PERCONA

From LAMP Stack to Kube - Moving Your Old Websites into the Cloud Without Leaving Chemical Trails

PERCONA

Databases run better with Percona

Dave Stokes
@Stoker
David.Stokes@Percona.com

Presentation

The LAMP - Linux, Apache, MySQL and PHP/Perl stack was the internet model for so many years. But now everything is 'cloud this' or 'Kubernetes that'.

But how do you move from LAMP to this new medium, how do the various 'tinker toys' work together, and what tricks does an 'old dog' need to learn to accomplish all this?

Kubernetes may seem like a Rubik's Cube but there is some method in it's madness that have made it popular.

So if Kubernetes is in your future but you do not know where to start then you should probably be in this presentation. You will see exactly what you need to do to move from LAMP to Kube, why you have to do those steps, and how to use your new containerized environment.

Room:

Ballroom G

Time:

Saturday, March 11, 2023 - 13:30 to 14:30

I LOVE LAMP

Linux
Apache
MySQL
PHP

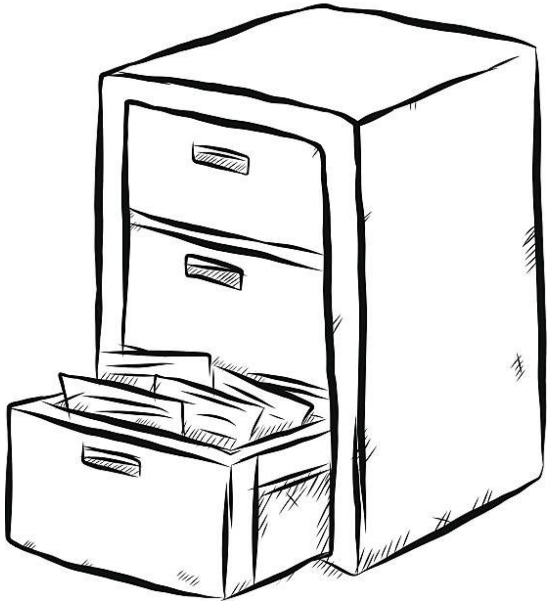
Conspiracy -

Once you understand a technology well enough to be really good enough it will be changed to something much more complex and not nearly as enjoyable!

Lets us ignore the Linux and PHP aspects for now

Many folks were happy with the single web server and single database

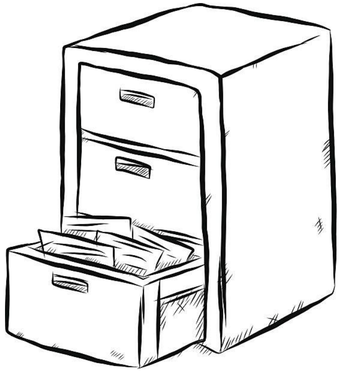
.. for a while



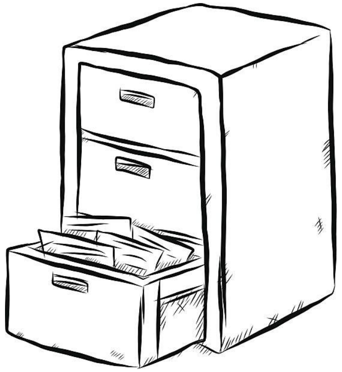
Lets us ignore the Linux and PHP for now

The ability to split read only and read/write database access can provide extra throughput

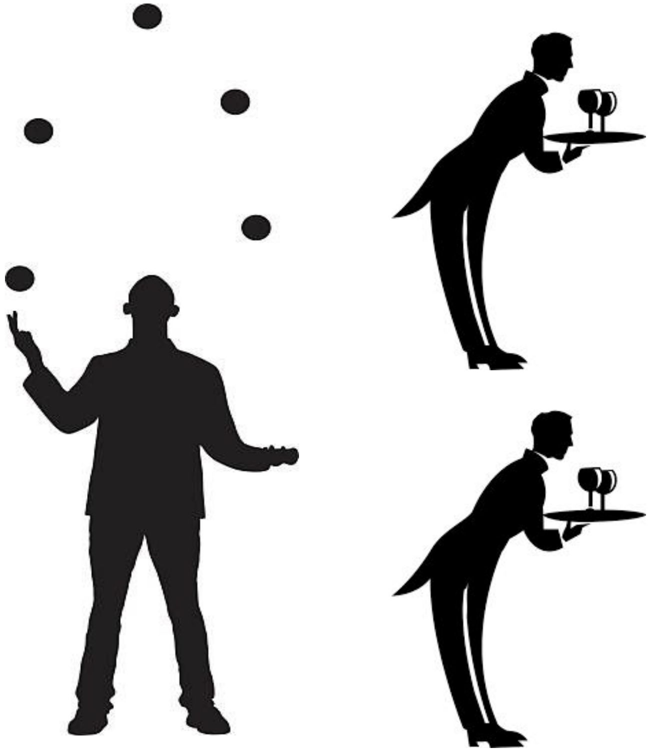
Read Only



Read/Write

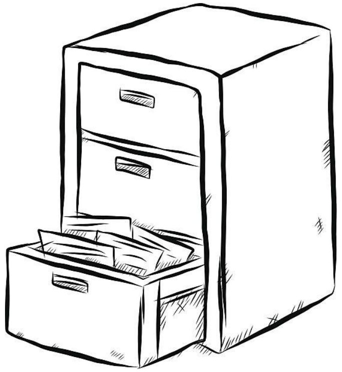


Lets us ignore the Linux and PHP for now

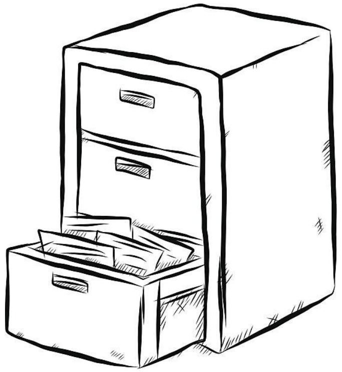


Multiple web servers could also improve performance but you probably had to add a load balancer

