

DBA MASTERY



Explorando SQL Server
containers en Docker y
Kubernetes





Carlos Robles

Principal Consultant, DBA Mastery



/croblesdba



@dbamastery



crobles@dbamastery.com

Experiencia

Microsoft Data Platform MVP
Mas de 10 años de experiencia
DBA Multi plataforma

Comunidad

Speaker internacional, autor, blogger, mentor
Guatemala SQL Server community leader
Simple Talk, SQL Server Central and MSSQL Tips

DBA Mastery

SQL Server tips, scripts, mejores practicas



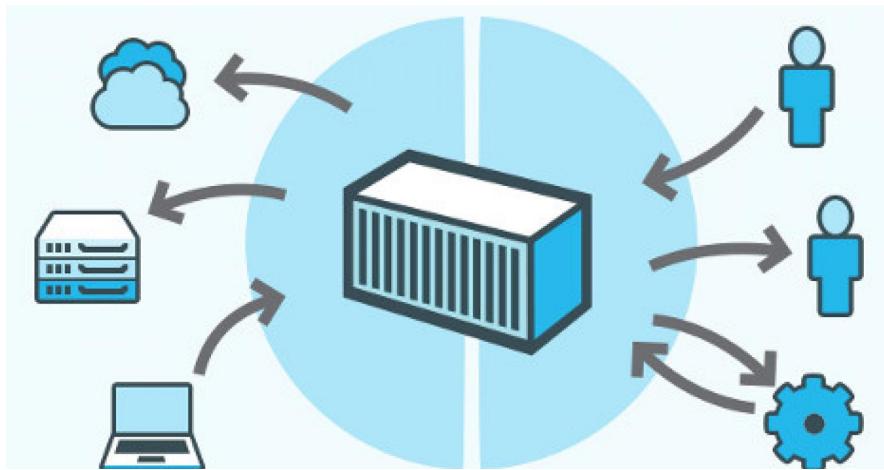
MAXDOP Calculator

Azure Data Studio wait stats widget
PerfMon para DBA's
MSDB tuning

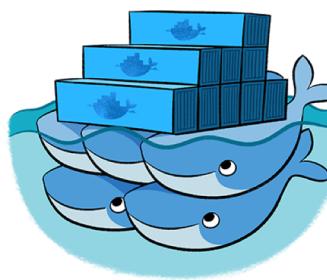
AGENDA

- Introducción a Docker
 - Definicion y componentes
 - Arquitectura de Docker
 - VM's vs Containers
 - Imagen de SQL Server
 - SQL Server Dockerfile
 - Ejecutando un SQL container
 - Administrando containers
 - Casos de uso
 - Ventajas
 - Demo
-
- Introduction a Kubernetes
 - Definicion y componentes
 - Arquitectura de Kubernetes
 - Demo





Containers son el futuro!



DOCKER



- Que es docker?

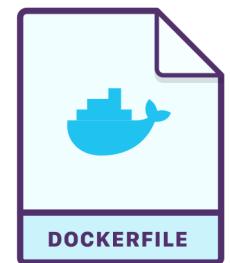
Docker es una plataforma abierta que permite el desarrollo, distribucion y ejecucion de aplicaciones. Docker permite separar las aplicaciones de la infraestructura, lo cual agiliza el tiempo de entrega del software.

Con Docker, la infraestructura se maneja de la misma manera que se manejan las aplicaciones.



