/usr/bin/whoami

- Ben Bromhead
    - CTO/Co-founder @ Instaclustr
- We provide managed Cassandra, Spark and Kafka in the cloud (AWS and others).
- We provide support (24x7x365) and services for customers in private data centers.
- Manage and support 3k+ nodes.

Agenda

- Containers and Kubernetes

- Kubernetes and state

- EKS (Amazon Elastic Container Service for Kubernetes)
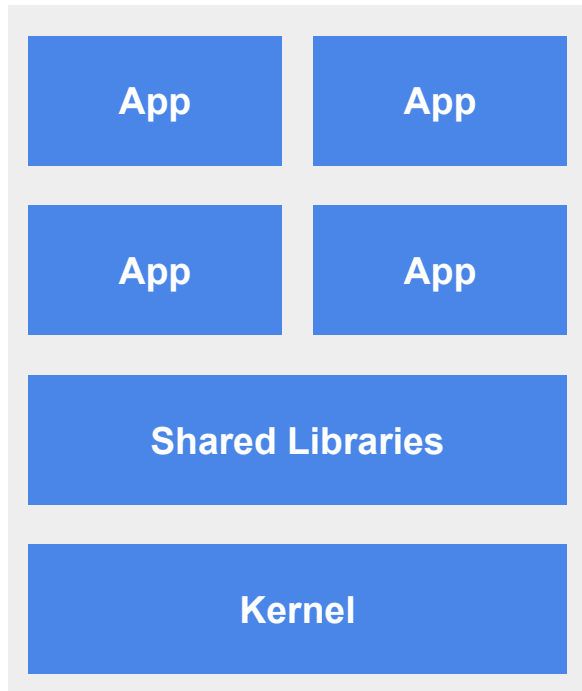
- Running Cassandra on EKS
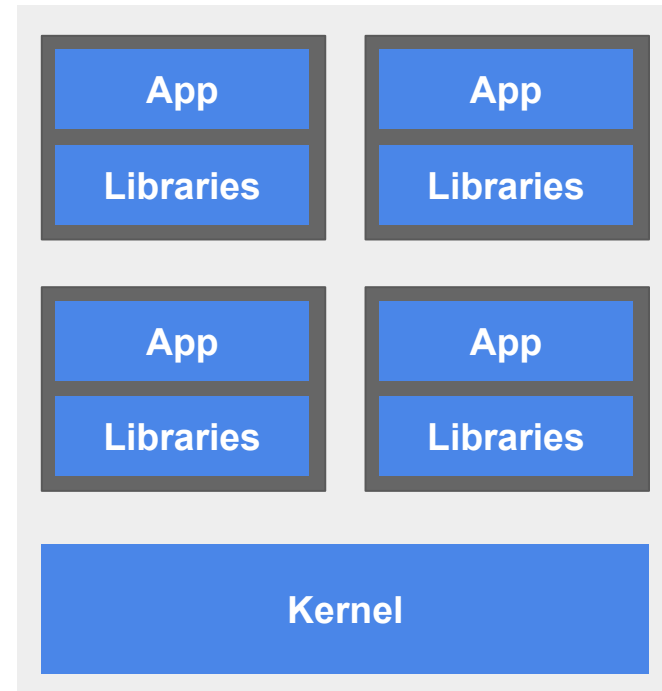
Containers - For managers

Essentially a way to bundle all the dependencies of a given process and keep it isolated…

Containers - For managers

instaclustr

Old: Applications on host

App	App
App	App

Shared Libraries

Kernel

New: Apps in Containers

App	App
Libraries	Libraries
App	App
Libraries	Libraries

Kernel

Containers - For managers

What does this actually get you

- Separation of concerns.
  - Developers can build an application/service and deliver it as a container that has defined interfaces.
  - Operators don't (generally) care what's inside the container.
- Reproducible artefacts that are the same across all environments.
  - That image you built on your laptop can be validated, tested and put into production with no changes.
- "Lightweight VMs"
- Simplifies package management
- A building block for a microservices architecture

Containers - For engineers

A container is made up of a few things:

- Process and resource isolation. Shares the host kernel but can't "see" other processes, can have CPU, IO and RAM limits, etc.

- Equivalent of a chroot environment. Bring your own userland. Need specific/unique libraries, services, binaries, or distro for your app? Done.

