

Introduction to Containers Cloud Kubernetes

Whitney High Hackathon

Steven Wong
Engineer, VMware
February 2019



Hackathon 2019 →

Speaker



Steve Wong

Open Source Engineer

VMware Cloud Native Applications Business Unit

Community participant:

- Kubernetes

Agenda

Recap: What is a container?

The world is changing

Why are people using container orchestrators and clouds

The Kubernetes project

Workshop – you use Kubernetes yourself to deploy a full n-tier application, running in the cloud, serving users on the internet

What is a container?



Dictionary definition:
something used to transport something else, holds things

Software containers

In the software world containers are used to hold and transport applications.



Container

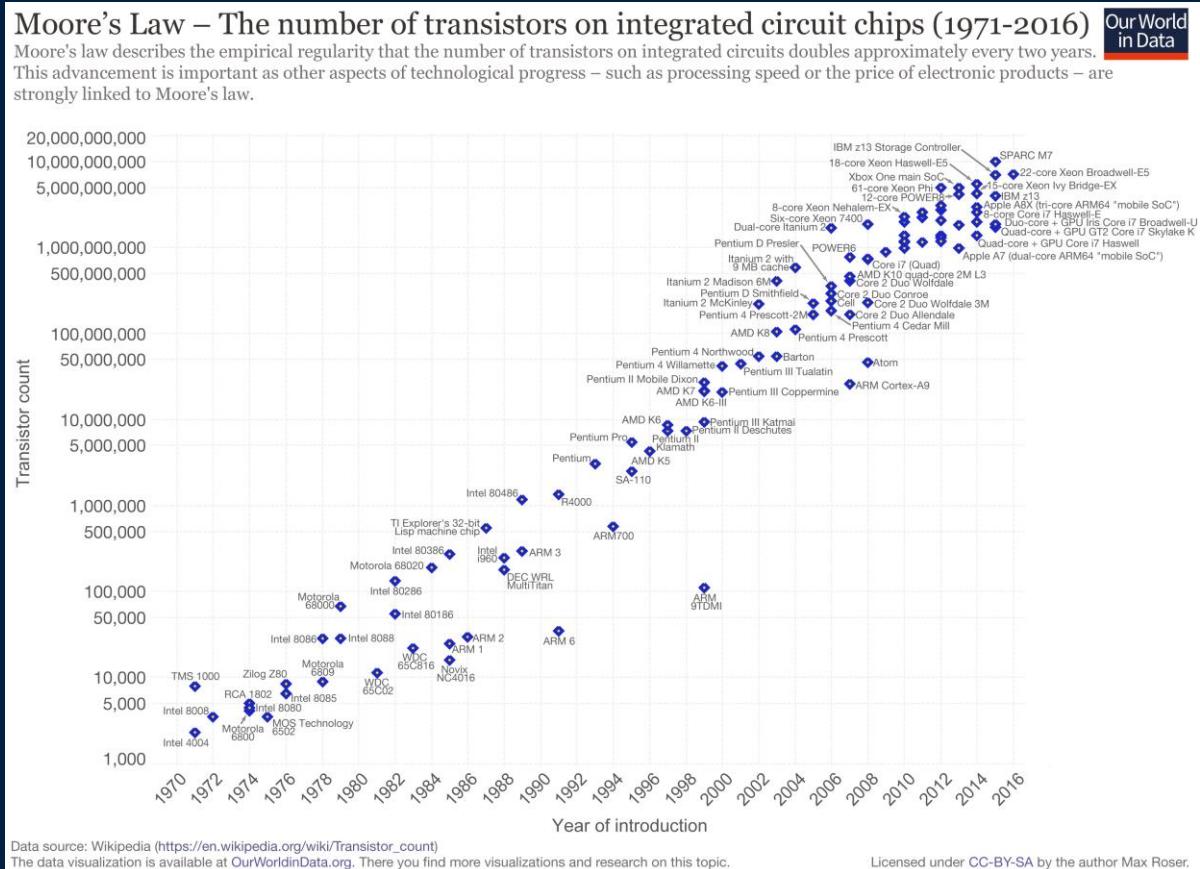
- Application
- Files
- Libraries
- Services

Photo: Matthew Paul Argall [CC BY-SA 3.0 (<https://creativecommons.org/licenses/by-sa/3.0>)], from Wikimedia Commons

Moore's Law

Often erroneously misstates as speed/capacity of computers doubles every 18 months

Used to be more true than it is now



Hackathon 2019

Seymour Cray

“Which would you rather use: two strong oxen or 1024 chickens?”