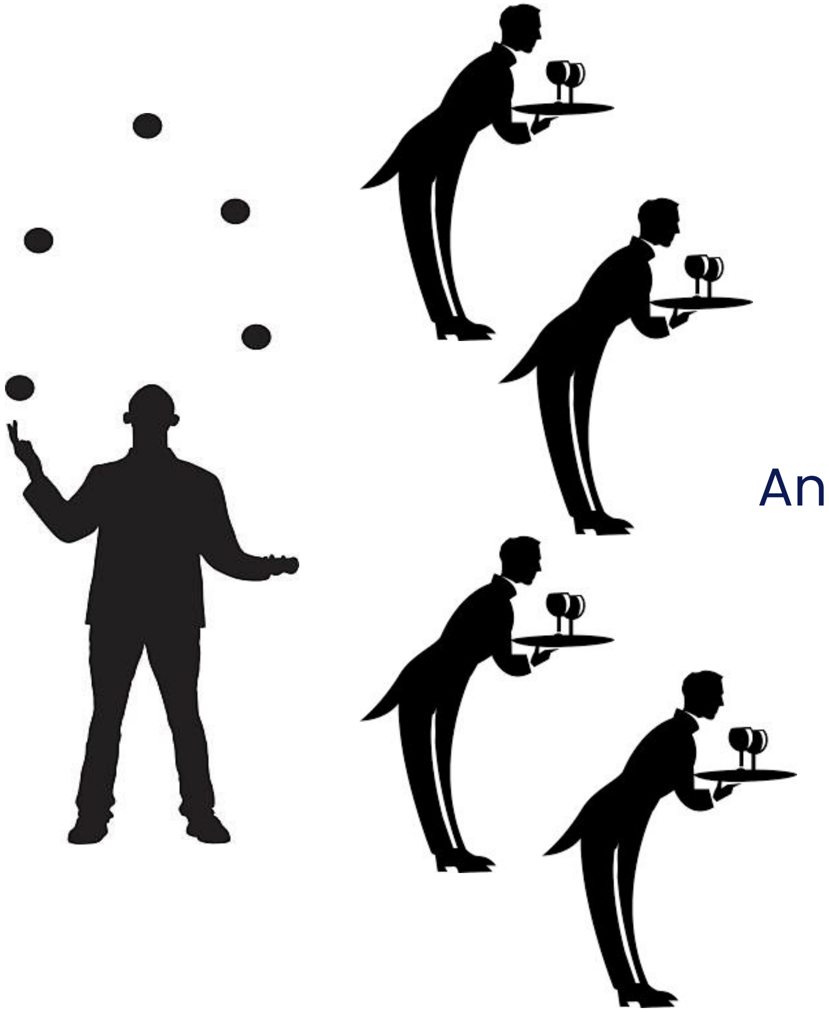
Read Only



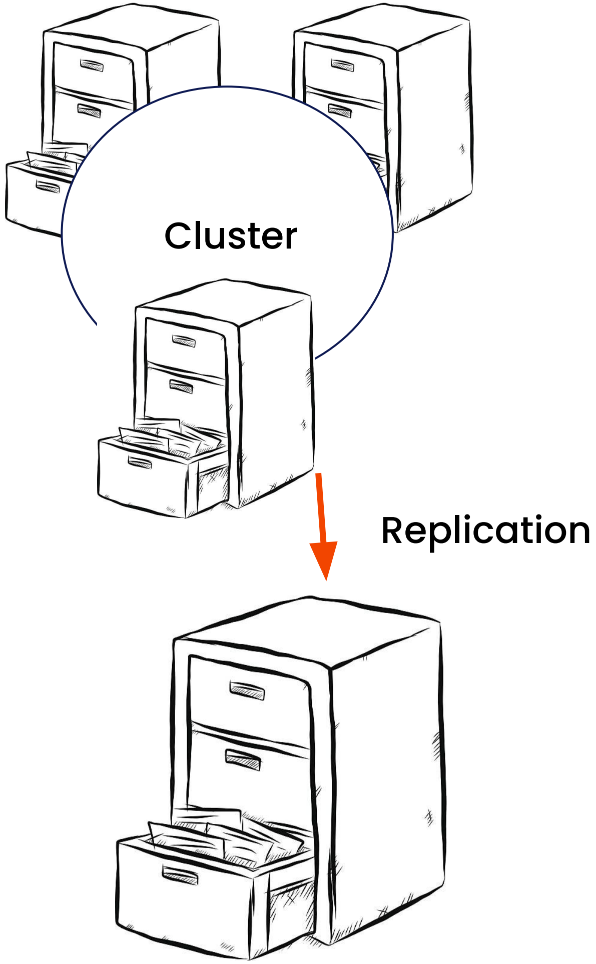
Read/Write



Lets us ignore the Linux and PHP for now



And of course, things get complex



Two Obvious Problems

s/obvious/expensive/

Problem #1

Not all applications utilized all the resources

In many cases they were using only a fraction of the available resources

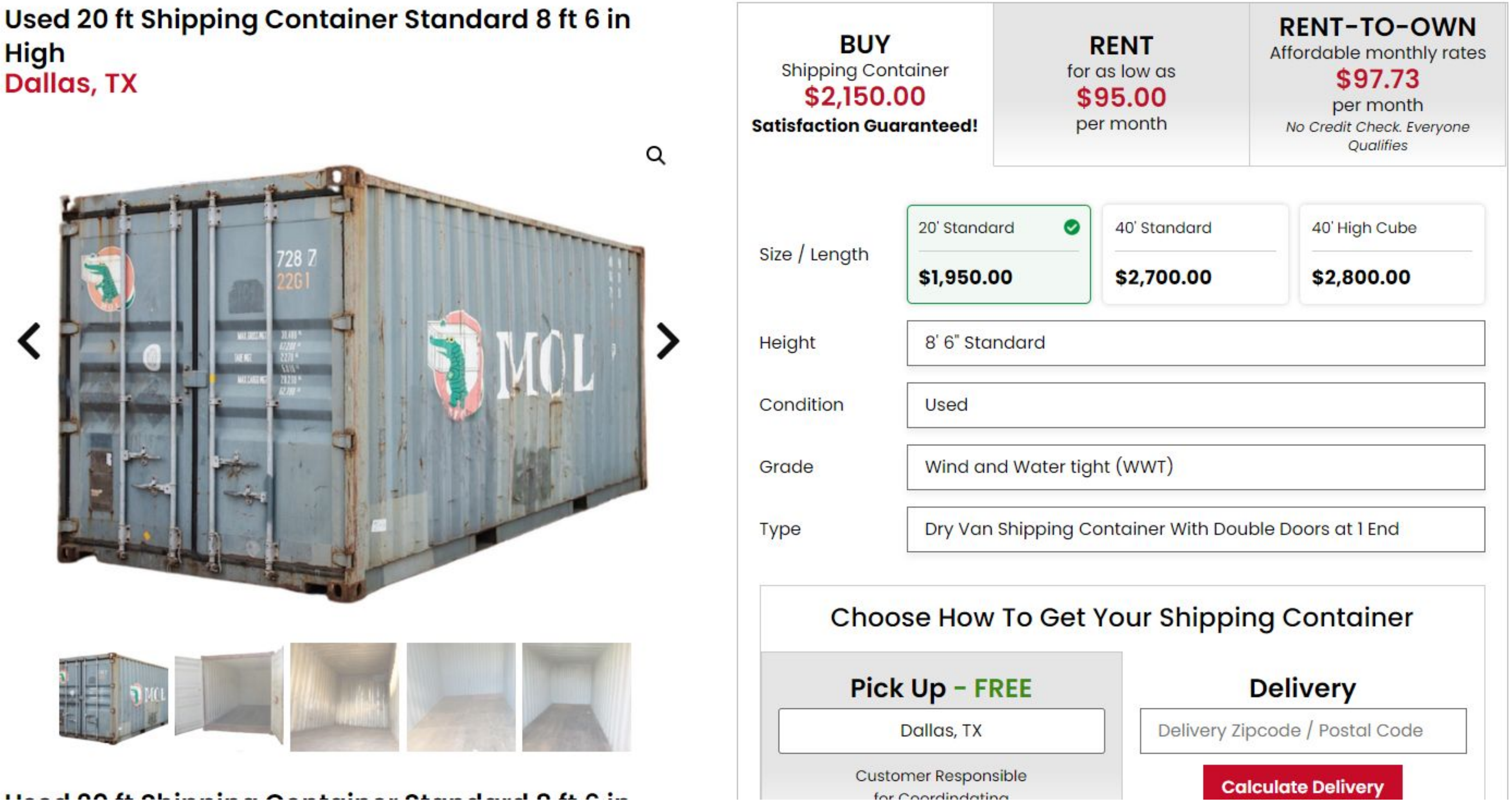
Business speak - Excess capacity

Containers

More bang?

For more bucks??

What if you could package things better



Used 20 ft Shipping Container Standard 8 ft 6 in High
Dallas, TX

BUY	RENT	RENT-TO-OWN
Shipping Container $2,150.00 **Satisfaction Guaranteed!**	for as low as $95.00 per month	Affordable monthly rates $97.73 per month *No Credit Check. Everyone Qualifies*

Size / Length	20' Standard ✓ **$1,950.00**	40' Standard **$2,700.00**	40' High Cube **$2,800.00**
Height	8' 6" Standard		
Condition	Used		
Grade	Wind and Water tight (WWT)		
Type	Dry Van Shipping Container With Double Doors at 1 End		

Choose How To Get Your Shipping Container

Pick Up - FREE	Delivery
Dallas, TX	Delivery Zipcode / Postal Code
Customer Responsible for Coordinating	Calculate Delivery

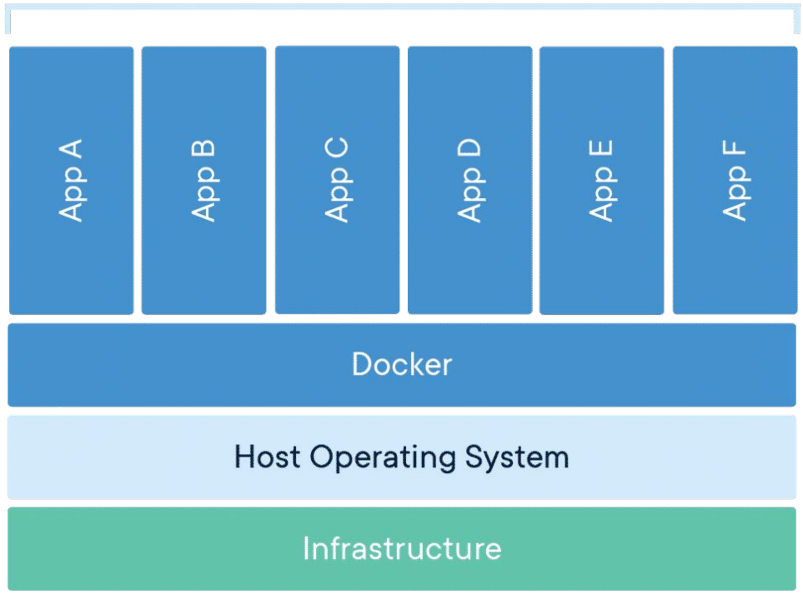
Used 20 ft Shipping Container Standard 8 ft 6 in

Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels.