- Some sort of image. Contains files, binaries, libraries, for everything that will be run in the isolated environment. Typically an archive, but can also be a disk image (separate filesystem)

Containers

**Containers
(Linux)**

Cgroups
Namespaces
chroot env
AUFS
etc...

**Zones
(Solaris)**

First class
concept

**Jails
(BSD)**

First class
concept

VMs

First class
concept

from: https://twitter.com/jessfraz

Containers

This mix of components is not a bug, but a feature!

- VMs, Jails, Zones, etc. do everything for you, with minimal choice.

- Containers, everything is optional or pluggable

    - Want to allow two container to share the same network namespace? Sure go for it!

    - Don't want AUFS? Fine, use BTRFS.

    - Want a good filesystem? Mount a host directory into the container (yay XFS)

- Docker, rkt, containerd, Kubernetes, etc. all try to give you sane defaults so that containers work (somewhat) like VM/Jail/Zones.

Containers

instaclustr

Awesome, so a container is an isolated process that gets its own userspace, which has the side effect of making operations easier!

Kubernetes - For managers

A service that runs your containers for you, across lots of servers and tries to be smart about it.

Kubernetes - For managers

Officially: Kubernetes is an open-source platform designed to automate deploying, scaling, and operating application containers.

Kubernetes - For managers

It won the war:

EKS, Mesosphere, Docker Swarm

All support Kubernetes as a first class citizen

Kubernetes - For engineers

Kubernetes is made up of a few things:

- A database that manages state.

- Services that manage your system and move it from its current state to its intended state

- Tools, APIs, methods and formats for telling Kubernetes what state you want it to be in.

Kubernetes

What do you get with Kubernetes? A lot!

- Managing dependent/related containers

- Managing storage

- Distributing secrets

- Managing application health

- Replication

- Scaling

- Load balancing

- Updates

- RBAC!

- more

Fundamentals of Kubernetes

Before we get any deeper, an introduction to some Kubernetes specific terms

- K8s (industry approved abbreviation for Kubernetes)

- Pod - Represents a running process on your cluster.

- Controller - A control loop that resolves intended state to actual, the fundamental automation process in Kubernetes. E.g.

  - ReplicaSet - A object that defines how pods should be replicated. i.e. N pods for a ReplicaSet

 - Deployment controller - declarative updates for Pods and ReplicaSets.

Fundamentals of Kubernetes

Controllers are the primary method of mutating infrastructure in Kubernetes. All controllers use the following basic control loop:

