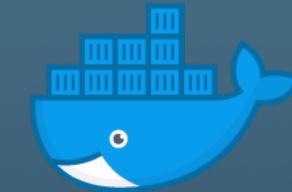


DBA MASTERY



Working with SQL Server
containers on Docker





Carlos Robles

Principal Consultant, DBA Mastery



/croblesdba



@dbamastery



crobles@dbamastery.com

Experience

Microsoft Data Platform MVP
Over 10 years of experience
Multi platform DBA

Community

International speaker, author, blogger, mentor
Guatemala SQL Server community leader
Simple Talk, SQL Server Central and MSSQL Tips
author

DBA Mastery

SQL Server tips, scripts, best practices and more



MAXDOP Calculator

Azure Data Studio wait stats widget

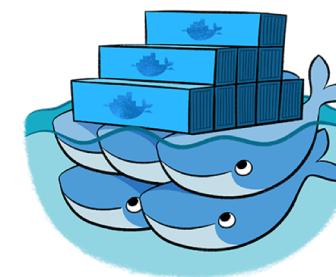
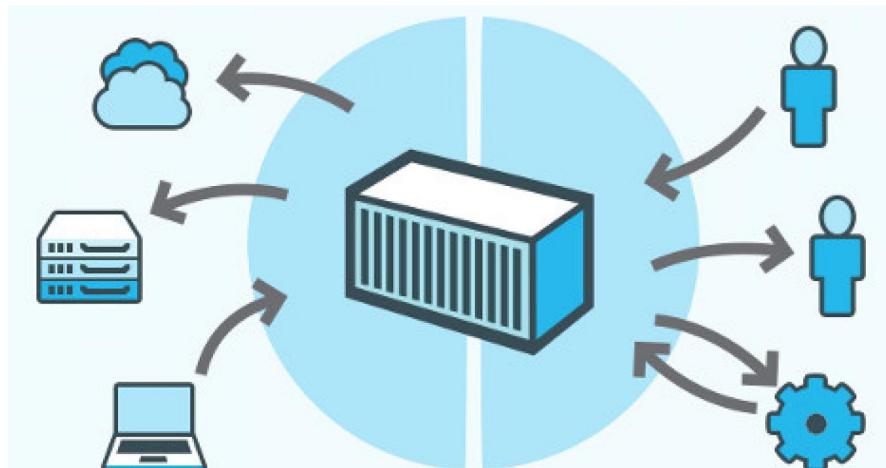
PerfMon for DBA's

MSDB tuning

AGENDA

- Introduction to Docker
 - Definition and components
 - Architecture
 - VMs vs Containers
- The SQL Server docker image
- The SQL Server Dockerfile
- Running a SQL Server container
 - Managing containers
 - Demo





Build, ship and deploy



DOCKER



- From Docker docs:

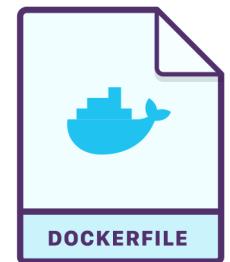
Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly.

With Docker, you can manage your infrastructure in the same ways you manage your applications.



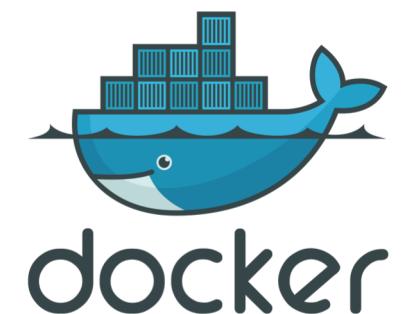
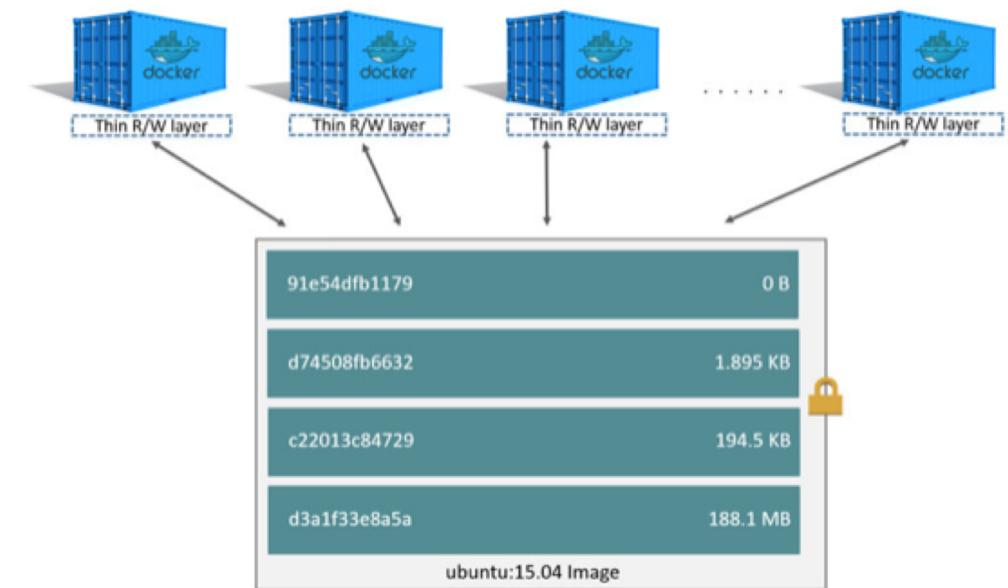
IMAGES

