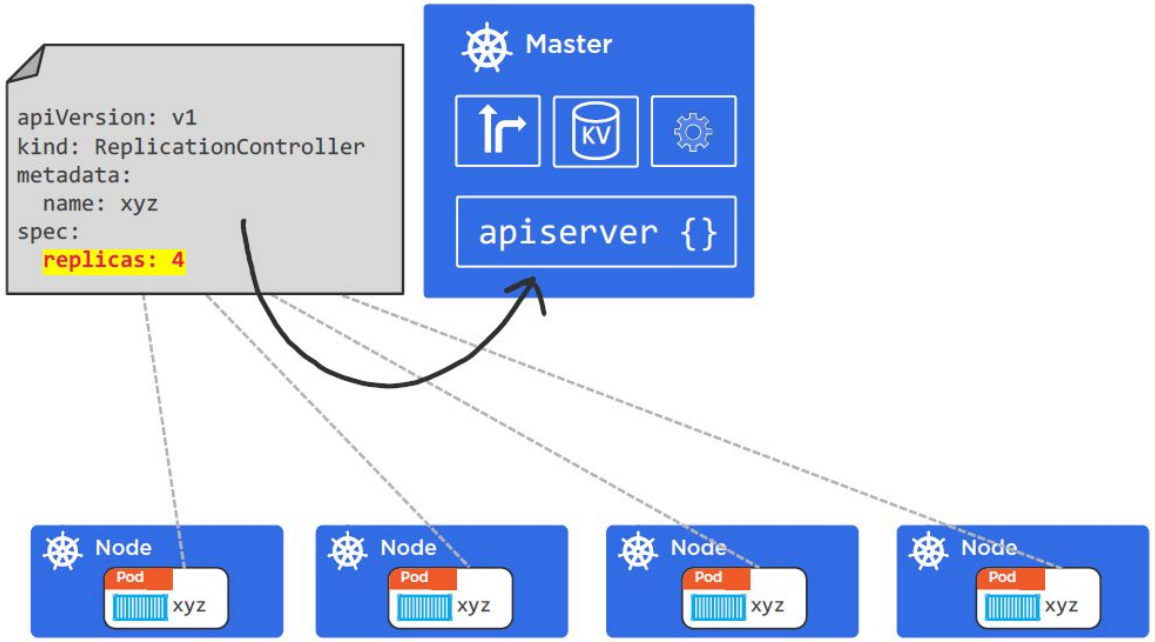
- **Kubectl**: interage com o nó master por meio de comandos;

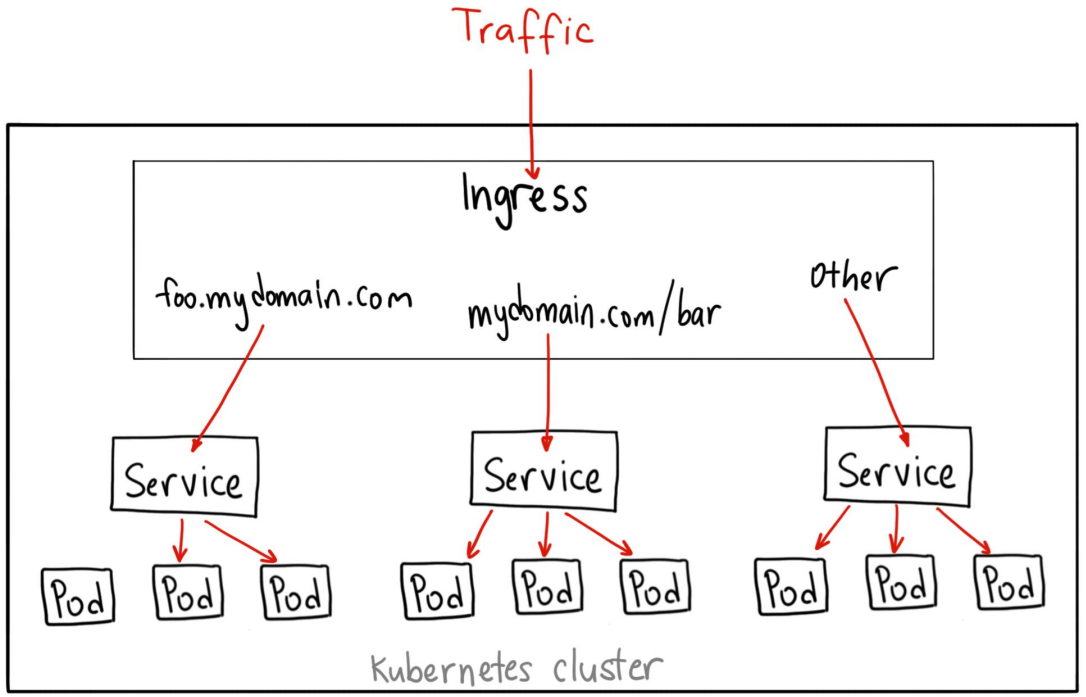
_Kubernetes – Replica Sets e Pods





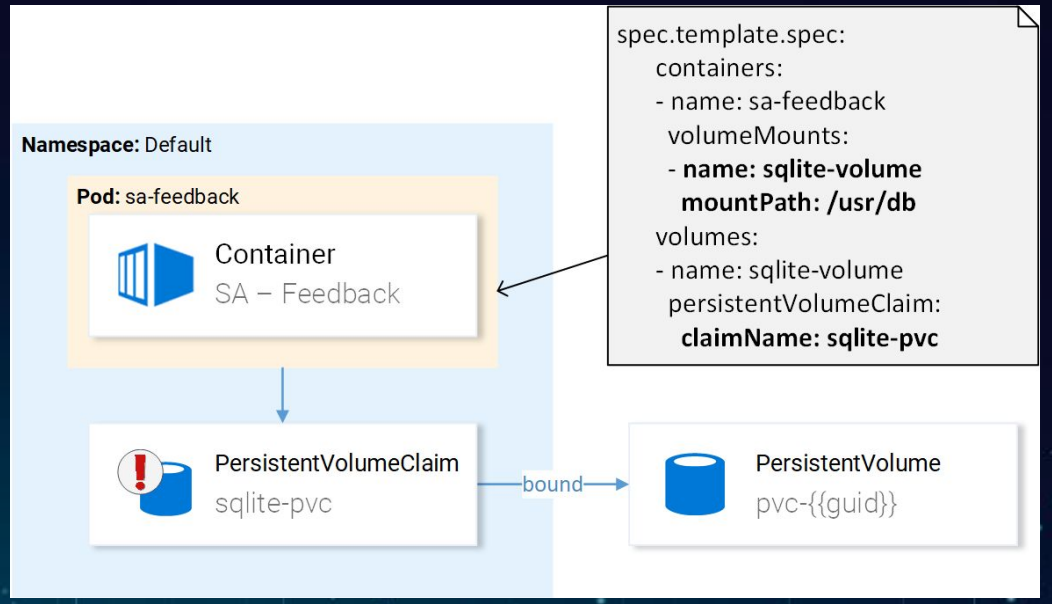
_Kubernetes – Deployments e Services





_Kubernetes - Volumes

- **Volumes:** armazenamento disponibilizado para um container (Docker) \ Pod (K8S);
- **PersistentVolumes**: volume que pode ser disponibilizado para um node \ pod;
- **PersistentVolumeClaims**: requisição de um usuário por um PersistentVolume;