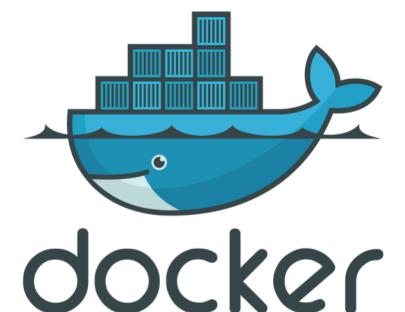
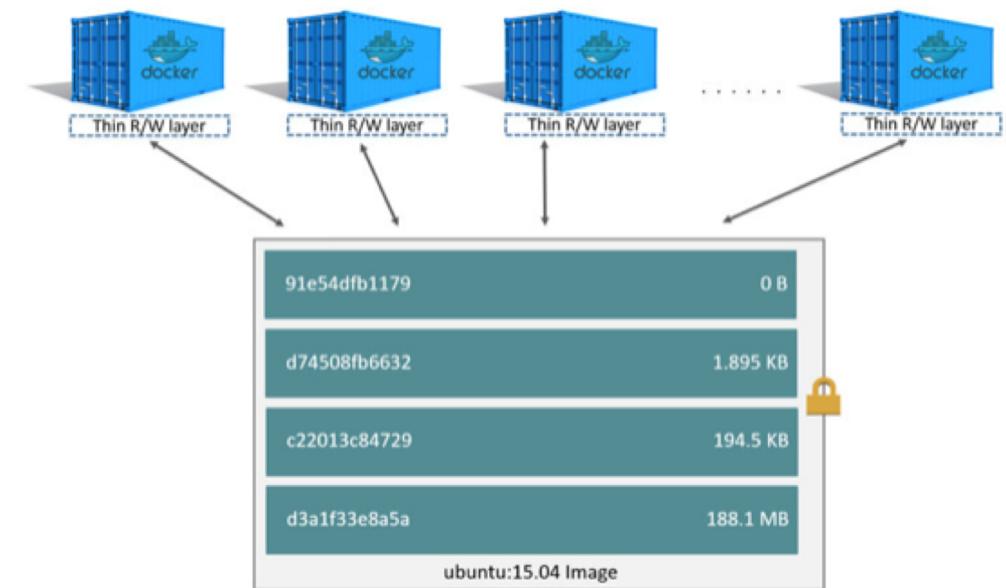
IMAGENES

- Una plantilla de lectura, que contiene instrucciones para crear un container
- Imagenes son creadas usando Dockerfile
- Es un snapshot de un conjunto de archivos necesarios para la applicacion
- Portable y consistente
- Una nueva imagen puede ser creada de una ya existente
 - SQL Server por ejemplo, es basada en Ubuntu o RedHat