This quip was viewed as obvious into the 90s – but today you want the chickens

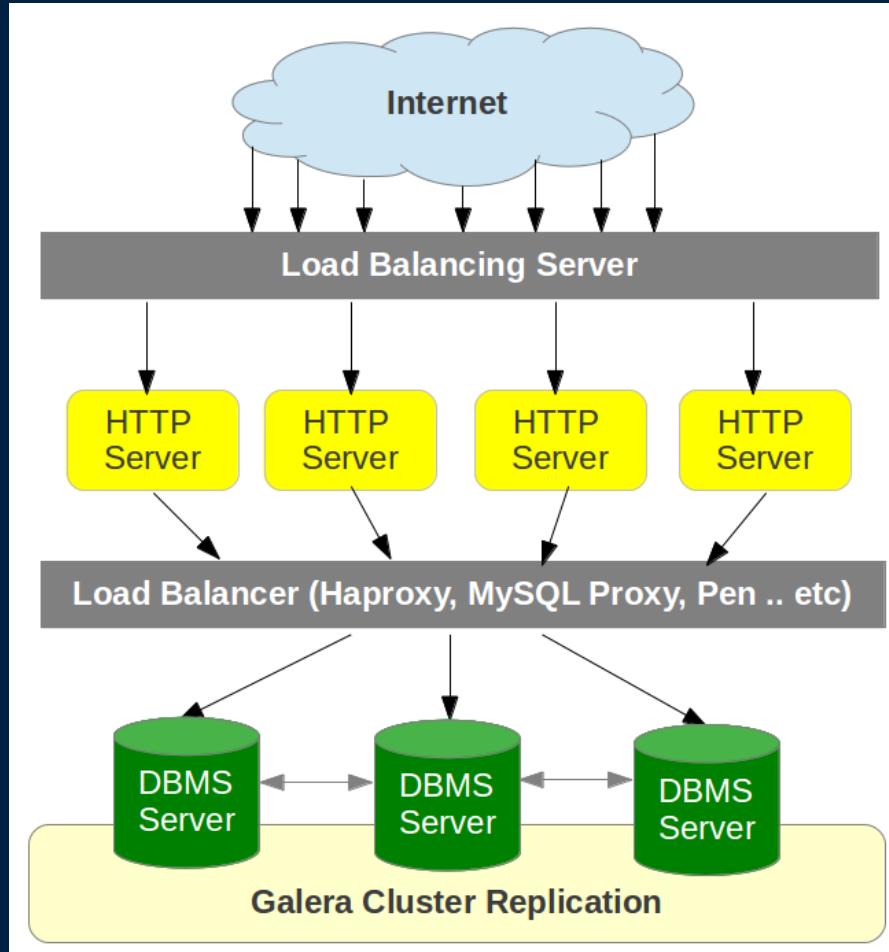


Photo Bibliophile [CC BY-SA 3.0]
(<https://creativecommons.org/licenses/by-sa/3.0/>)



Photo Joshua Coleman on Unsplash

What is a load balancer?



Consumer interacts with load balancer as if it is a single entity (service)

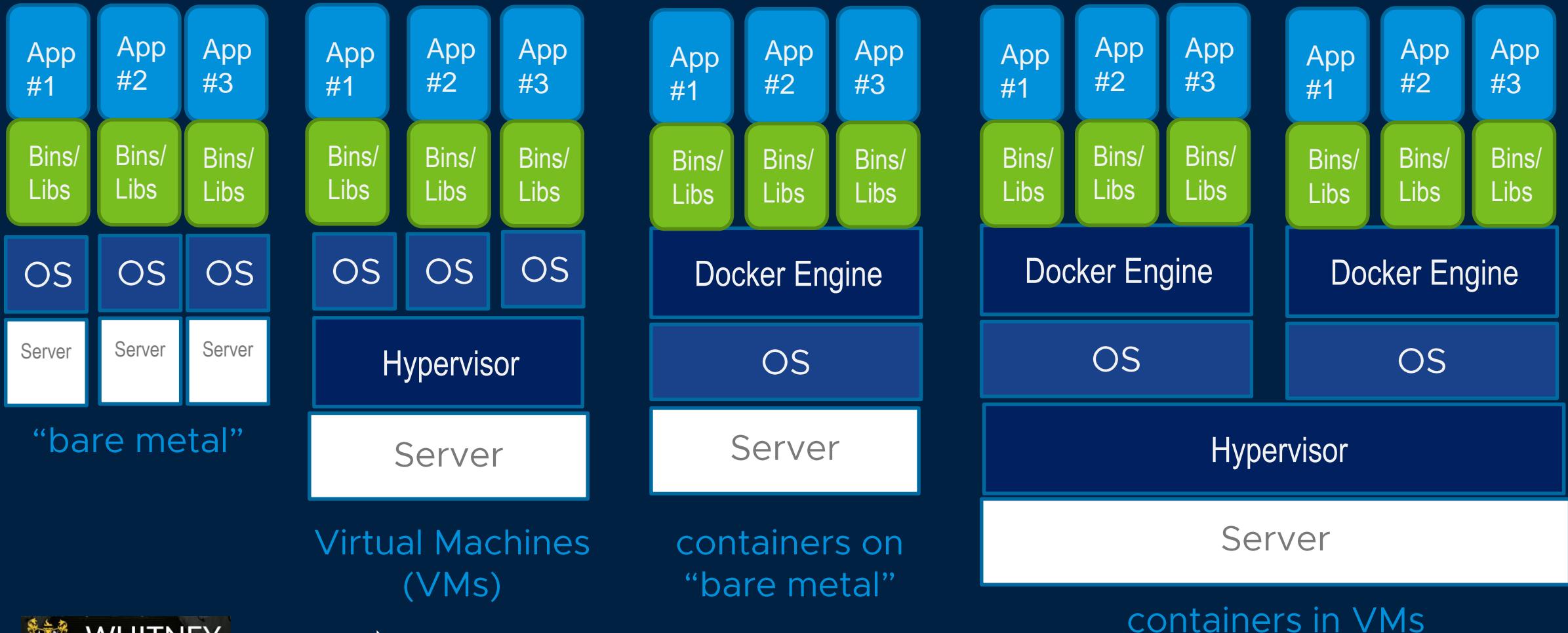
The load balancer off loads requests to a variable sized pool of server apps

The size of the the pool of server apps can change while the load balancer remains running

Elements of the server pool can be replaced (updated)

Options

deploy apps on machines



Container advantages

Compared to running code directly on hardware or in virtual machines

Start up faster

Are portable (unless you build them stupidly) – run them in a cloud or on your own computer. Run them on different types of computers.

Code is easily shareable, re-usable, publicly published if you desire.

Are useful for microservice architectures (building something big with small modular replaceable building blocks)

Are useful for building very large services that can deploy parallel instances of execution to meet demand peaks

Less work for you. Huge numbers of pre-packaged useful open source components are already published. For many popular open source projects, the container packaged form is the most tested and fully supported way to use it.

Containers are they really revolutionary?

Sometimes a great thing needs a second great thing?

Thomas Edison Franklin Institute Awards ceremony

06/28/51 11 years 3 months

Samuel Insull and the history of the electric utility



How fast can things change?

Can you spot the horse?