- Is a read-only template with instructions for creating a Docker container
- Images are created using a Dockerfile
- A snapshot of a set of files required to run an application (portable)
- A new image can be created from an existing image (make your own)
 - SQL Server for example, based on Ubuntu or RedHat

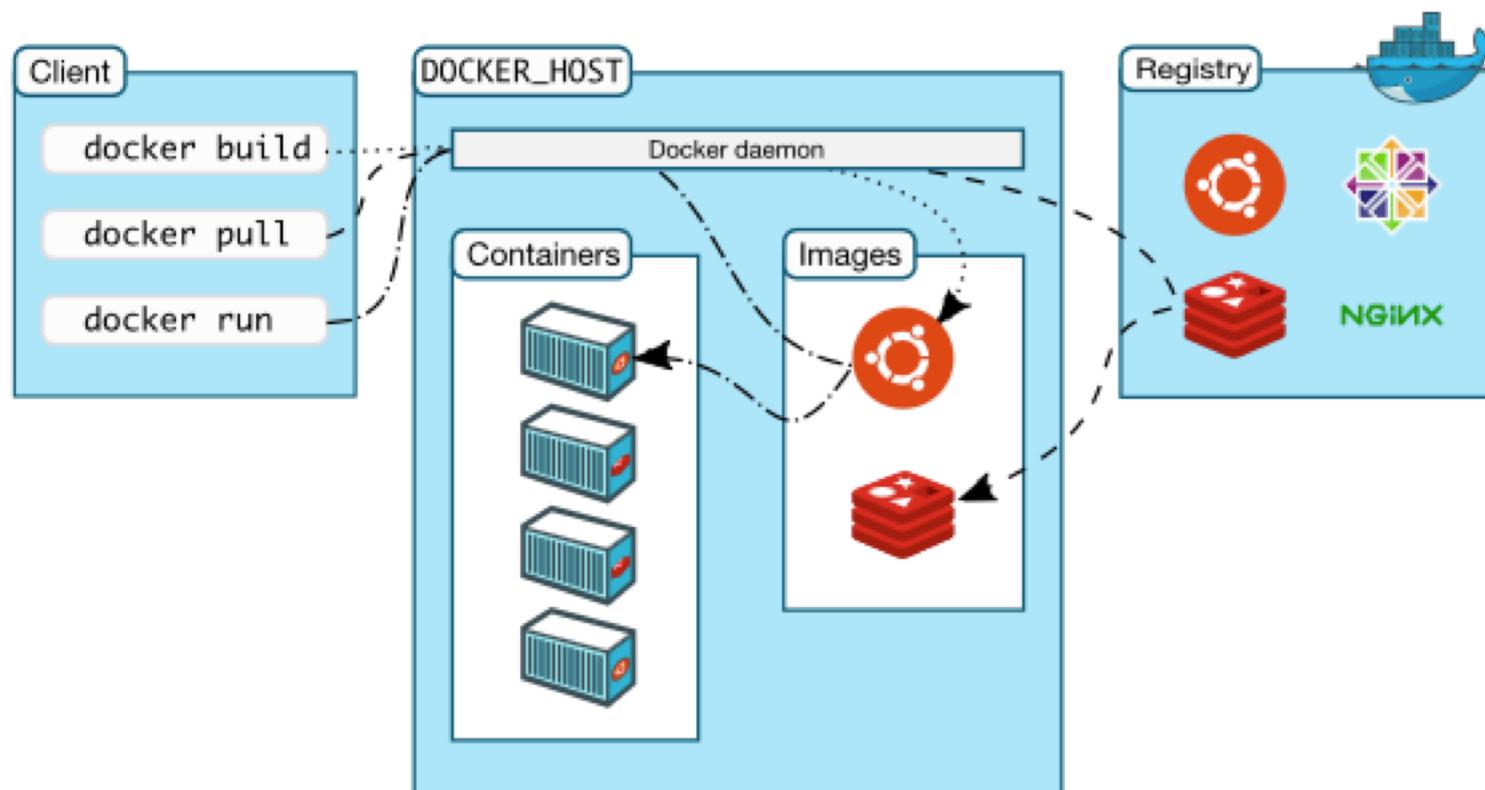


CONTAINERS

- The runnable instance of a Docker image