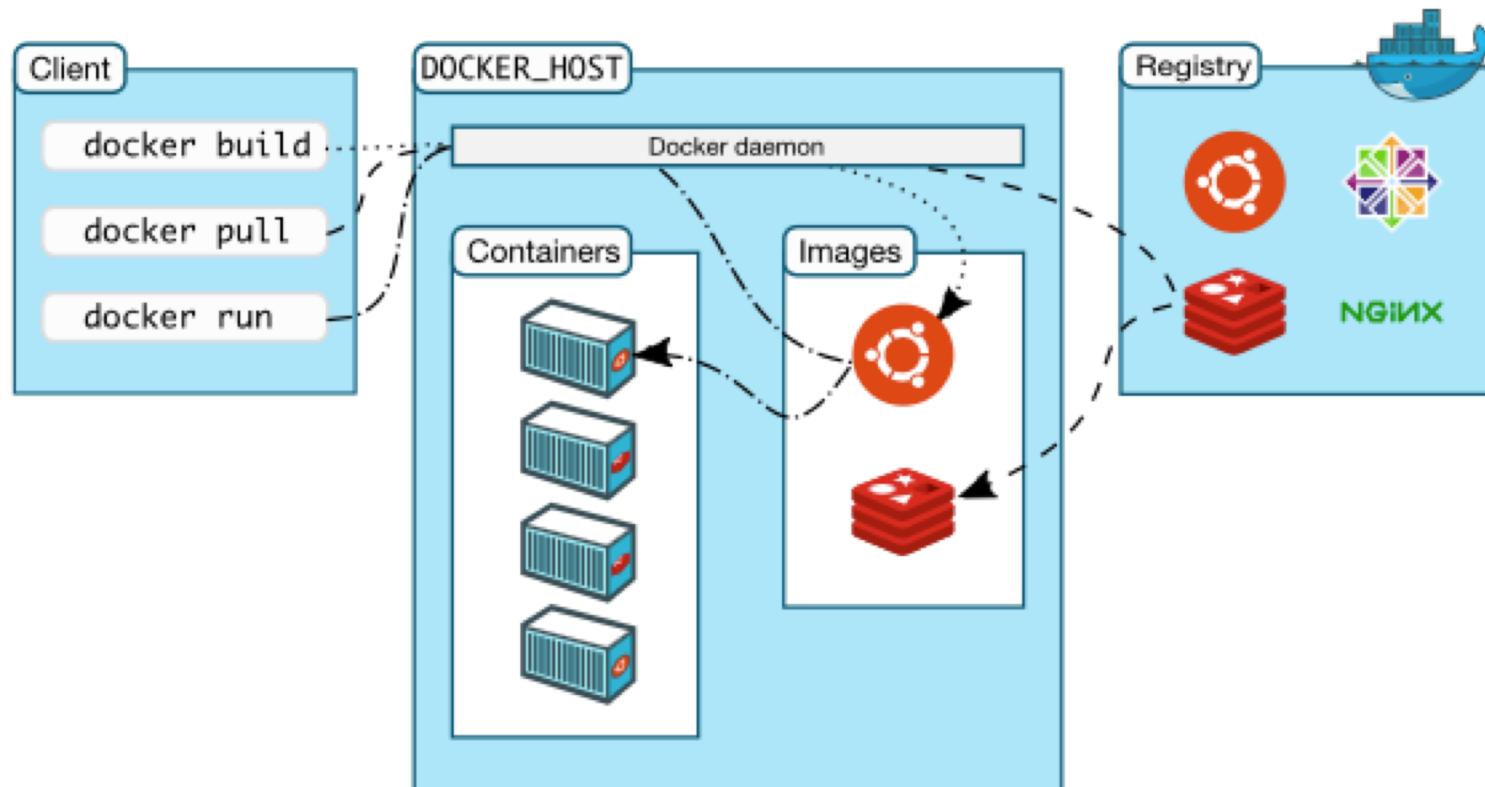


CONTAINERS

- Es la instancia ejecutable de una imagen
- Es una unidad standard de software
- Containers no son mas que un programa
- Capa de escritura y capa compartida de lectura
- Containers tienen acceso a todos los recursos de la maquina host
- Volumenes = Almacenamiento persistente



ARCHITECTURA DE DOCKER



VM'S VS CONTAINERS



-
- Virtualization +15 años
 - A veces muy pesadas
 - Virtualización de hardware
 - Cada VM tiene un OS



-
- No requiere instalación
 - Ligeras
 - Virtualización OS
 - Todos los containers se ejecutan en la misma maquina