New York City - 1900



WHITNEY
HIGH SCHOOL

Hackathon 2019 →

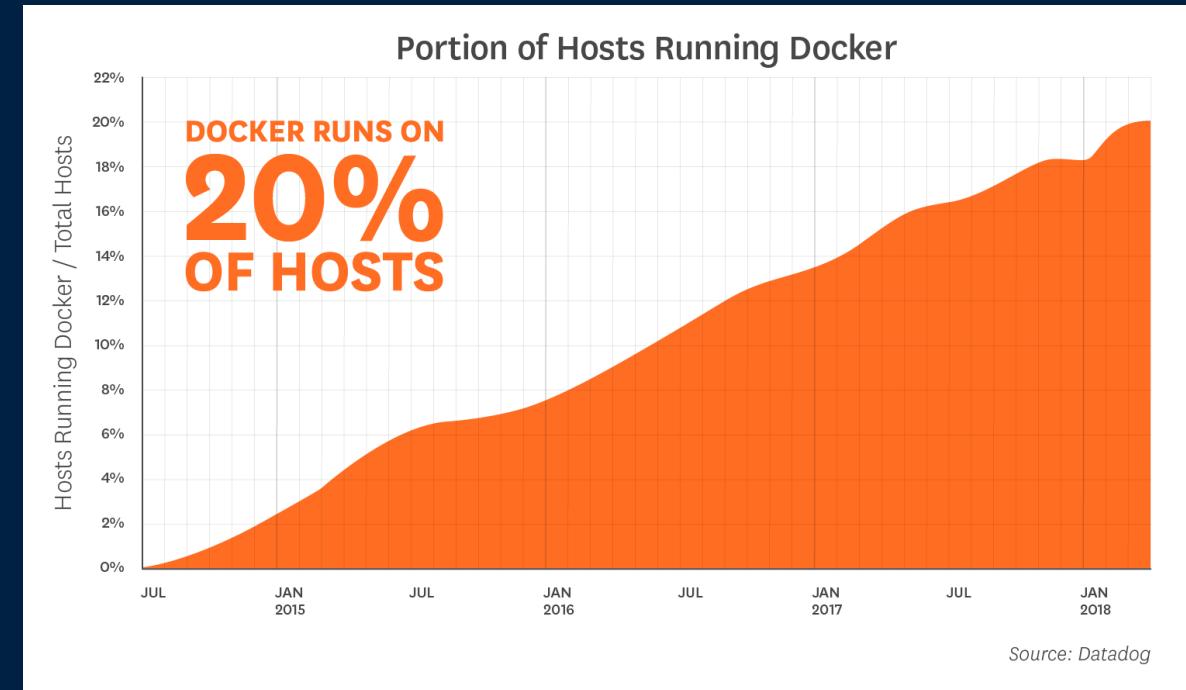
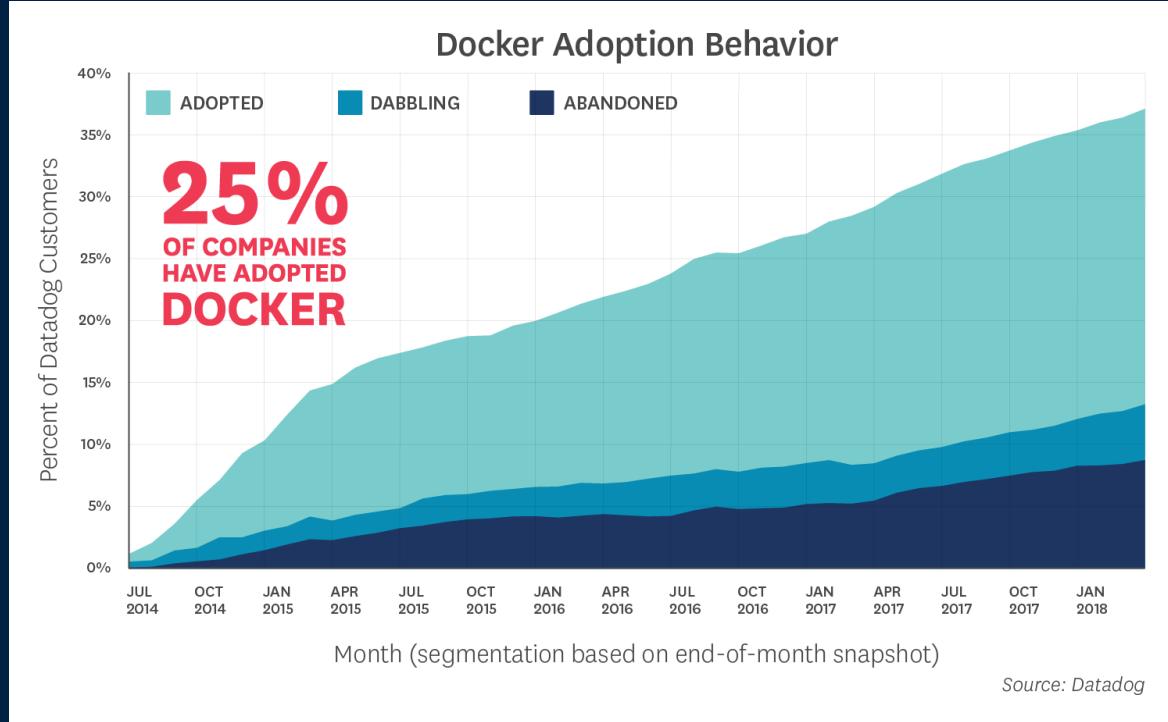
13 years later
Spot the horse