- **StorageClass**: fornece uma forma de definir um perfil de storage para uma determinada necessidade de armazenamento.



_Kubernetes - Storage

Suporte para discos RAW (Raw Block Volume)

- AWSElasticBlockStore
- AzureDisk
- FC (Fibre Channel)
- GCEPersistentDisk
- iSCSI
- Local volume
- RBD (Ceph Block Device)
- VsphereVolume (alpha)

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: block-pv
spec:
  capacity:
    storage: 10Gi
  accessModes:
    - ReadWriteOnce
  volumeMode: Block
  persistentVolumeReclaimPolicy: Retain
  fc:
    targetWWNs: ["50060e801049cfd1"]
    lun: 0
    readOnly: false
```

DataTuning_

_Kubernetes - Storage

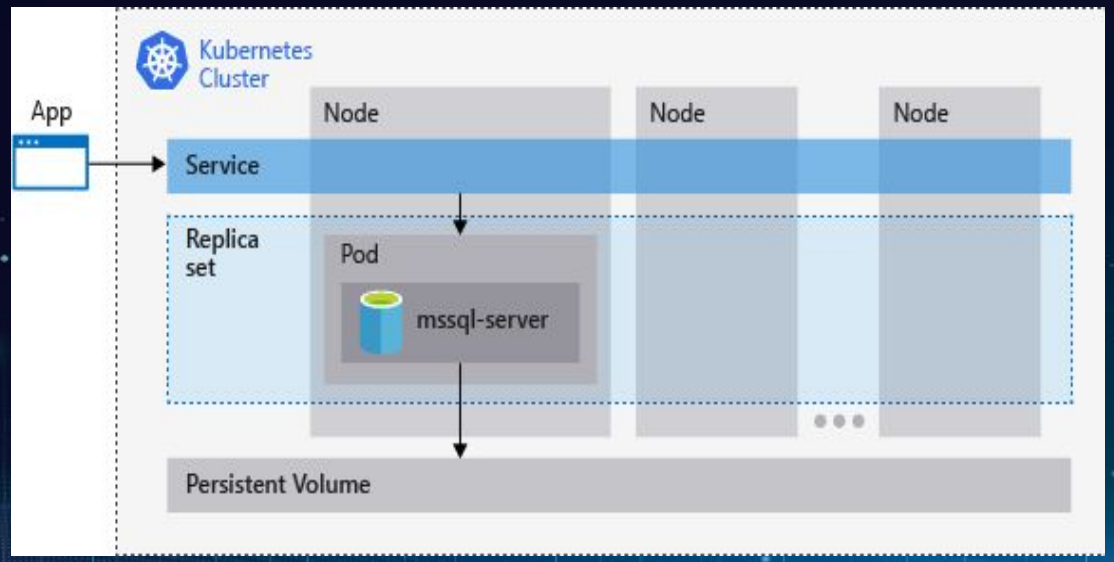
DataTuning_

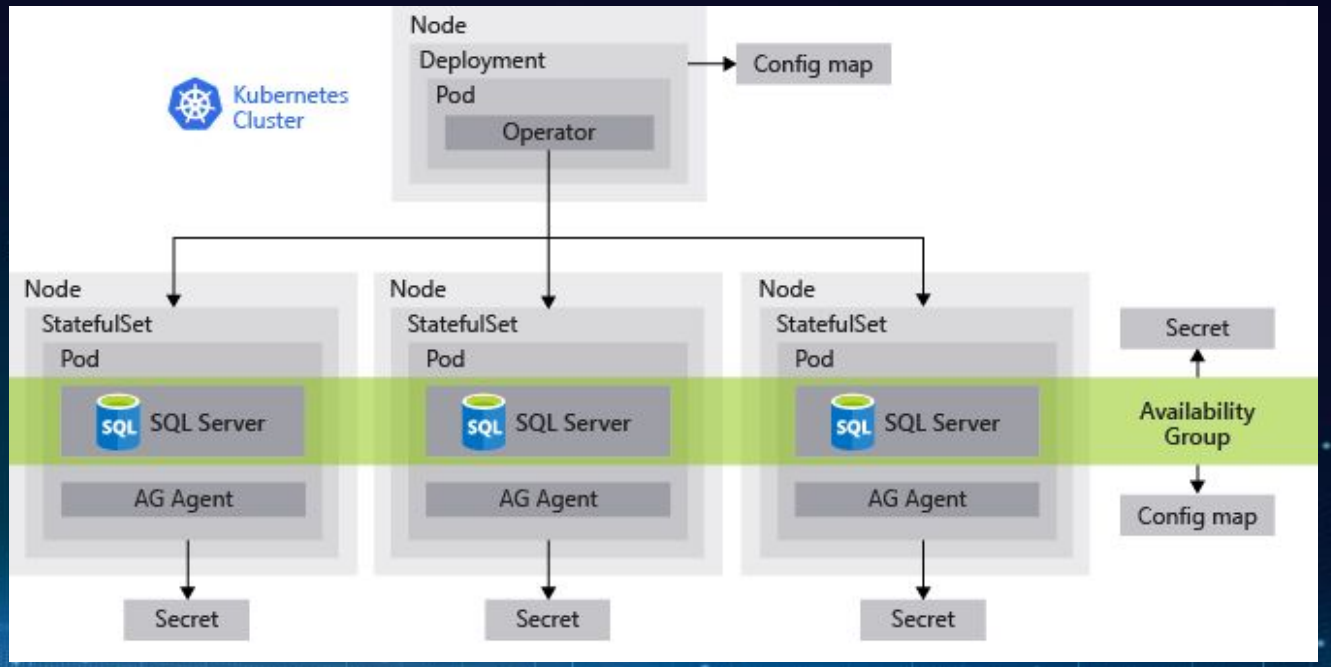
_Kubernetes - StatefulSets

- Em beta até a versão 1.9
- Importante para aplicações que precisam de:
  - Estabilidade utilizando identidades de rede únicas e storage persistente;
  - Implantação e escalabilidade graciosas e ordenadas;
  - Atualizações automatizadas e ordenadas.