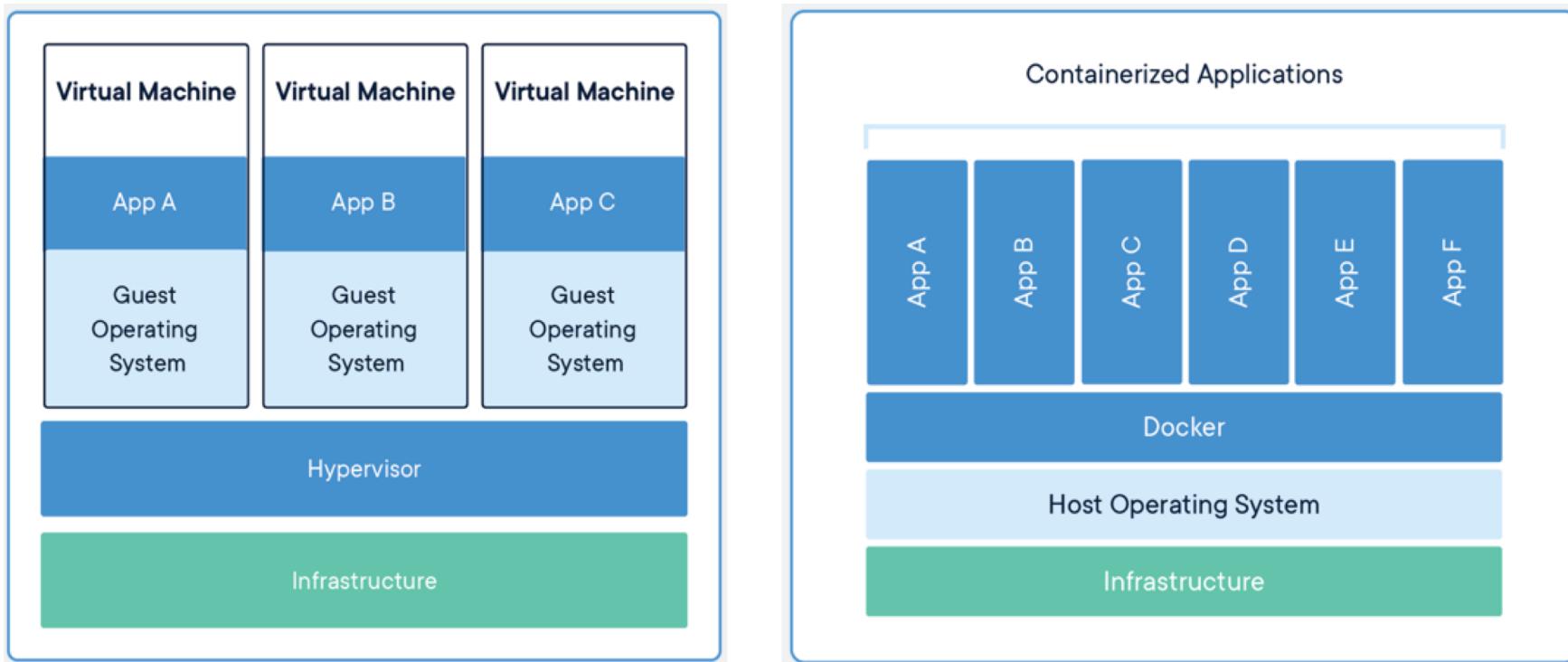
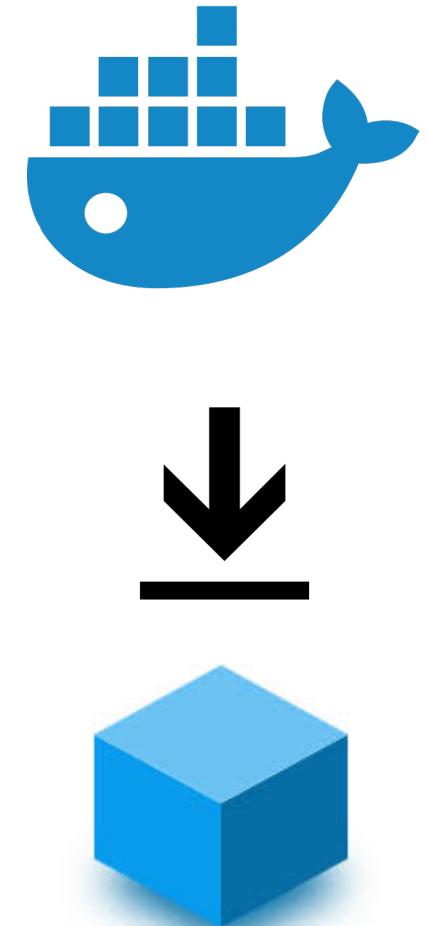


IMAGEN DE SQL SERVER

- [Docker Hub](#) – [Microsoft container registry](#)
- SQL Server 2017
 - Solo Ubuntu desde RTM al ultimo CU y GDR
- SQL Server 2019 (CTP)
 - Ubuntu y RedHat
 - Desde CTP 1.0 al ultimo
- SQL Server esta pre-instalado (standard)
- Backups son compatibles – multi plataforma