New York City - 1913

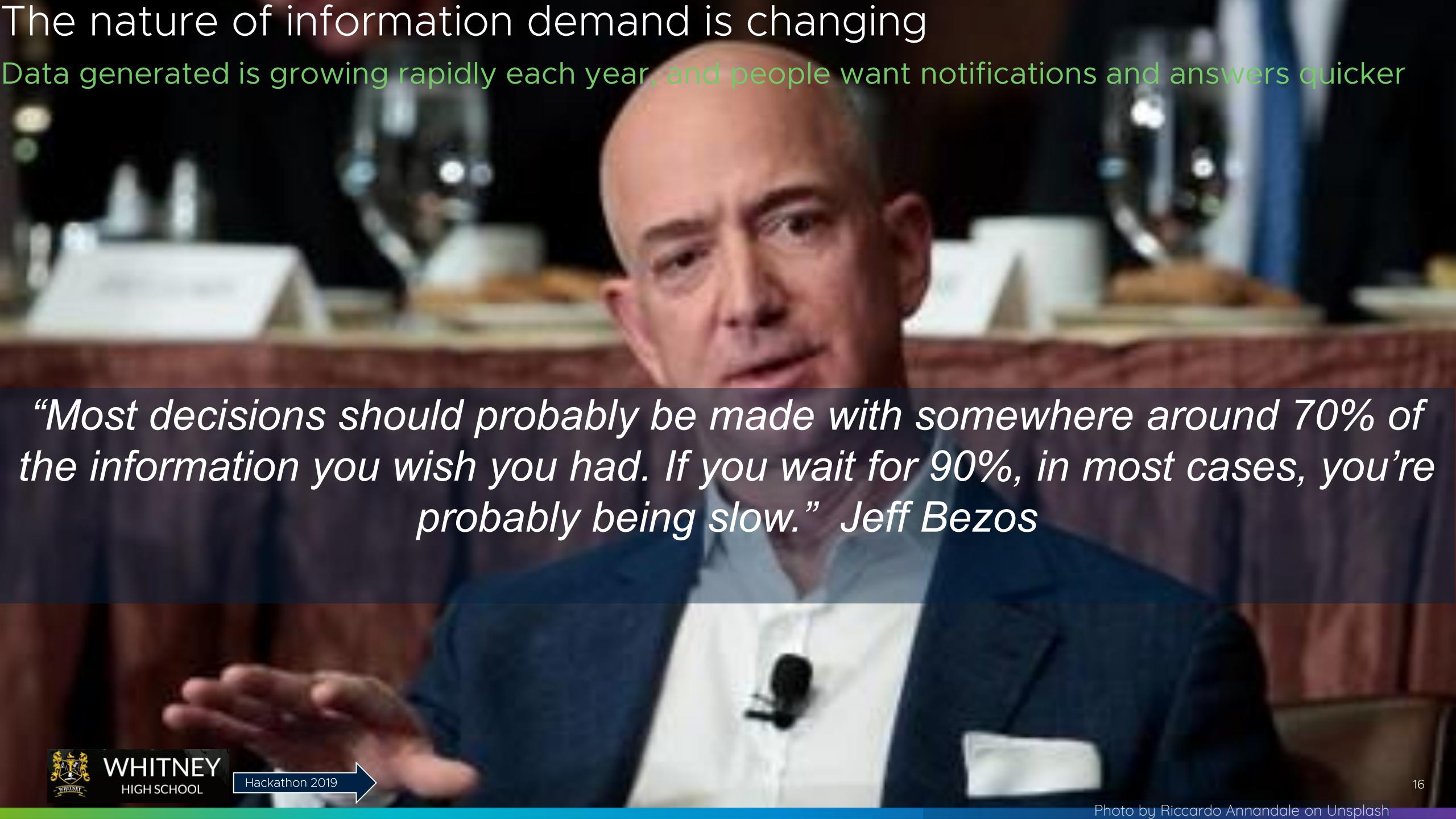
A black and white historical photograph showing a street scene in New York City from 1913. In the foreground, several streetcars are lined up along a city street. One streetcar has "1913" written on its side. The background features a dense cluster of tall, multi-story buildings, likely office or apartment buildings, typical of early 20th-century urban architecture.

Containers use is growing rapidly



The nature of information demand is changing

Data generated is growing rapidly each year, and people want notifications and answers quicker



“Most decisions should probably be made with somewhere around 70% of the information you wish you had. If you wait for 90%, in most cases, you’re probably being slow.” Jeff Bezos

Boeing 787

Dreamliner



Imagine having this in the 1800's?



WHITNEY
HIGH SCHOOL

→ Hackathon 2019

Containers

What if you need more than one, a lot more than one?



WHITNEY
HIGH SCHOOL

→ Hackathon 2019

What do you need to deliver consistent service?

Health Monitoring

Communications

Configuration

Enable Discovery

Security

Scaling

Scheduling



WHITNEY
HIGH SCHOOL

Hackathon 2019 →

Photo by Manuel Nageli on Unsplash

Kubernetes

Inspired by Google's experience

Kubernetes (Greek for "helmsman" or "pilot")

Its development and design are heavily influenced by Google's internal Borg system and many of the top contributors to the project previously worked on Borg.

The original name for Kubernetes within Google was project Seven of Nine, a reference to a Star Trek character that is a 'friendlier' Borg

Google's lawyers rejected taking the internal codename public, thus the Kubernetes name. The seven spokes on the wheel of the Kubernetes logo is an acknowledgment of the banished name.

Kubernetes v1.0 was released on July 21, 2015.



Kubernetes

Inspired by Google's experience

Manage applications, *not machines*

Open source

Supports multiple cloud plus
on premise datacenters

Inspired and informed by Google's experiences and
internal systems

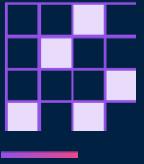
Things that do what Kubernetes does are often
called “container orchestrators”



Why would you want to use a container orchestrator?



Docker, Microservices and container-based development, with CI/CD



Machine Learning, AI, Streaming Data Analytics, Fast Data, Big Data



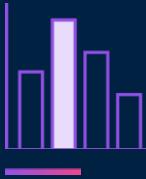
Need self service IT user experience, without using Amazon, or using clouds supplemented with on-prem and edge capacity



Seek Google-like datacenter operations, off the shelf, instead of home-grown



Need app and service portability across public clouds + on-prem... with consistent staff skill set, and no cloud provider lock-ins



Automated and Integrated security, networking, storage management, logging, health monitoring