- Observe - Gather the current state of the system
- Analyze - Determine the differences between the current state and intended state
- Act - Implement a single action to drive current state closer to intended state.
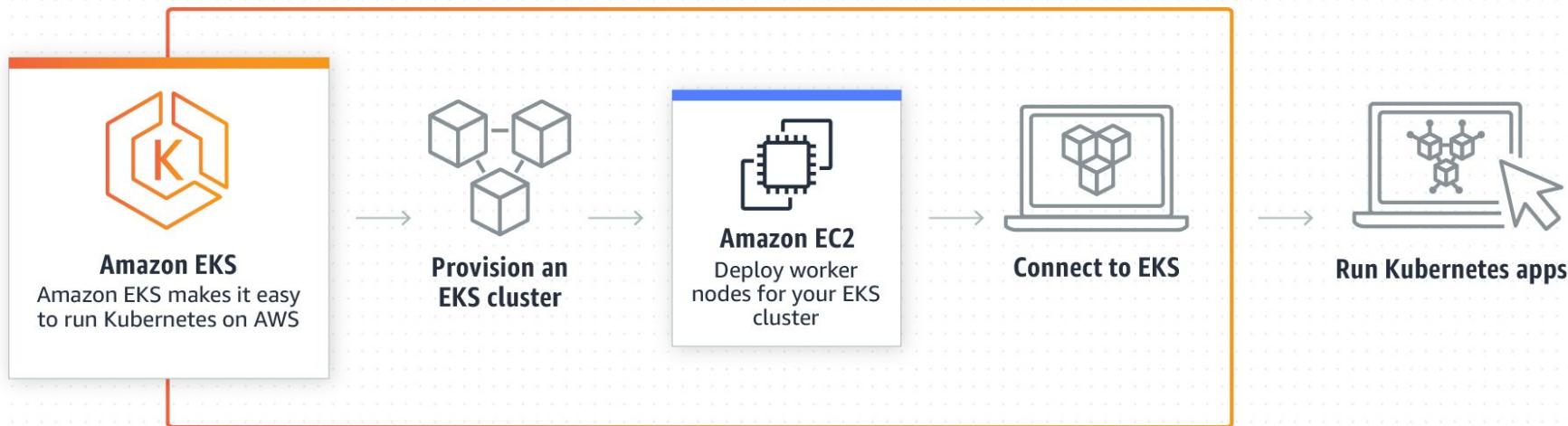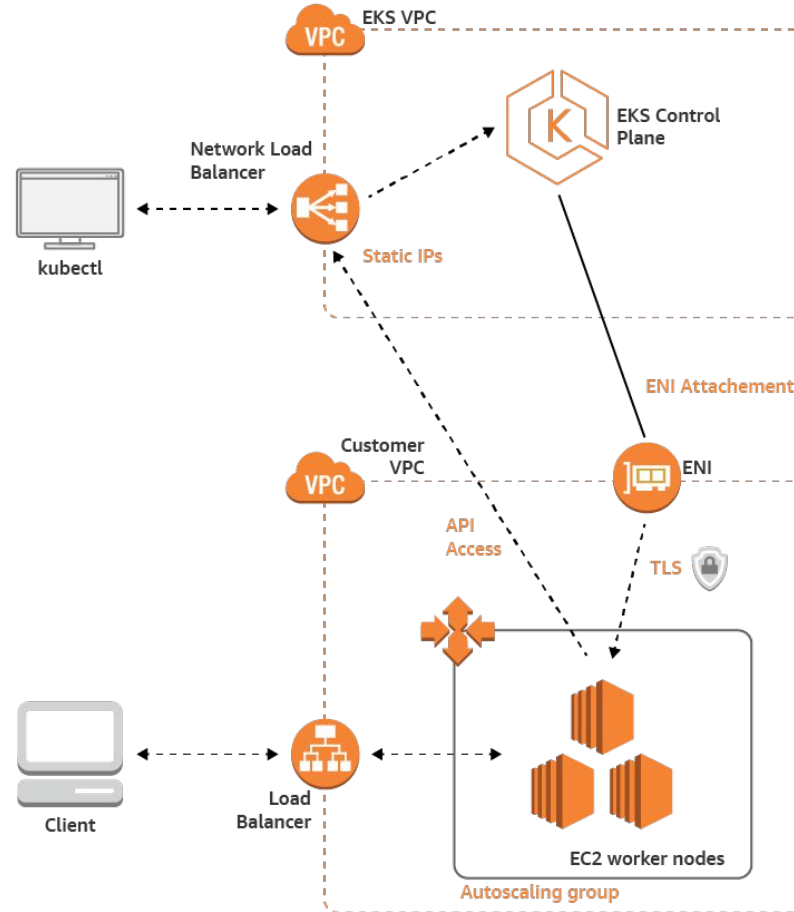
Fundamentals of Kubernetes
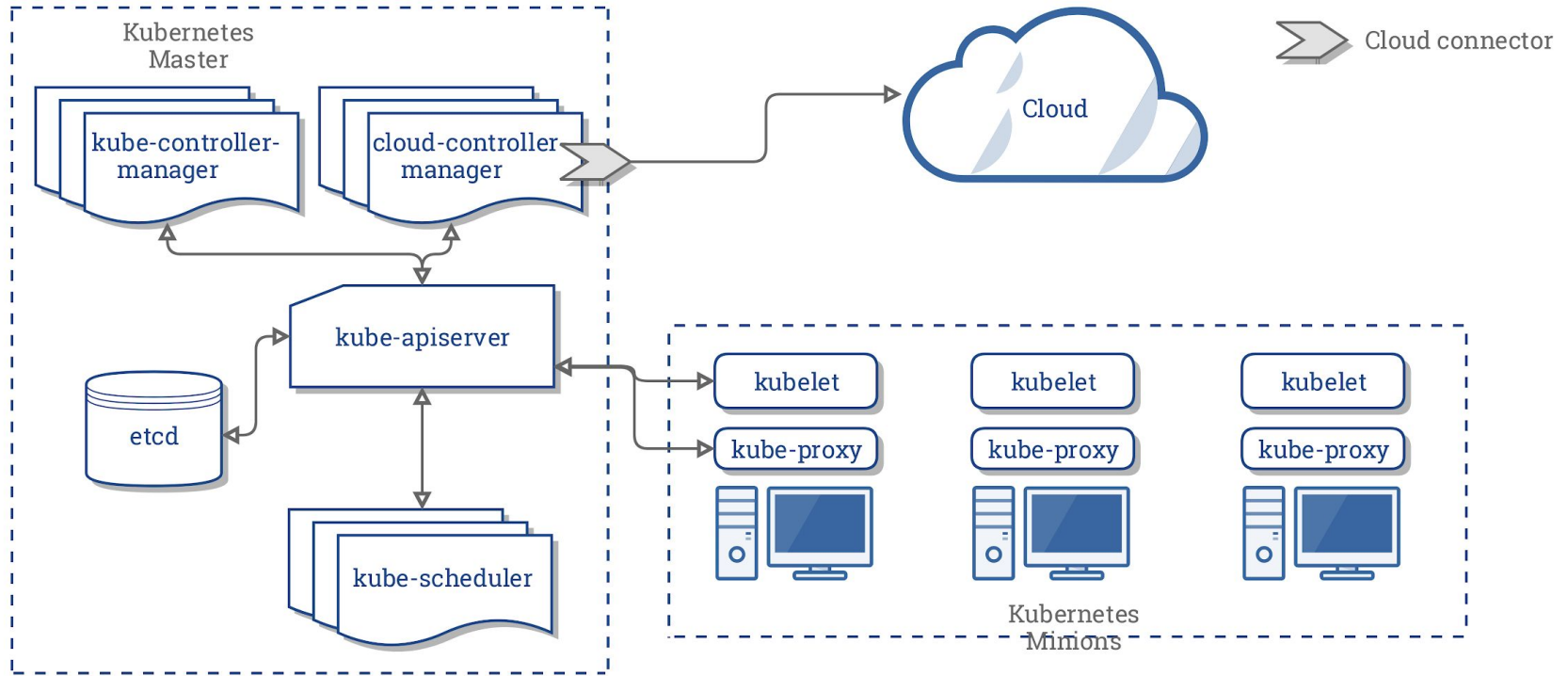
Kubernetes