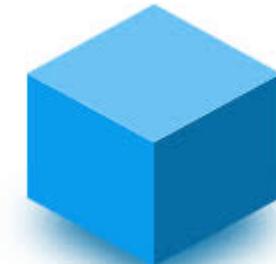
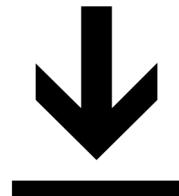
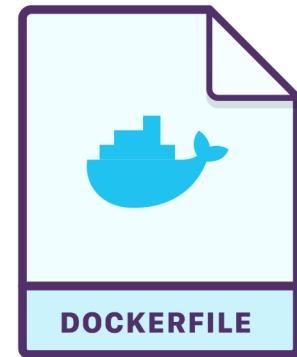
SQL SERVER DOCKERFILE

```
FROM ubuntu:16.04
```

```
EXPOSE 1433
```

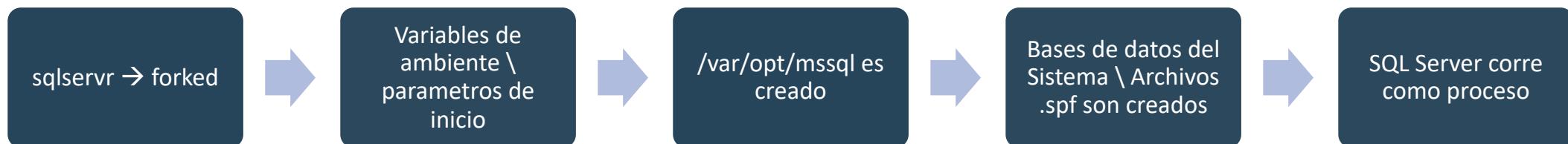
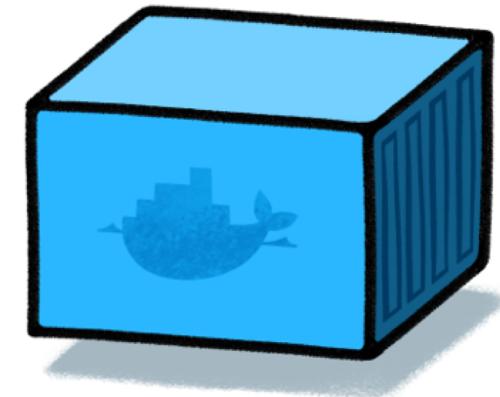
```
COPY ./install /
```

```
CMD ["/opt/mssql/bin/sqlservr"]
```



EXECUTANDO UN SQL CONTAINER

```
docker run \
--name SQLDay_Coban \
--env 'ACCEPT_EULA=Y' \
--env 'MSSQL_SA_PASSWORD=Sq1D4yC0b@n' \
--publish 1400:1433 \
--detach mcr.microsoft.com/mssql/server:2017-CU15-ubuntu
```



ADMINISTRANDO CONTAINERS

docker pull

docker run

docker start | stop

docker image | container

docker rm | rmi

docker exec

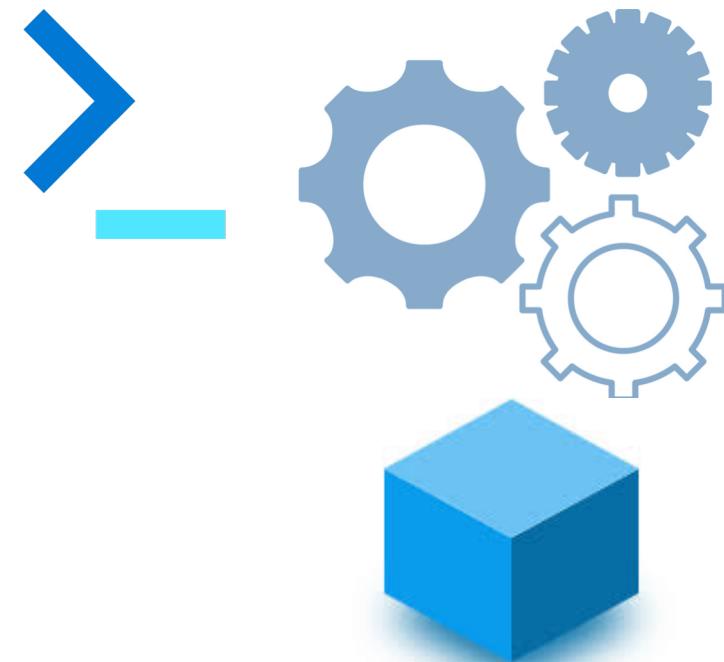
docker build

docker logs

docker inspect

docker volume

docker save



Preguntas ?