Called an orchestrator but more like Jazz Improv



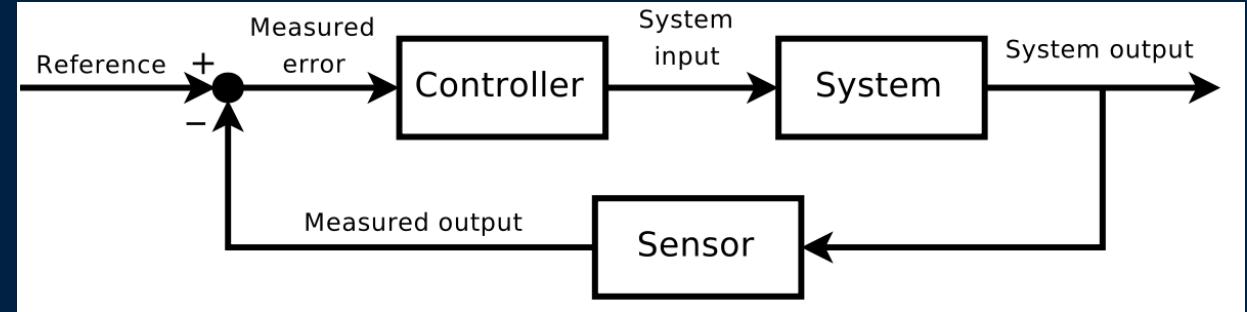
WHITNEY
HIGH SCHOOL

→ Hackathon 2019

Kubernetes

Based on control loops

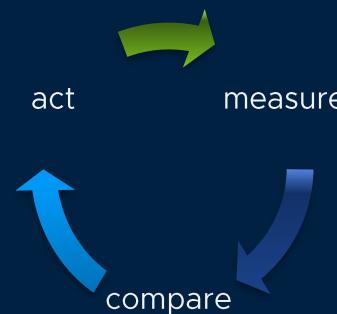
You tell Kubernetes the desired state



Kubernetes relentlessly:

- measures current state
- drives **current state -> desired state**

Recurring pattern of aspects in the system



Desired state analogy

Say what you want – something else takes care of details



Child

- Go upstairs
- Get undressed
- Put on pajamas
- Brush your teeth
- Pick out 2 stories



Employee

- Go get some sleep

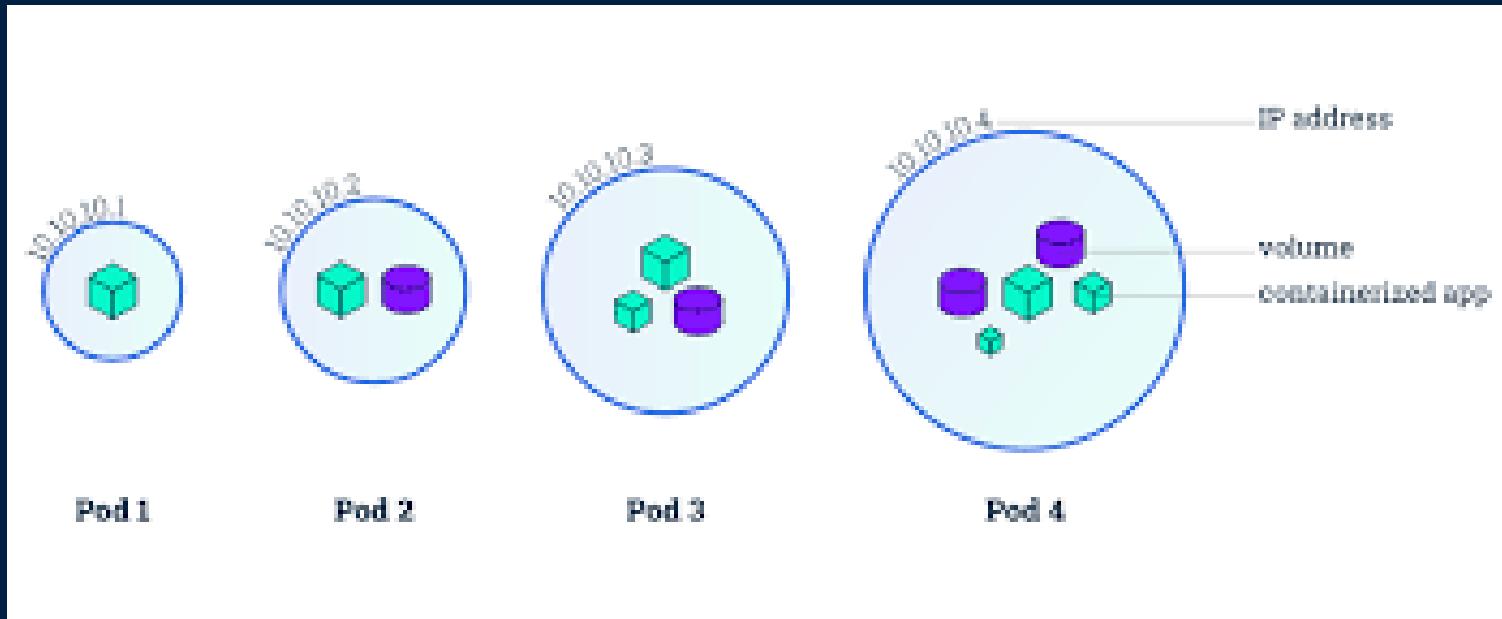
A Pod of **whales** containers

Scale out 3 tier application

The “atom” of Kubernetes. A group of containers – along with configuration

Kubernetes runs your application and services in pods, but much of Kubernetes itself is running as pods

Pods run on worker nodes – they never move once started – but they can be killed and restarted