- Container is nothing more than a program
- Containers has full access to all resources
- Writable layer and shared read-only layer
 - Small storage footprint
 - Volumes = Persistent storage



DOCKER ARCHITECTURE



VM'S VS CONTAINERS

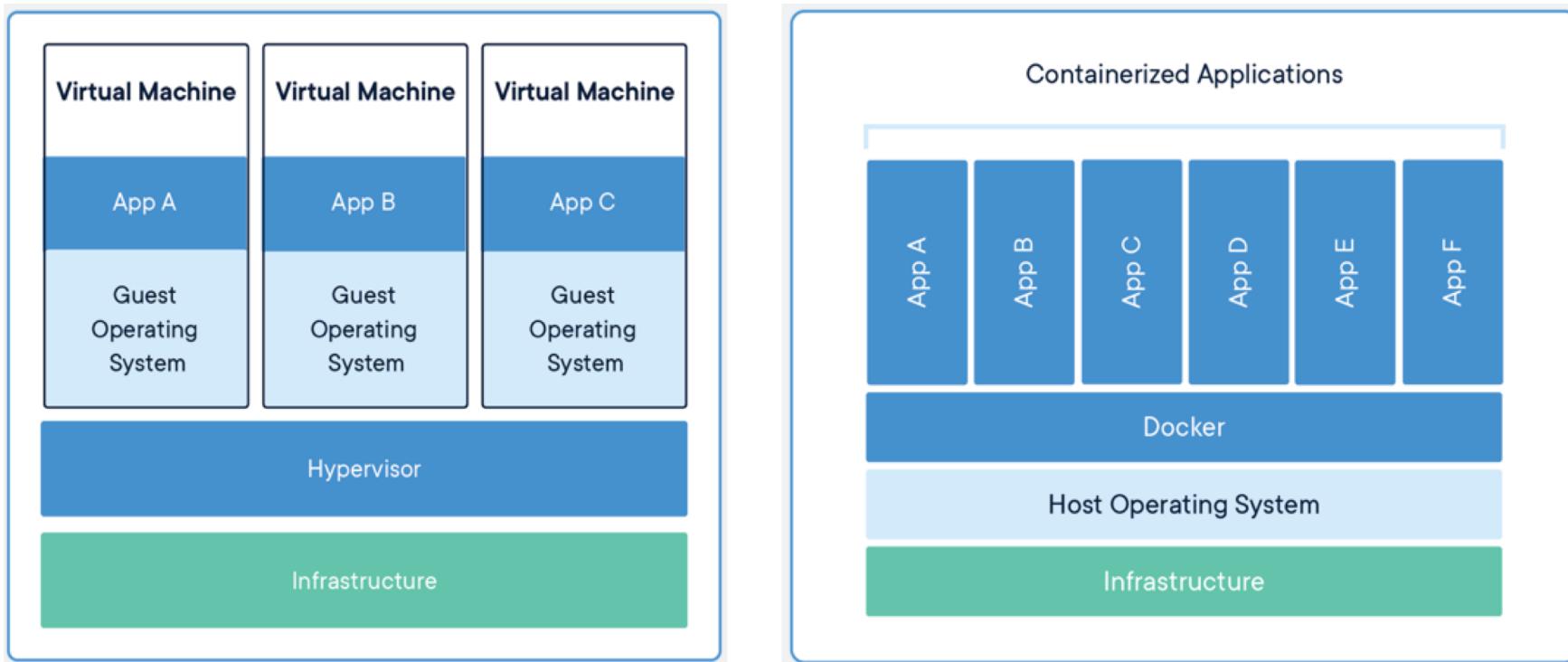


-
- Virtualization +15 years
 - Sometimes heavyweight
 - Hardware virtualization
 - Each VM has an entire OS