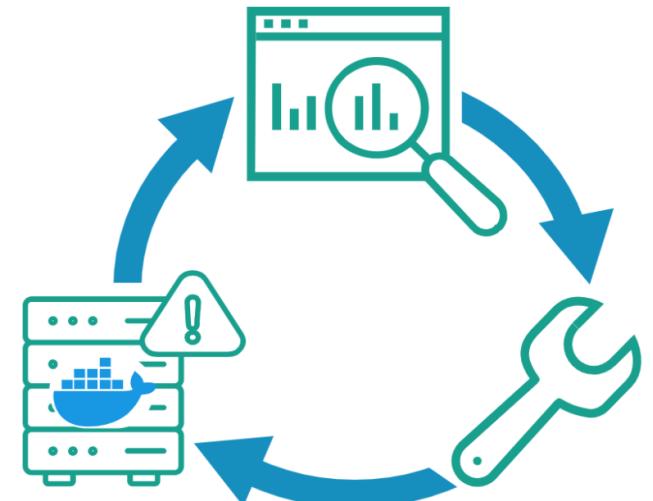


DEMO



CASOS DE USO

- Desarrollo local
- Troubleshooting
- Demostraciones
- Elimina ambientes compartidos
- Elimina problemas de recursos
- Ambientes temporales
 - No requiere instalación \ parcheo



KUBERNETES



kubernetes

- Que es Kubernetes?

Kubernetes es una plataforma extensible de tipo open source, que nos permite manejar containers y sus cargas de trabajo o servicios de manera sencilla travez de configuraciones declarativas y automatización.