Deployments

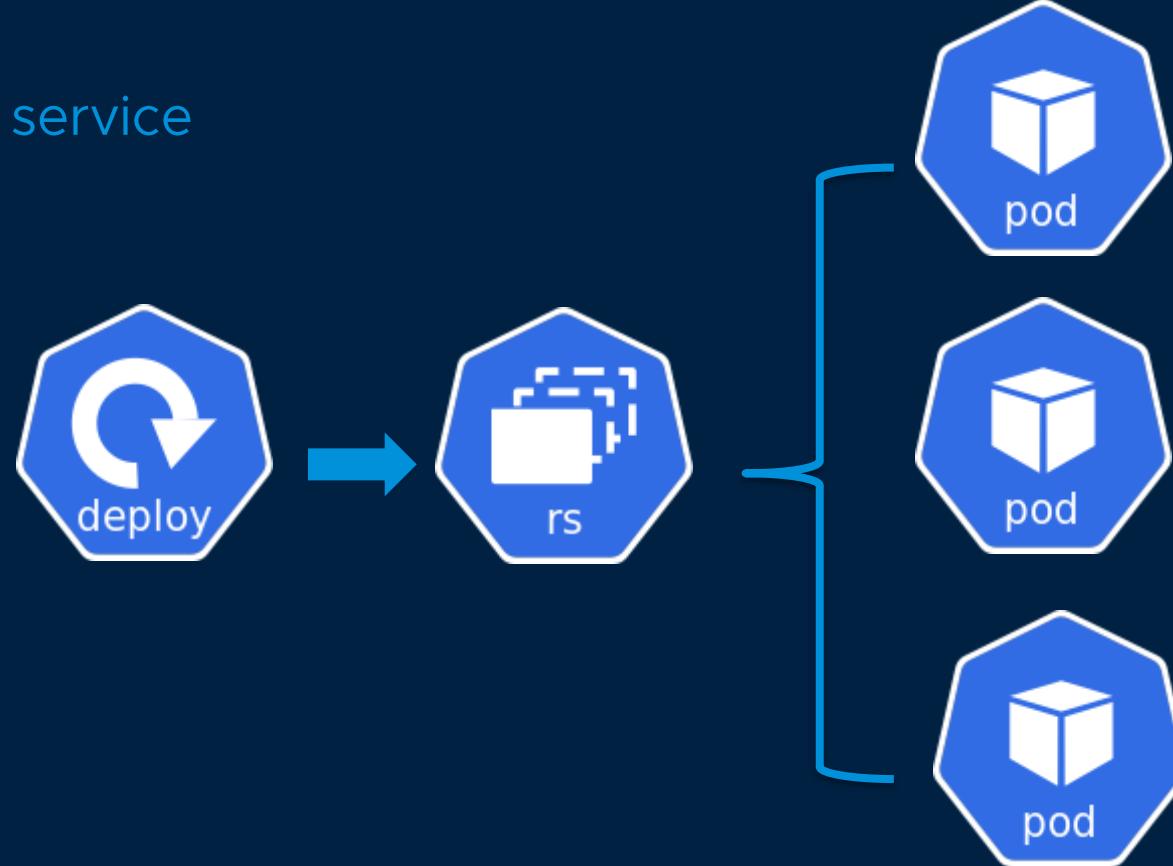
Scale out multi-pod application or service

There should be:

3 front-ends (UI)

2 Services

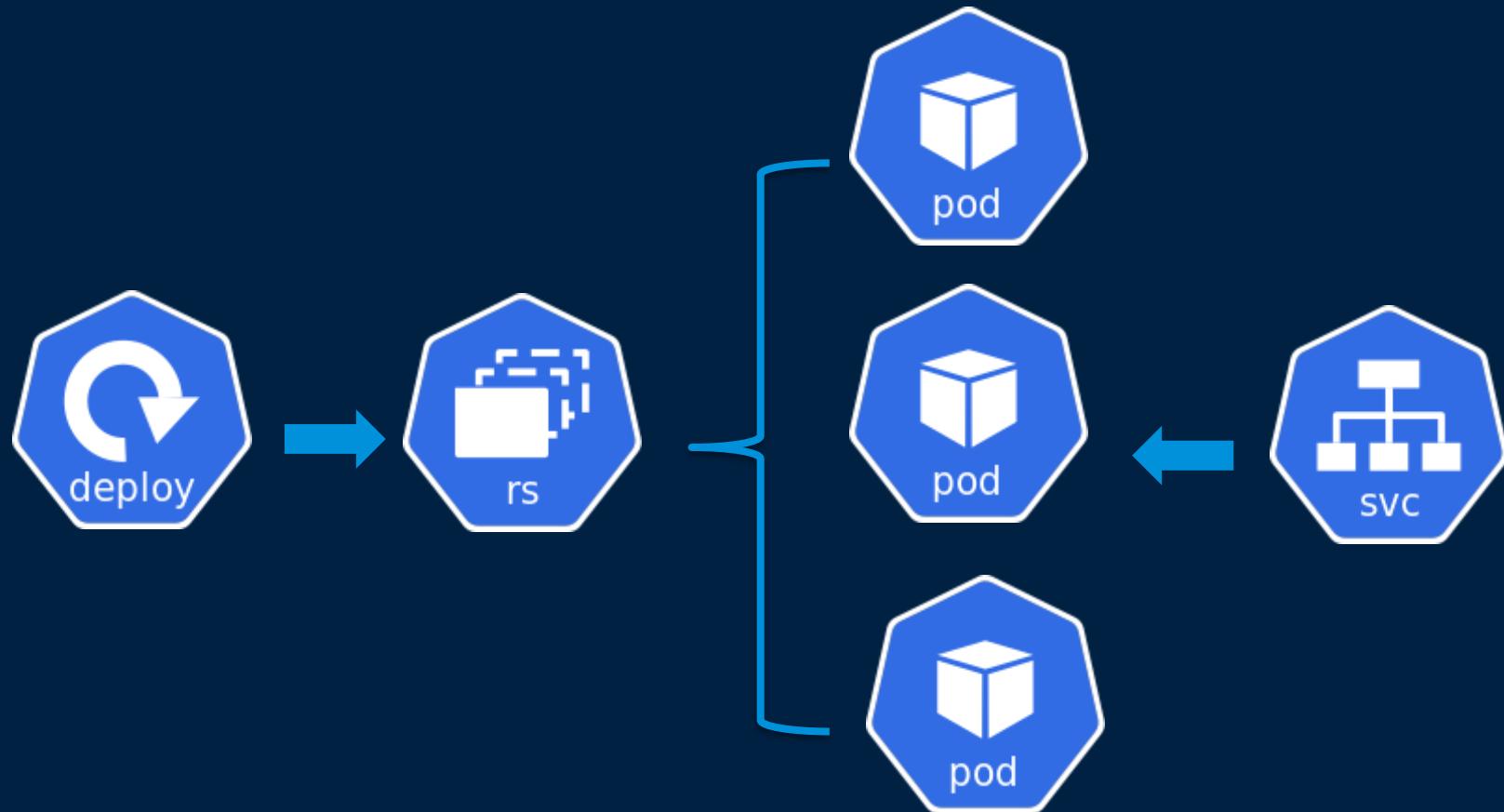
1 Backend (database)



Services

Consume pod based service without a need to know details
an abstraction that hides whether scale out is going on