- Parecido com Deployment porem tem série de particularidades como:
  - Pods com identidades únicas;
  - Precisa de um PersistentVolume já provisionado associado ao StatefulSet atrelado a um Storage Class;
  - Requere um Headless Service para ser responsável pelo service discovery e identidade de rede interna dos Pods (geralmente chamado de Operator);
  - https://kubernetes.io/docs/concepts/workloads/controllers/statefulset/

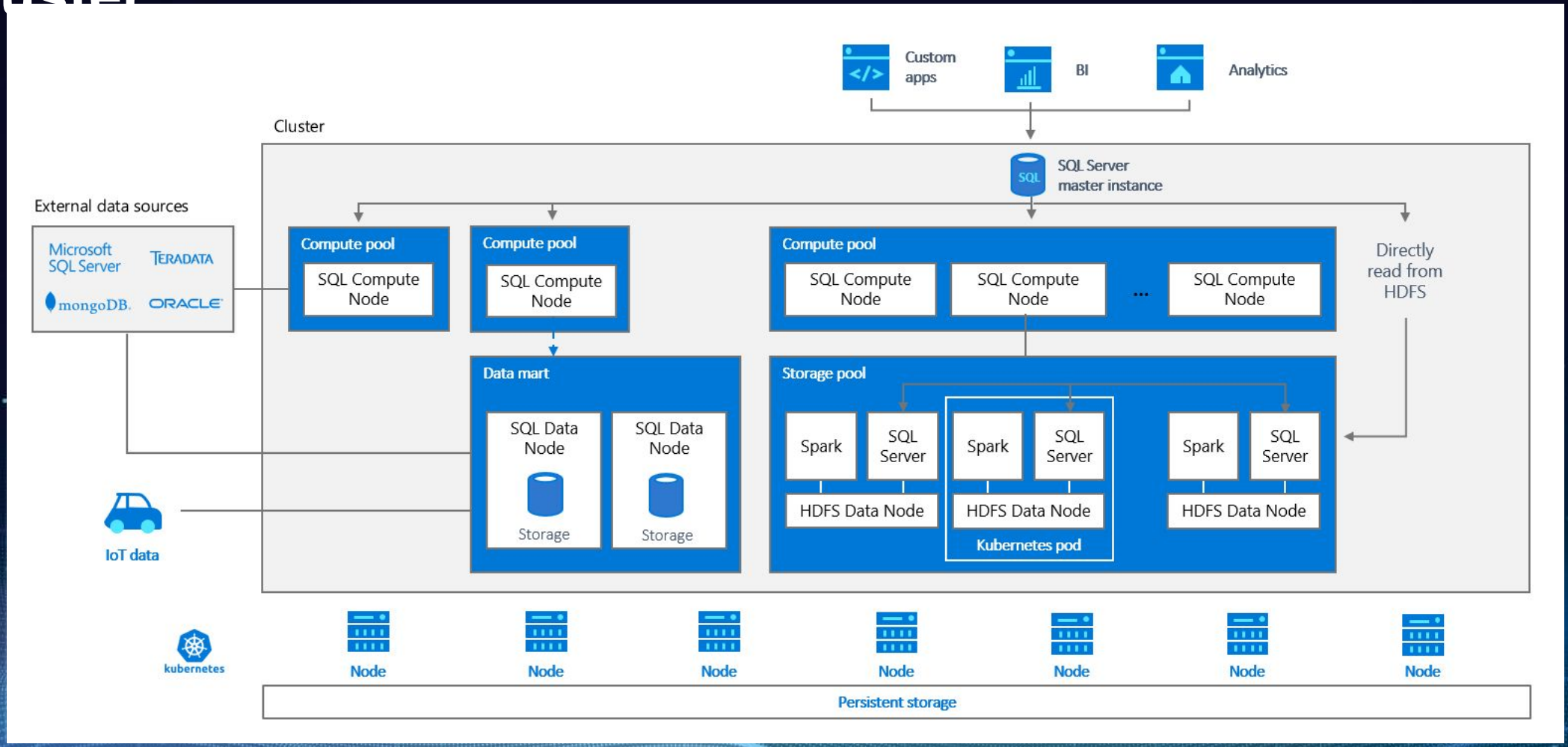