COMPONENTES

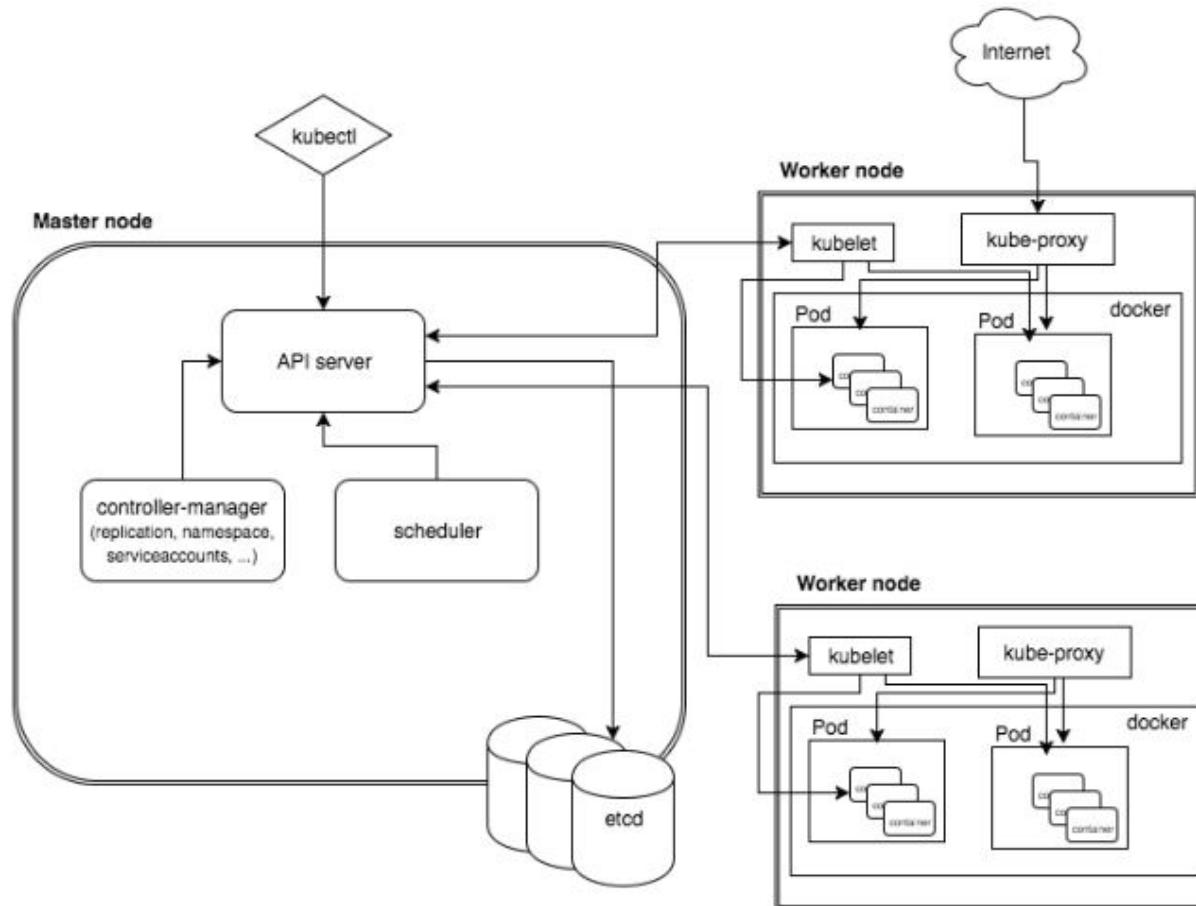
- Masters
 - Multiples procesos \ partes moviendose
 - Se ejecuta en solo un nodo
 - Le dice a otros que hacer (desired state)
- Nodes
 - Hacen el trabajo, corren las aplicaciones
 - Antes llamados “minions”
 - Reportan estado actual al master



- Pods
 - Containers se ejecutan dentro de los pods
 - Se puede tener uno o mas containers
- Servicios
 - DNS \ IP
- Implementaciones
 - Modelo declarativo
 - Desired state (cantidad de POD's)
 - Manifest file (YAML, JSON)