Exposes a set of one or
more pods for use by
others as a unit –
sometimes called a
microservice



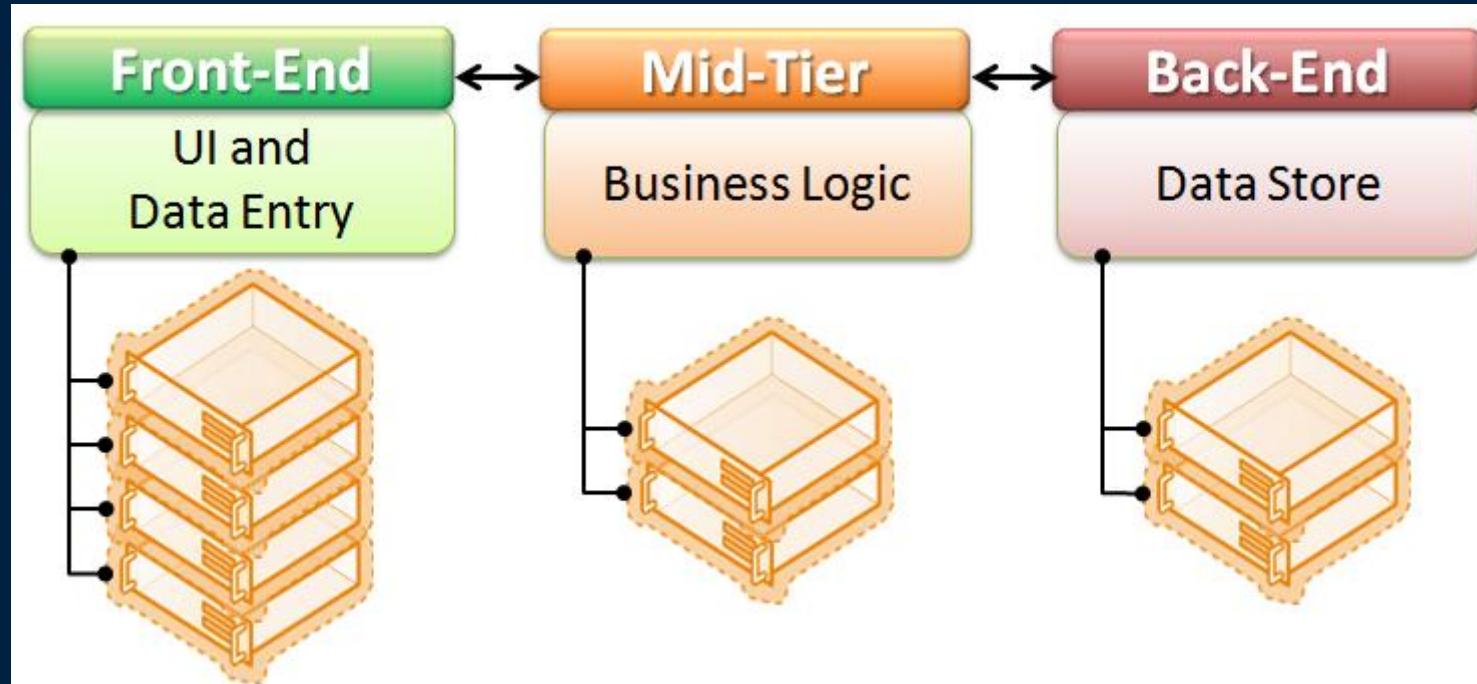
Ingress

Give services externally reachable URLs



What is n-tier

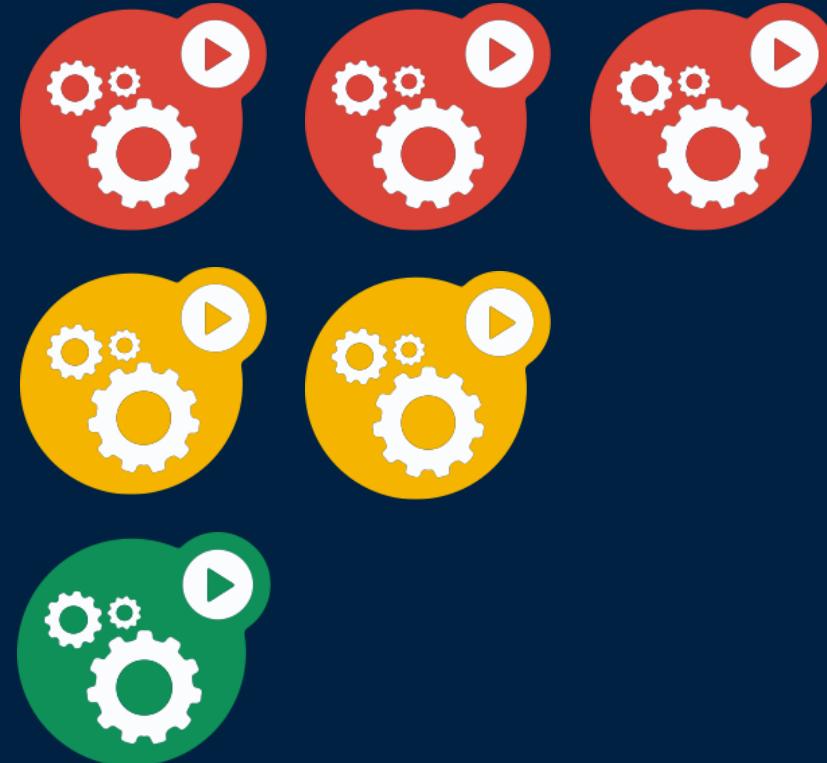
In this example n = 3



Desired state example

Scale out 3 tier application

There should be:
3 front-ends (UI)
2 Services
1 Backend (database)