Because all of the containers share the services of a single operating system kernel, they use fewer resources than virtual machines

https://en.wikipedia.org/wiki/Docker_(software)

Containerized Applications



Containers emerged as a way to make software portable. The container contains all the packages you need to run a service.

The provided file system makes containers extremely portable and easy to use in development.

A container can be moved from development to test or production with no or relatively few configuration changes.

DOGBERT'S TECH SUPPORT
IT WORKS FINE ON MY MACHINE.
YES, BUT THIS CALL IS ABOUT MY PC. MAY WE TALK ABOUT MY PROBLEM NOW?
OKAY. YOUR PC IS DEFECTIVE AND YOU'RE SELFISH. THAT'S AN ATTRACTIVE PACKAGE YOU'VE GOT GOING THERE.
4/5/02 © 2002 United Feature Syndicate, Inc.

Containers - database example

install curl

install docker

```
docker run -d --name percona-server-1 -e \
MYSQL_ROOT_PASSWORD=hidave percona/percona-server:8.0
```

What it looks like while running

```
$ sudo docker image ls
REPOSITORY               TAG       IMAGE ID       CREATED       SIZE
percona/percona-server   8.0       0dda075f0d2d   2 weeks ago   1.4GB
```

Status

$ sudo docker container ps

```
CONTAINER ID   IMAGE                         COMMAND                  CREATED          STATUS          PORTS
NAMES

bebf363885e8   percona/percona-server:8.0    "/docker-entrypoint.…"   2 minutes ago    Up 2 minutes    3306/tcp, 33060/tcp
percona-server
```

```
$ sudo docker exec -it percona-server /bin/bash
[mysql@bebf363885e8 /]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.31-23 Percona Server (GPL), Release 23, Revision 71449379

Copyright (c) 2009-2022 Percona LLC and/or its affiliates
Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Stopping that container

```
$ sudo docker container ps
```

```
CONTAINER ID   IMAGE                          COMMAND                  CREATED          STATUS         PORTS                 NAMES
bebf363885e8   percona/percona-server:8.0     "/docker-entrypoint.…"   7 minutes ago    Up 6 minutes   3306/tcp, 33060/tcp   percona-server
```