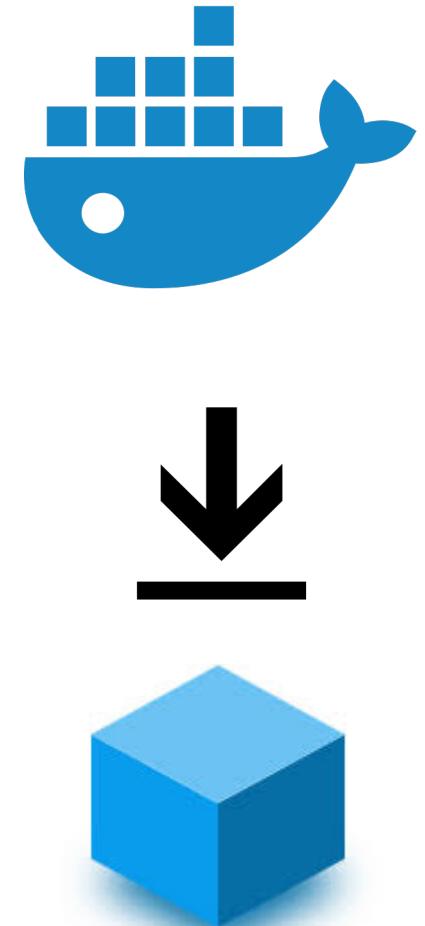
-
- No installation
 - Lightweight
 - OS virtualization
 - All containers run in the same host OS





SQL SERVER IMAGE

- [Docker Hub – Microsoft container registry](#)
- SQL Server 2017
 - Just Ubuntu from RTM to latest CU
- SQL Server 2019 (RC)
 - Ubuntu and RedHat
 - From CTP to latest RC



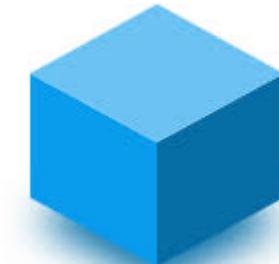
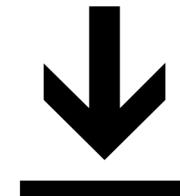
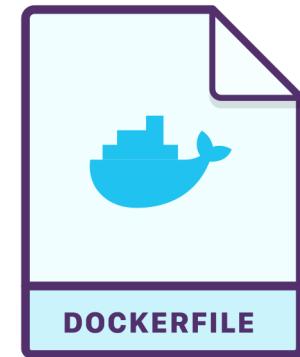
SQL SERVER DOCKERFILE

```
FROM ubuntu:16.04
```

```
EXPOSE 1433
```

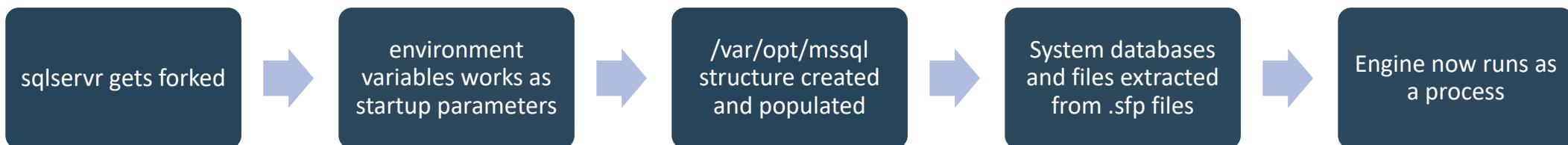
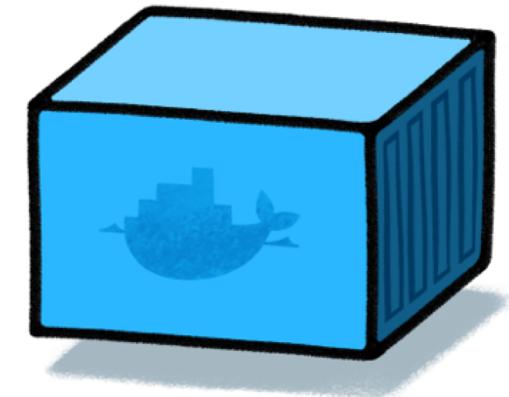
```
COPY ./install /
```