DataTuning_

_Casos de Uso – Docker e K8S

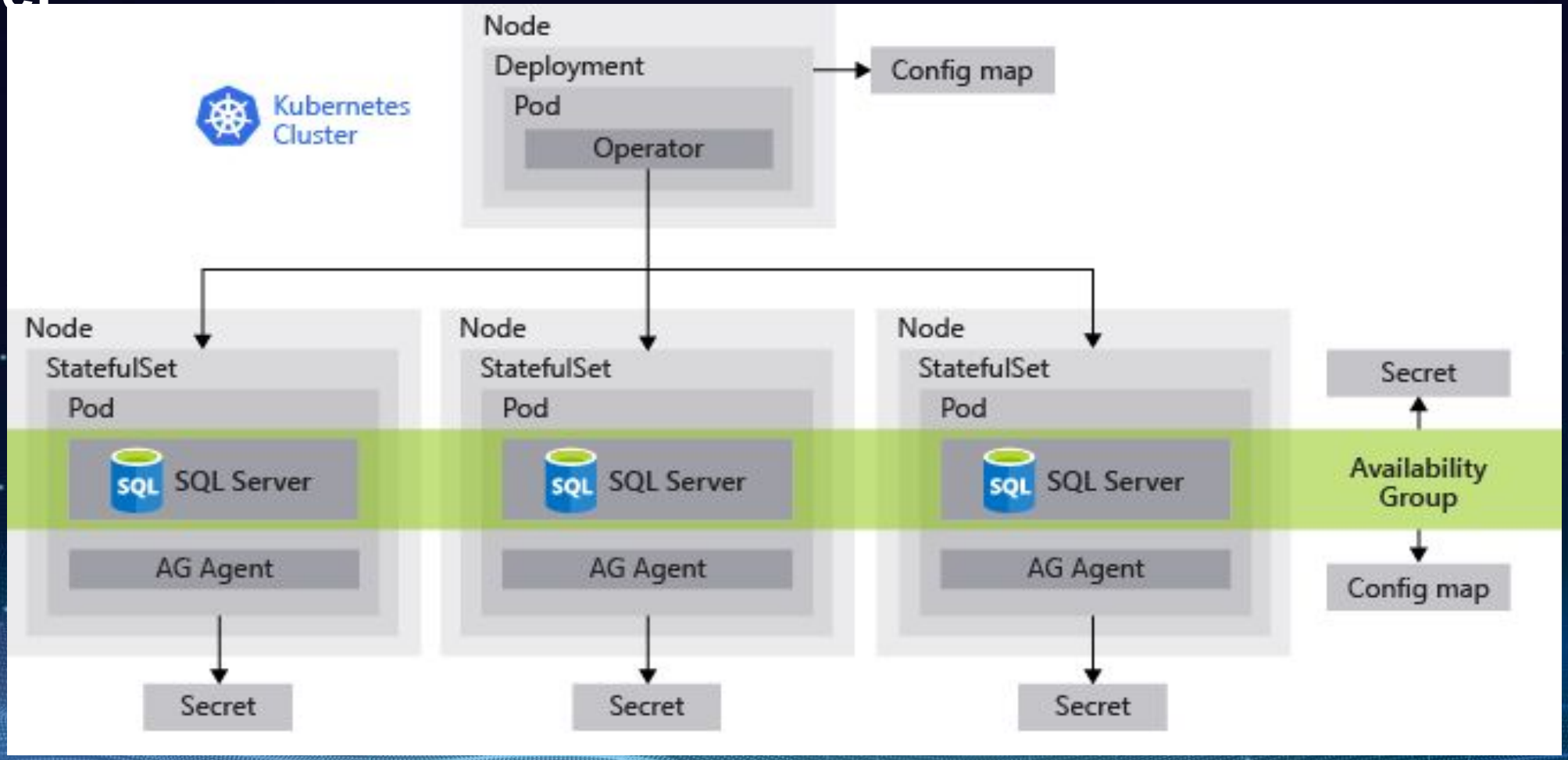




_Casos de Uso – Big Data Cluster



_Casos de Uso – Big Data Cluster



_Referencias

- https://medium.com/zero-to/setup-persistence-redis-cluster-in-kubertenes-7d5b7ffdbd98
- http://www.centinosystems.com/blog/sql/deploying-sql-server-in-kubernetes/
- https://docs.microsoft.com/pt-br/sql/linux/tutorial-sql-server-containers-kubernetes?view=sql-server-2017

_OBRIGADO!

DataTuning_