ARQUITECTURA DE KUBERNETES



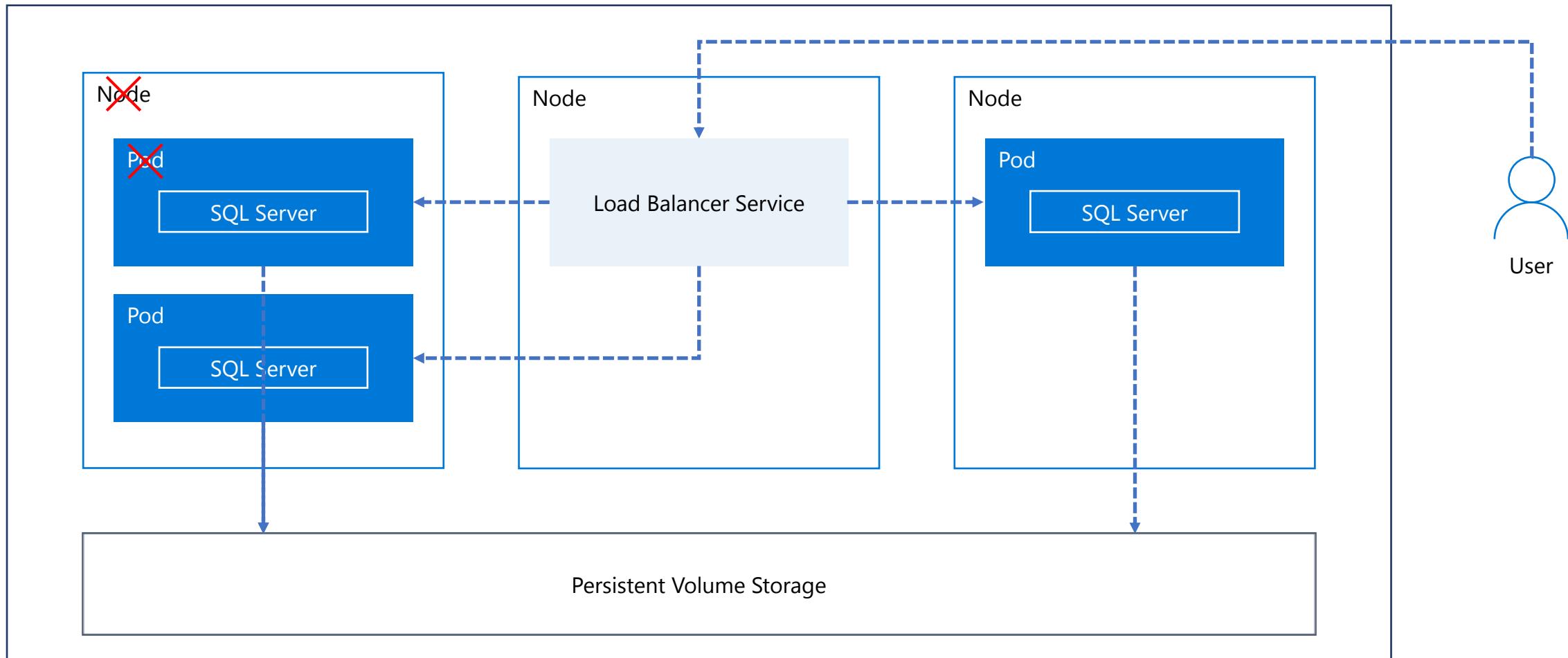
Preguntas ?



DEMO



Master

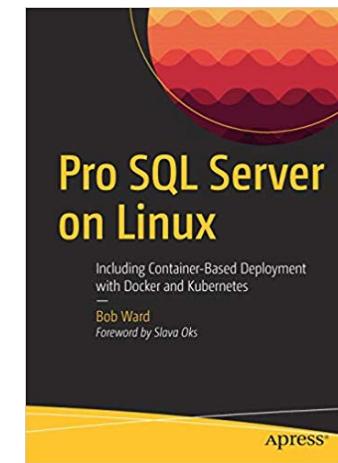


Preguntas ?



REFERENCIAS

- Official documentation
 - [Docker Docs](#)
 - [Kubernetes Docs](#)
- SQL Server
 - [SQL Server workshops](#)
 - [SQL Server samples](#)
- Books
 - [Docker Deep dive](#)
 - [The Docker book](#)
 - [Kubernetes: Up and Running](#)
 - [The Kubernetes book](#)
 - [Pro SQL Server On Linux by Bob Ward](#)
- Pluralsight Courses
 - [Getting Started with Docker](#)
 - [Docker Deep Dive](#)
 - [Docker and Kubernetes: The big picture](#)
 - [Kubernetes Installation and Configuration fundamentals](#)
- Microsoft Learning Courses
 - [Kubernetes Learning Path](#)
 - [SQL Workshops](#)
- Katacoda
 - [Docker](#)
 - [Kubernetes](#)



MAS CONTENIDO

- **24 Hours of PASS**
 - [YouTube recording](#)
- **Simple Talk**
 - [SQL Server Docker Containers in macOS](#)
- **SQL Server Central**
 - [Creating Aliases for Docker commands](#)
 - [Managing SQL Server containers using Python – Part 1](#)
 - Managing SQL Server containers using Python – Part 2 (**Coming soon**)
- **MSSQL Tips**
 - [SQL Server 2019 CT2 RHEL Docker containers](#)





croblesdba



@dbamastery



crobles@dbamastery.com



DBA Mastery



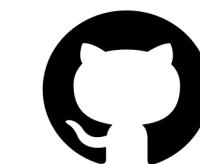
carlos-lopez-taks



@CarlosLopezSQL



caltls@gmail.com



Muppiy