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



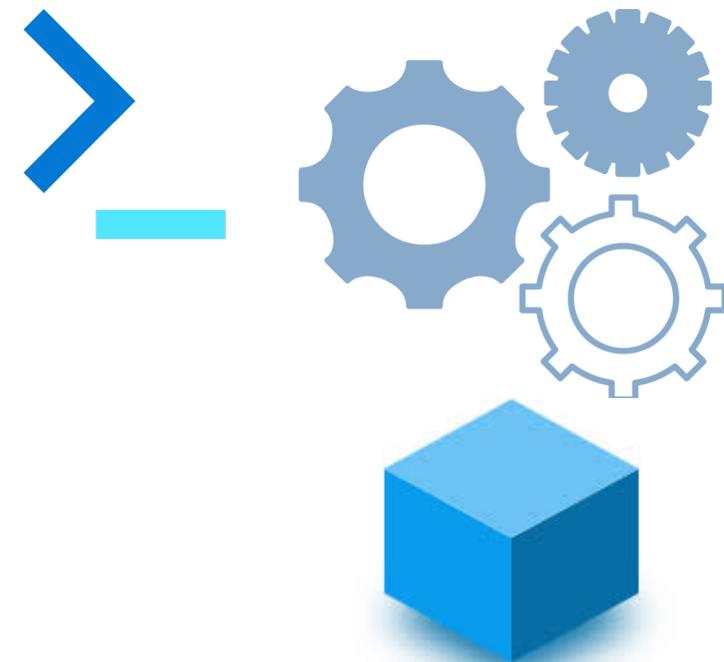
RUNNING A SQL CONTAINER

```
docker run \
--name SQLSat912 \
--env 'ACCEPT_EULA=Y' \
--env 'MSSQL_SA_PASSWORD=N3wY0rR0cks' \
--publish 1400:1433 \
--detach mcr.microsoft.com/mssql/server:2017-CU16-ubuntu
```



MANAGING 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

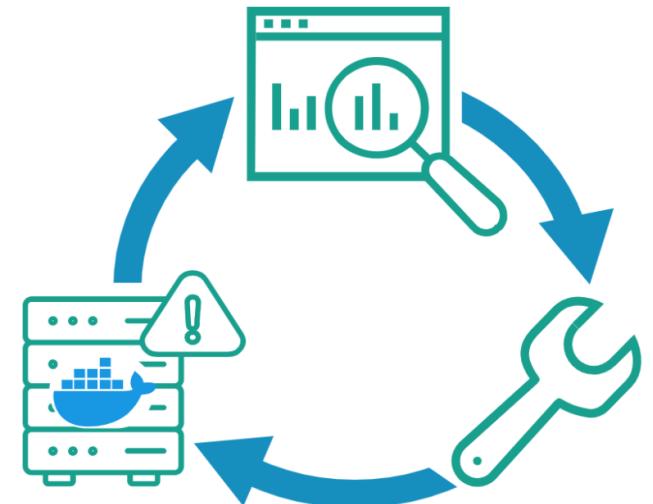


DEMO



USE CASES

- Local development
- Troubleshooting
- Demonstrations
- Eliminates shared environments
- Eliminates resource contention
- Temporal environments
 - No installation \ patching



KUBERNETES



kubernetes

- From Kubernetes docs:

Kubernetes is a portable, extensible open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation.

It also make possible the container orchestration for automating application deployment, scaling, and management.