```
$ sudo docker stop bebf363885e8
```

```
bebf363885e8
```

Cloud

Before we had airplanes and astronauts, we really thought that there was an actual place beyond the clouds, somewhere over the rainbow. There was an actual place, and we could go above the clouds and find it there.

Barbara Walters

The Rush to the Cloud

1. It has to be cheaper, right?
2. No longer need a computer room, those compute operators, big air handlers, high electricity bills, an ongoing capital budget fight for new computer stuff, hardware service contracts, and all the *yucky* stuff.
3. Need an upgrade? Put it on the credit card! It is still cheaper, right?
4. Faster to provision a server.
5. Better integration between our handy dandy CI/CD system and containerized software and easy to provision servers.
6. Almost infinite scaling, as long as your credit holds!

INE
ONE

High Cube
DRYU 958501 2
45G1
MAX.GROSS
TARE
NET
CU. CAP.
CAUTION 9'6 HIGH CONTAINER

Kubernetes

“Open the pod bay doors, HAL”

IS Kubernetes the Operating System of the Cloud?

Kubernetes (/ˌk(j)uːbərˈnɛtɪs, -ˈneɪtɪs, -ˈneɪtiːz, -ˈnɛtiːz/, commonly stylized as **K8s**[) is an open-source container orchestration system for automating software deployment, scaling, and management. Originally designed by Google, the project is now maintained by the Cloud Native Computing Foundation.

The name Kubernetes originates from Greek, meaning helmsman or pilot. Kubernetes is often abbreviated as K8s, counting the eight letters between the "K" and the "s" (a numeronym).