What don't you get with Kubernetes?

- Someone to run it for you

- Getting Kubernetes to interact with your hardware/environment and other infra is hard.

- Enter Amazon Elastic Container Service for Kubernetes (EKS)

  - Bridge between Kubernetes and AWS

  - Kubernetes workers sit in an ASG

  - Built in IAM support

  - Built in native AWS networking (no overlay, full ENI support)

EKS

Amazon EKS
Amazon EKS makes it easy to run Kubernetes on AWS

Provision an EKS cluster

Amazon EC2
Deploy worker nodes for your EKS cluster

Connect to EKS

Run Kubernetes apps

EKS

So... what about Cassandra?

- As Kubernetes becomes a defacto orchestration API, people will (and do) want to run Cassandra on Kubernetes
- It's easy to get started, harder to run.
- For Instaclustr, Kubernetes does a lot of what we had to do in the past
  - It abstracts the environment we run in
  - Let's us focus on doing cool Cassandra things
  - Less focus on doing boring cloud things

Introducing Cassandra-operator
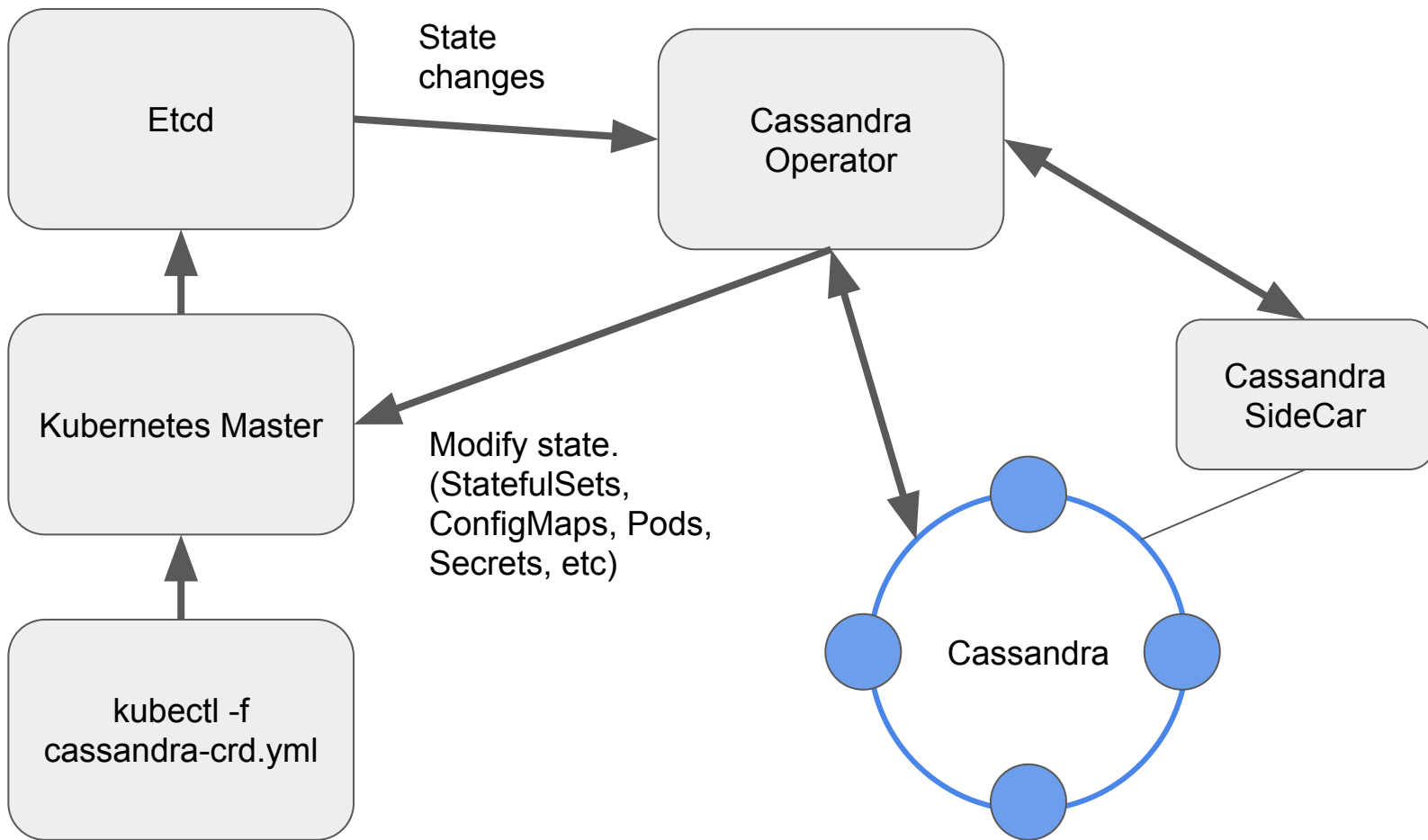
⟨⟩ instaclustr

- Let's build something that runs and **operates** Cassandra in Kubernetes
  - A controller
  - A Custom Resource Definition
- Cassandra-as-a-Service on top of Kubernetes
- EKS takes care of K8s management and AWS orchestration
- Cassandra operator manages Cassandra on K8s without needing to know about AWS (kinda, backups still rely on S3).

Awesome!...what does it get me?

instaclustr

- Operations "free" Cassandra
- Consistent, reproducible environments
- Best practices are built-in
- Let's your team focus on what matters

How does it work?

◌ instaclustr

Etcd

State changes

Cassandra Operator

Cassandra SideCar

Kubernetes Master

Modify state. (StatefulSets, ConfigMaps, Pods, Secrets, etc)

kubectl -f cassandra-crd.yml

Cassandra

Putting it all together

- Containers - Build, run and deploy things easier

- Kubernetes - Run, manage, operate things easier

- EKS - Kubernetes takes a lot of effort to run - get AWS to do it for us

- Cassandra-operator - run Cassandra super duper easy on EKS!

- Let's look at a demo (if time permits)!

Where to get it

instaclustr

https://github.com/instaclustr/cassandra-operator

- Pull requests accepted
- See the project issue tracker for current and future features