BASIC CONCEPTS

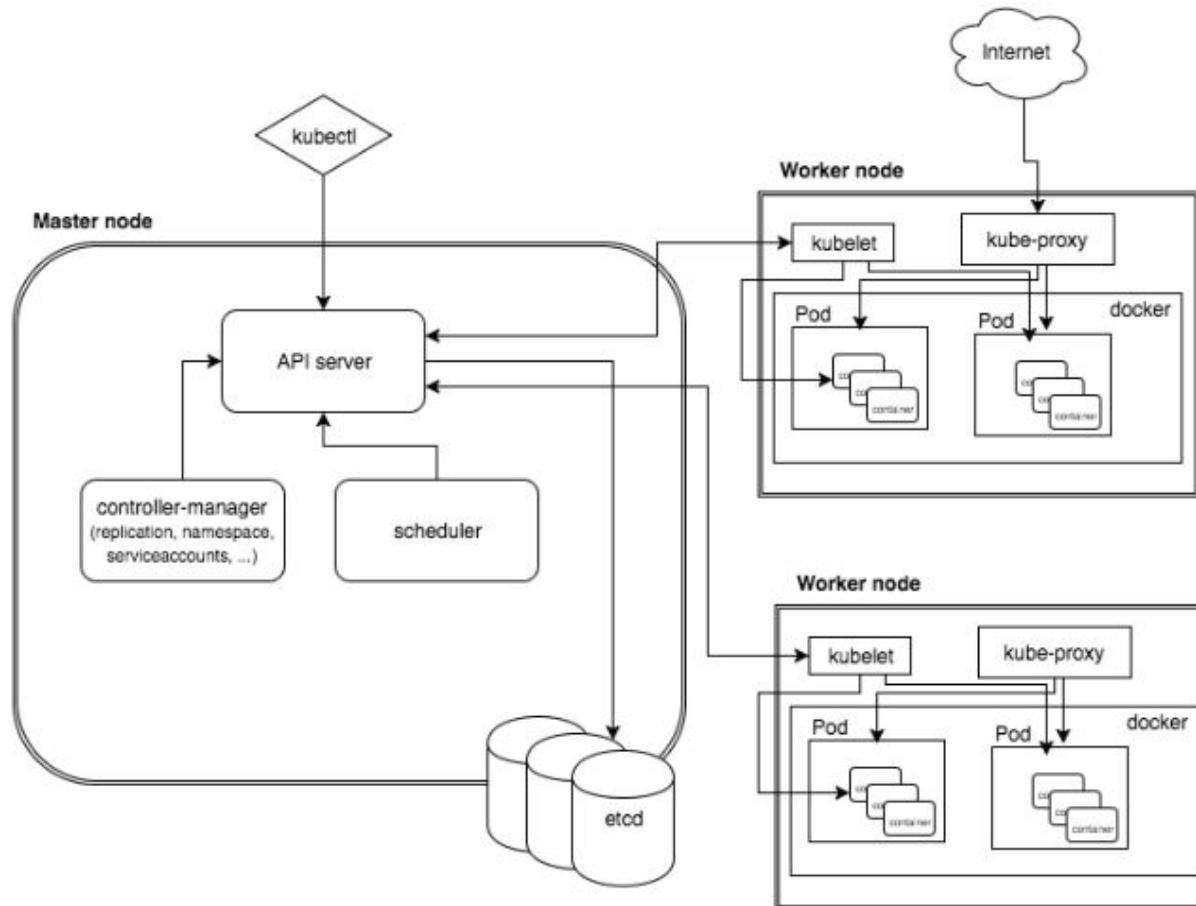
- Masters
 - Multiple moving parts \ processes
 - Runs on a single node in the cluster
 - Tells what to do – desired state
- Nodes
 - Do the work, runs applications
 - Aka “minions”
 - Reports the state back up to the master



- Pods
 - Containers runs inside of pods
 - Can have one or more pods within a node
- Services
 - Hiding multiple pods behind a service IP address
- Deployments
 - Declarative model
 - Desired state (number of POD's)
 - Manifest file (YAML, JSON)