Its suitability for running and managing large cloud-native workloads has led to widespread adoption of it in the data center. There are multiple distributions of this platform - from ISVs as well as hosted-on cloud offerings from all the major public cloud vendors.

https://en.wikipedia.org/wiki/Kubernetes

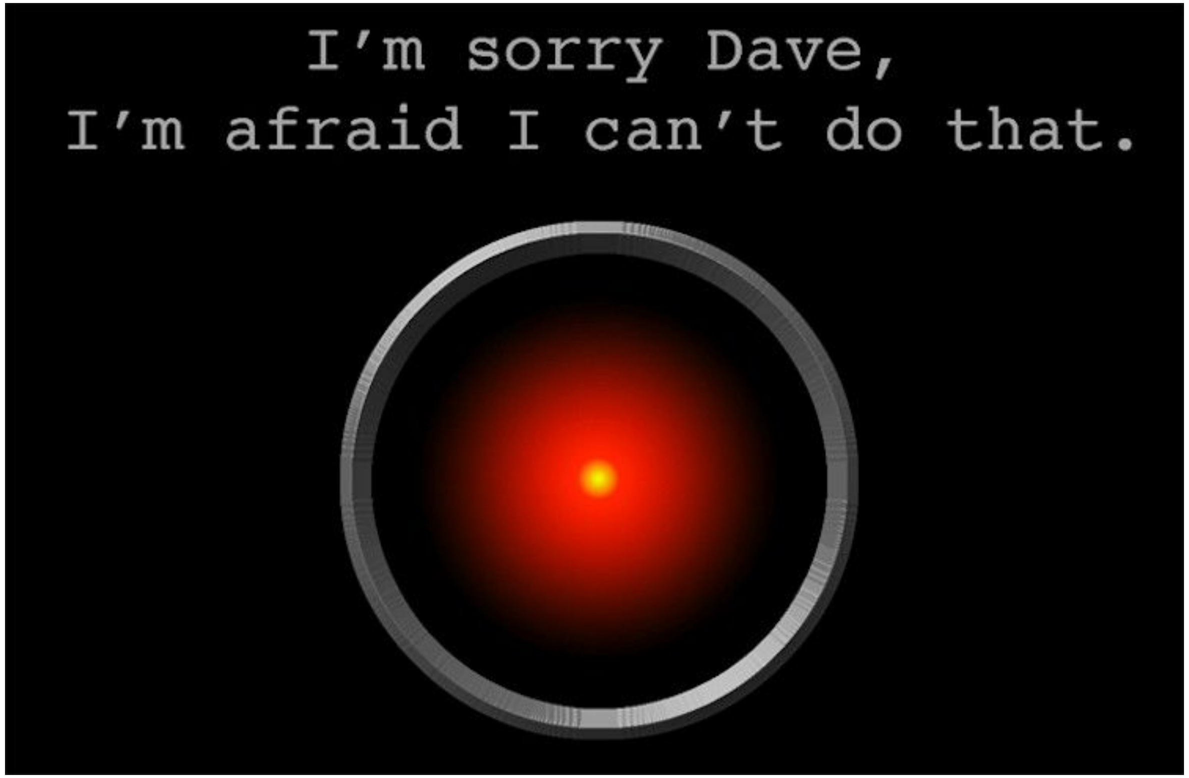
The POD

The basic scheduling unit in Kubernetes is a pod,[which consists of **one or more containers that are guaranteed to be co-located on the same node**.

Each pod in Kubernetes is assigned a unique IP address within the cluster, allowing applications to use ports without the risk of conflict.**Within the pod, all containers can reference each other**.

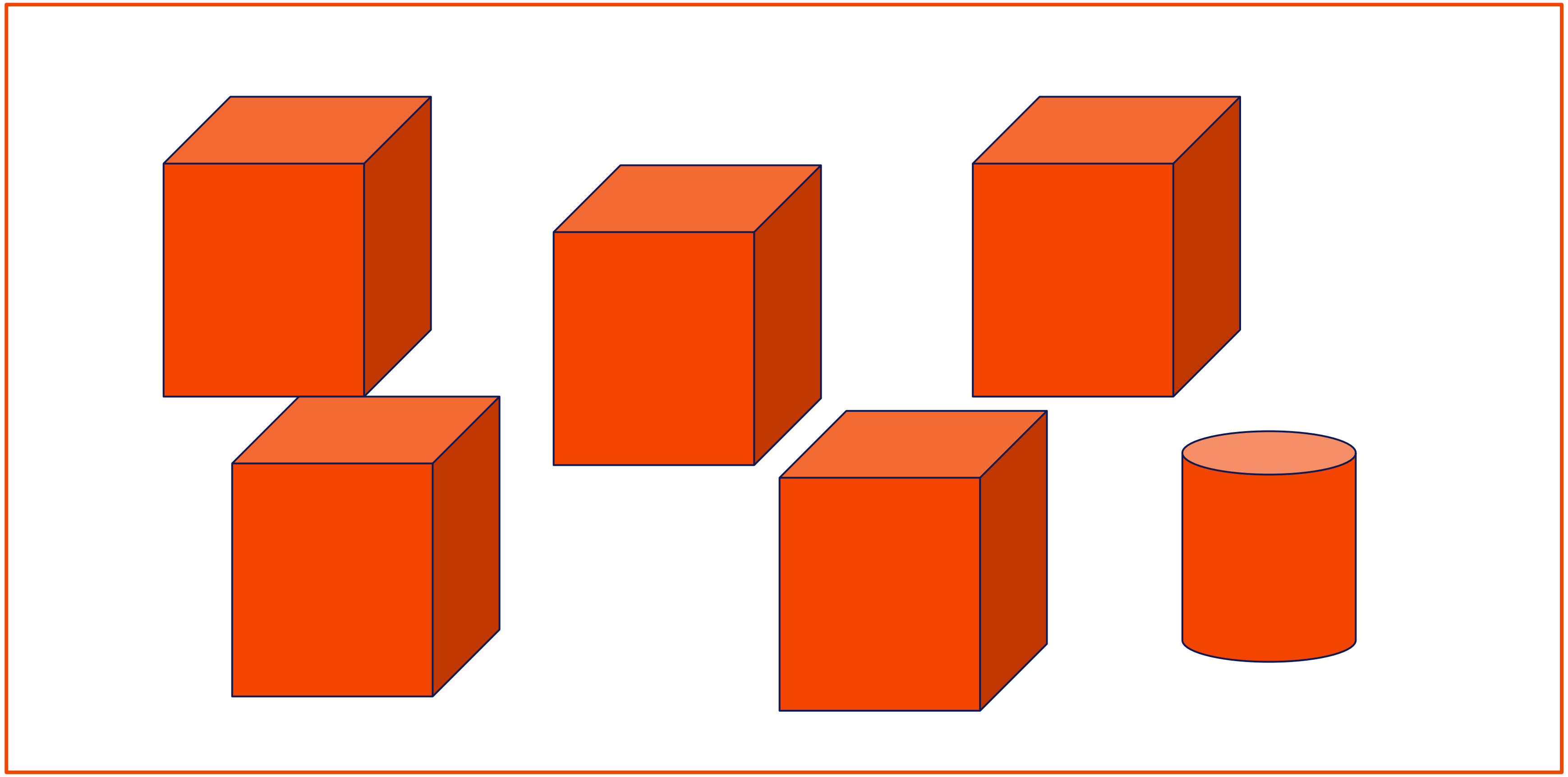
Open the pod-bay doors, HAL.

Open the Pod bay doors, HAL.



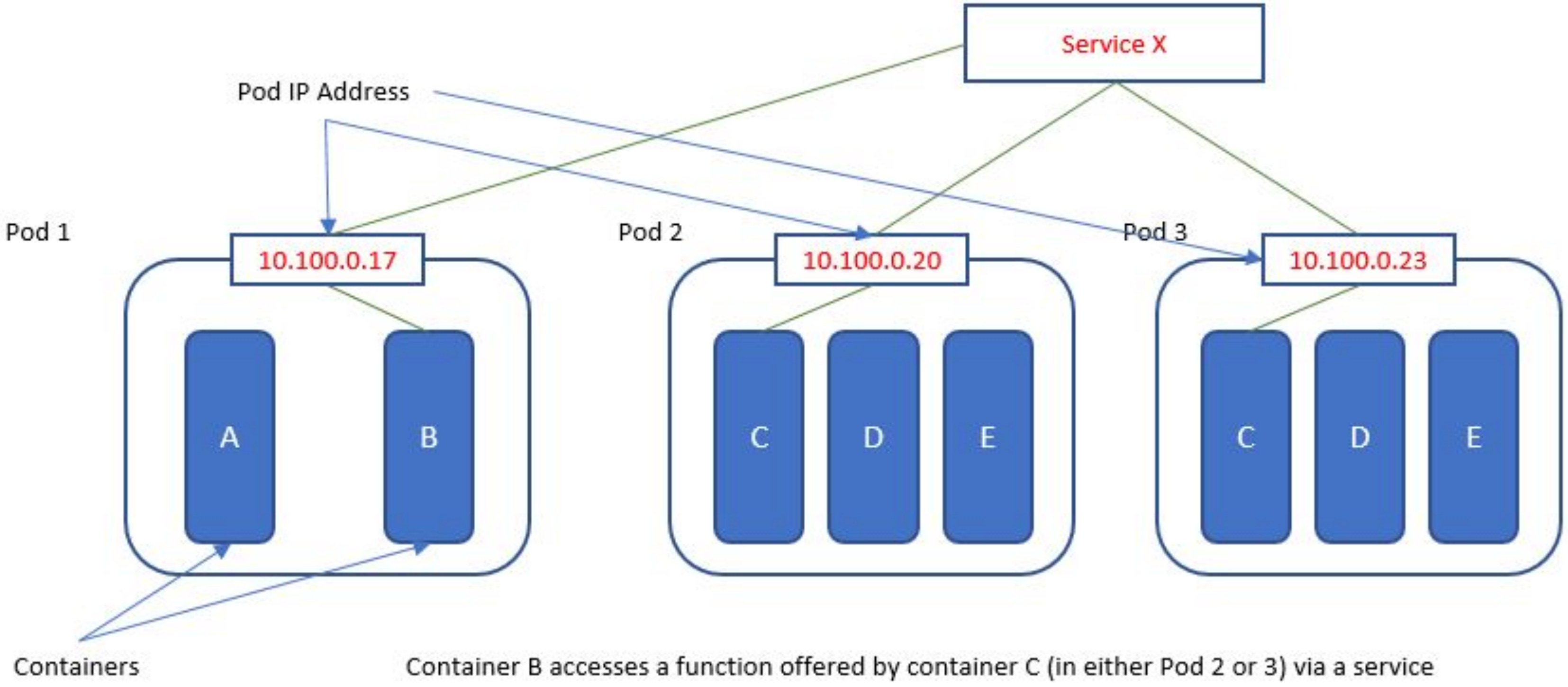
I'm sorry Dave,
I'm afraid I can't do that.

Node has one or more containers



Node

PODs can interact


Service X
Pod IP Address
Pod 1
10.100.0.17
A
B
Pod 2
10.100.0.20
C
D
E
Pod 3
10.100.0.23
C
D
E
Containers
Container B accesses a function offered by container C (in either Pod 2 or 3) via a service

A MySQL Example

With persistent storage

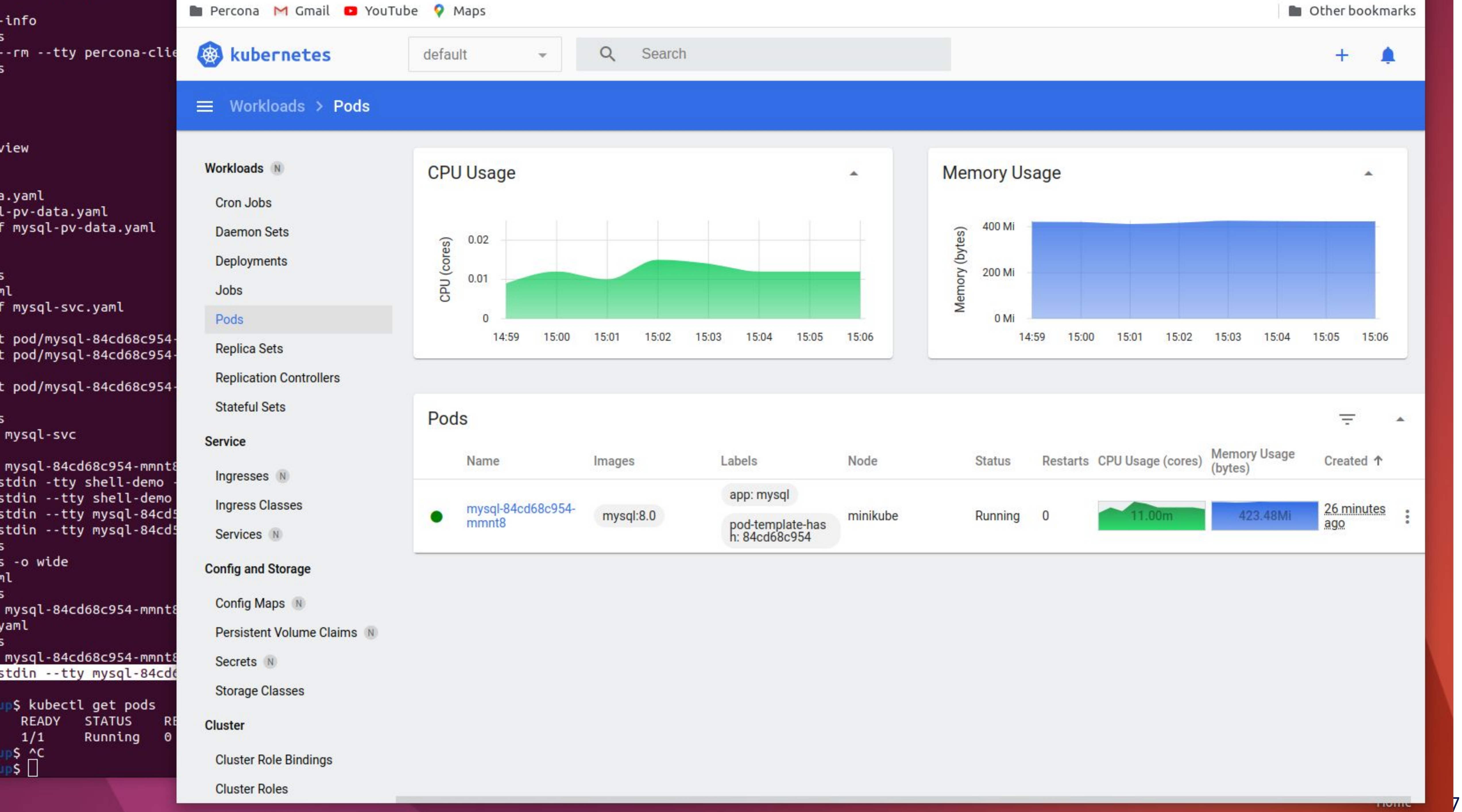
```
$ minikube start --driver=docker
😄  minikube v1.29.0 on Ubuntu 22.04
✨  Using the docker driver based on user configuration
📌  Using Docker driver with root privileges