Workshop time

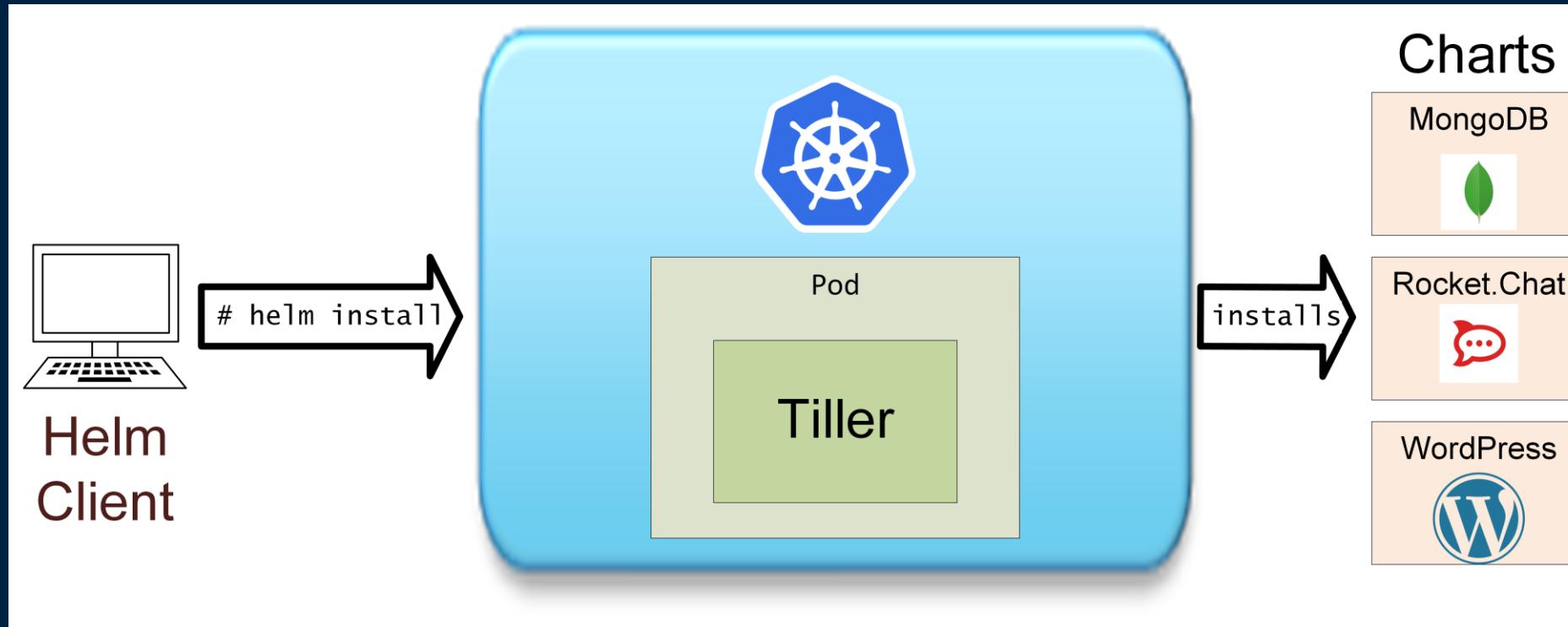
Helm

Package manager for
Kubernetes

Simplifies deployment and
management of Kubernetes
applications.



Helm Deployment: Components and steps



A typical Helm chart on github

```
rocketchat/
  Chart.yaml          # A YAML file containing information about the chart
  requirements.yaml   # OPTIONAL: A YAML file listing dependencies for the chart
  values.yaml         # The default configuration values for this chart
                      # Users can override a default during install
  charts/             # A directory containing any charts upon which this chart
                      # will generate valid Kubernetes manifest files.
  templates/NOTES.txt # OPTIONAL: A plain text file containing short usage notes
```

Helm install

Value overrides can be provided in a yaml file:

```
$ helm install -f myvals.yaml wordpress
```

values.yaml:

```
imageRegistry: "quay.io/deis"  
dockerTag: "latest"  
pullPolicy: "Always"  
storage: "s3"
```

myvals.yaml:

 storage: "gcs" 

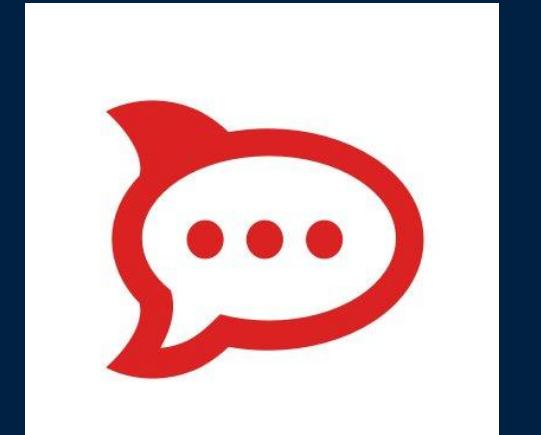
EFFECTIVE RESULT

```
imageRegistry: "quay.io/deis"  
dockerTag: "latest"  
pullPolicy: "Always"  
storage: "gcs"
```

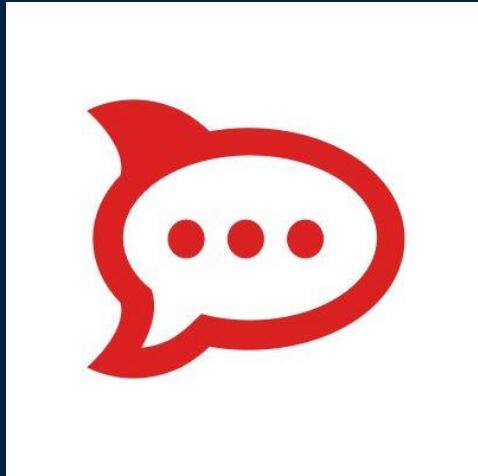
RocketChat deployment using Kubernetes with Helm

RocketChat is an open source multi-user messaging system that has many of the features of Slack

It is built using the Node.js based Meteor web app framework. Meteor uses a MongoDB NoSQL database as its backend to hold state.



What you will do



`myvals.yaml`

```
mongodb:  
  persistence:  
    enabled: true  
    storageClass: standard  
    accessMode: ReadWriteOnce  
    size: 32Gi  
  
persistence:  
  enabled: true  
  storageClass: xxx  
  accessMode: ReadWriteOnce  
  size: 32Gi
```

```
$ helm install -name chat1 -f myvals.yaml stable/rocketchat
```

Steve Wong
@cantbewong



Thank You