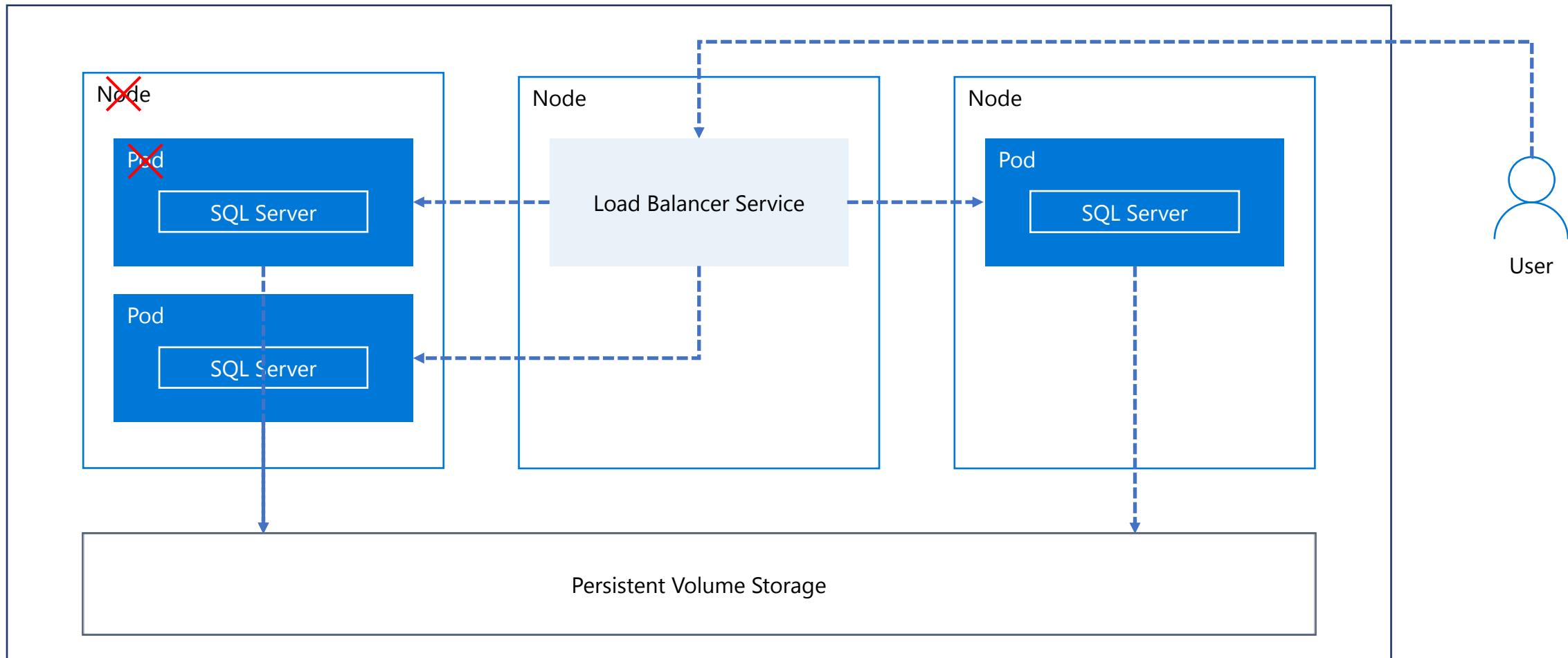
KUBERNETES ARCHITECTURE



DEMO



Master



QUESTIONS ?



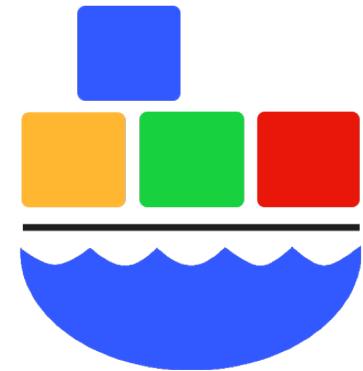
WHAT ABOUT WINDOWS CONTAINERS?

Microsoft: SQL 2016 to 2019 Dev editions

- “Not for production use” per Microsoft
- Last SQL 2016, 2017 image update <18 months ago
- No Windows Authentication
- Kubernetes support coming with Windows Server 2019

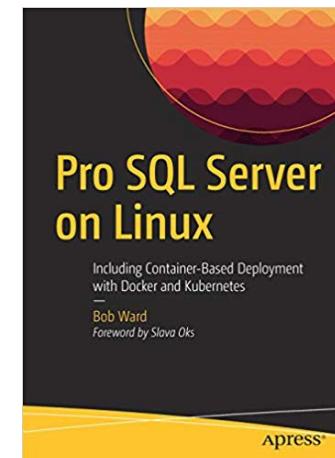
Windocks: All editions/releases SQL 2008 onward

- Containers = named instances w/Windows Authentication
- Containers are free as named instances under existing SQL licenses
- Supports DB engine, SSRS, & Linked Servers
- DB cloning with “incremental forever” updates with Diff or Transaction Log backups supports SQL Server instances and containers (MS Linux containers too)
- DevOps, CI, and SSRS for AWS RDS or SQL Azure are popular uses
- Kubernetes support available now
- Prices start at \$499/month



REFERENCES

- **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](#)



MORE FROM CARLOS

- **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](#)
 - **MSSQL Tips**
 - [SQL Server 2019 CT2 RHEL Docker containers](#)
 - [Getting started with SQL Server Azure Container Instances](#)
 - **PASS Summit 2019**
 - [Working SQL containers to build database solutions](#)
- 





/croblesdba



@dbamastery



crobles@dbamastery.com



DBA Mastery



Thank you
New York!!