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
💾  Downloading Kubernetes v1.26.1 preload ...
    > preloaded-images-k8s-v18-v1...:  397.05 MiB / 397.05 MiB  100.00% 5.90 Mi
    > gcr.io/k8s-minikube/kicbase...:  407.19 MiB / 407.19 MiB  100.00% 4.27 Mi
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.26.1 on Docker 20.10.23 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...
    ▪ Configuring RBAC rules ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🔎  Verifying Kubernetes components...
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default"
namespace by default
```

mysql-svc.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: mysql
spec:
  ports:
  - port: 3306
  selector:
    app: mysql
  clusterIP: None
---
apiVersion: apps/v1 # for versions before 1.9.0 use apps/v1beta2
kind: Deployment
metadata:
  name: mysql
spec:
  selector:
    matchLabels:
      app: mysql
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: mysql
    spec:
      containers:
      - image: mysql:8.0
        name: mysql
        env:
          # Use secret in prod use cases
        - name: MYSQL_ROOT_PASSWORD
          value: hidave
        ports:
        - containerPort: 3306
          name: mysql
        volumeMounts:
        - name: mysql-persistent-storage
          mountPath: /var/lib/mysql
      volumes:
      - name: mysql-persistent-storage
        persistentVolumeClaim:
          claimName: mysql-pv-data
```

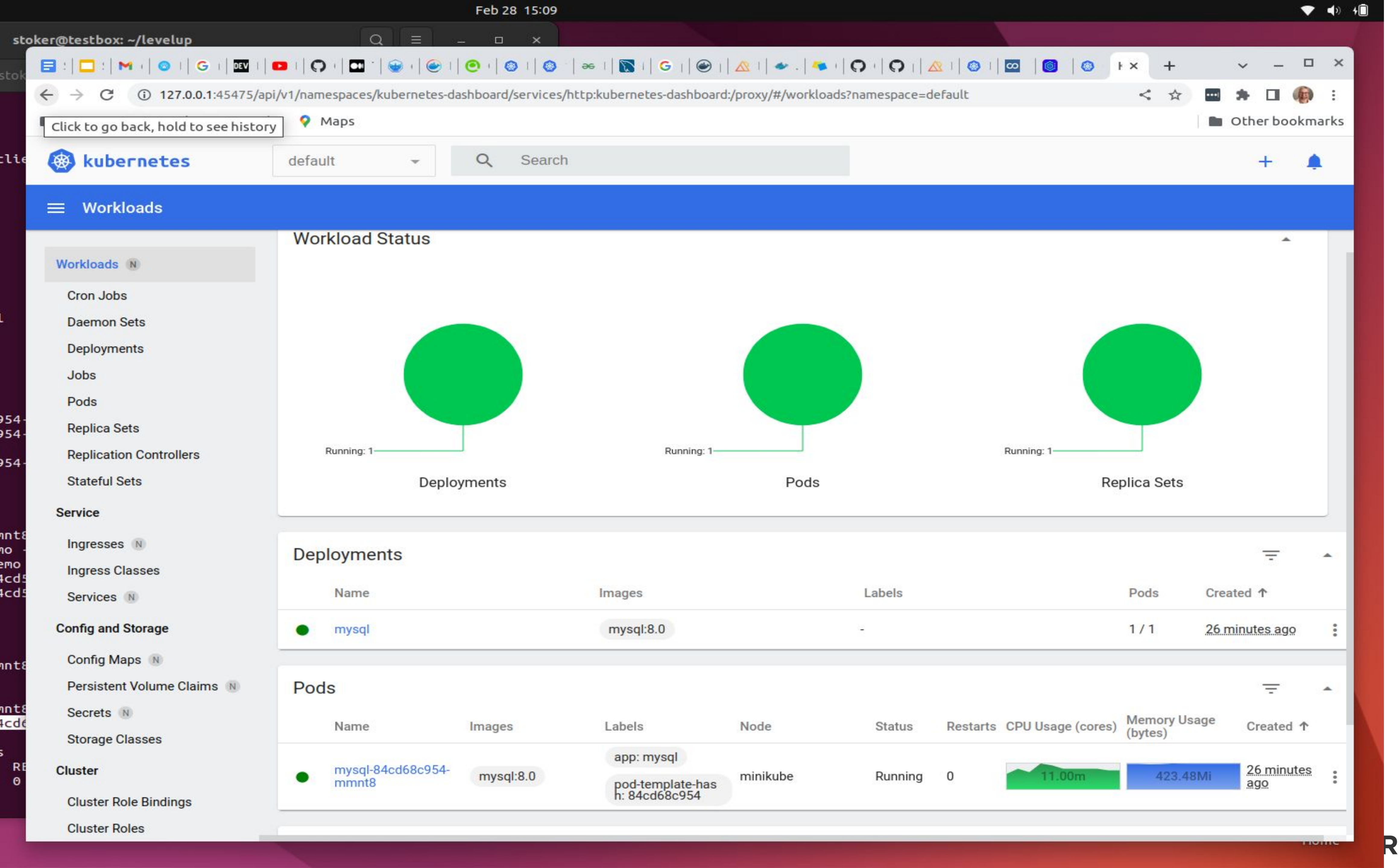
Get POD Running

```
$ kubectl apply -f mysql-pv-data.yaml
$ kubectl apply -f mysql-svc.yaml
$ kubectl get svc
NAME         TYPE        CLUSTER-IP    EXTERNAL-IP    PORT(S)     AGE
kubernetes   ClusterIP   10.96.0.1     <none>         443/TCP     46m
mysql        ClusterIP   None          <none>         3306/TCP    41m
$ kubectl get pods
NAME                        READY    STATUS     RESTARTS    AGE
mysql-84cd68c954-mmnt8      1/1      Running    0           41m
$ kubectl exec --stdin --tty mysql-84cd68c954-mmnt8 -- /bin/bash
```



```
-info
s
--rm --tty percona-clie
s



view


a.yaml
l-pv-data.yaml
f mysql-pv-data.yaml


s
ml
f mysql-svc.yaml

t pod/mysql-84cd68c954-
t pod/mysql-84cd68c954-

t pod/mysql-84cd68c954-

s
 mysql-svc

 mysql-84cd68c954-mmnt8
stdin -tty shell-demo -
stdin --tty shell-demo
stdin --tty mysql-84cd5
stdin --tty mysql-84cd5
s
s -o wide
ml
s
 mysql-84cd68c954-mmnt8
yaml
s
 mysql-84cd68c954-mmnt8
stdin --tty mysql-84cd6

up$ kubectl get pods
 READY   STATUS    RE
 1/1     Running   0
up$ ^C
up$ 
```

	Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
●	mysql-84cd68c954-mmnt8	mysql:8.0	app: mysql pod-template-hash: 84cd68c954	minikube	Running	0	11.00m	423.48Mi	26 minutes ago



127.0.0.1:45475/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/#/workloads?namespace=default

Workload Status

Running: 1 — Deployments

Running: 1 — Pods

Running: 1 — Replica Sets

Deployments

	Name	Images	Labels	Pods	Created ↑
●	mysql	mysql:8.0	-	1 / 1	26 minutes ago

Pods

	Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
●	mysql-84cd68c954-mmnt8	mysql:8.0	app: mysql pod-template-hash: 84cd68c954	minikube	Running	0	11.00m	423.48Mi	26 minutes ago

Actually Talk To The Database

```
bash-4.4# mysql -u root -p -h 127.0.0.1
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.32 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Start A Kubed LAMP

```
$ kubectl create -f wordpress-deployment.yaml
service/wordpress created
persistentvolumeclaim/wp-pv-claim created
deployment.apps/wordpress created
```

```
$ kubectl create -f mysql-deployment.yaml
service/wordpress-mysql created
persistentvolumeclaim/mysql-pv-claim created
deployment.apps/wordpress-mysql created
```

```
$ kubectl get deployment
NAME                READY    UP-TO-DATE    AVAILABLE    AGE
wordpress           0/1      1             0            32s
wordpress-mysql     0/1      1             0            18s
```

RIght after launch

$ kubectl get svc

```
NAME               TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
kubernetes         ClusterIP      10.96.0.1        <none>        443/TCP        45m
phpservice         LoadBalancer   10.104.2.144     <pending>     80:30080/TCP   40m
wordpress          LoadBalancer   10.102.181.25    <pending>     80:30357/TCP   115s
wordpress-mysql    ClusterIP      None             <none>        3306/TCP       102s
```

These are not the droids you are searching for

SCALING

Need more resources, add pods

Need less resources, remove pod

Scale across data centers

YAML configuration files

```
apiVersion: v1
kind: Pod
metadata:
  name: static-web
  labels:
    role: webserver
spec:
  containers:
    - name: web
      image: nginx
      ports:
        - name: web
          containerPort: 80
          protocol: TCP
```

Does somewhat end tabs versus spaces arguments

Persistent Volumes

Most containers are ephemeral but you do not want your data to be that way

Persistent volumes or DBaaS are available

Fiddly Bits

Good Eglish Term

My 2¢

Too complicated

Too many varieties

Need homogenization

One size does not fit all, or most

When the only tool you have is a hammer you wack the *&$ out of everything

THANK YOU!

David.Stokes@Percona.com
@Stoker
Speakerdeck.com/Stoker

percona.com