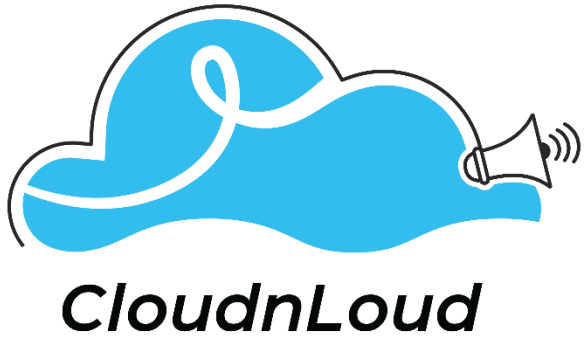
Amazon Elastic Kubernetes Service (EKS)



https://www.youtube.com/c/CloudnLoud

Fail More **Learn More** **Grow More** **Save Cancer Children**

Kubernetes on AWS before EKS

- Kubernetes is taking IT by storm, with tremendous adoption
- IT professionals need to be able to create and operate K8s clusters
- Traditionally, that meant deploy Kubernetes yourself:
  - Deploy Master Nodes
  - Deploy Etcd
  - Setup CA for TLS encryption
  - Setup monitoring / auto scaling / self-healing
  - Setup security such as authentication
  - Finally setup the worker nodes
- After all this you're finally done (just for the setup!)

Kubernetes on AWS before EKS

- On top of being hard to deploy, the Kubernetes Control plane is hard to manage and operate
- More and more teams want separate clusters in different environments:
  - Dev
  - Test
  - Pre-Prod
  - Prod
  - Finance Team
  - Reporting team
  - etc

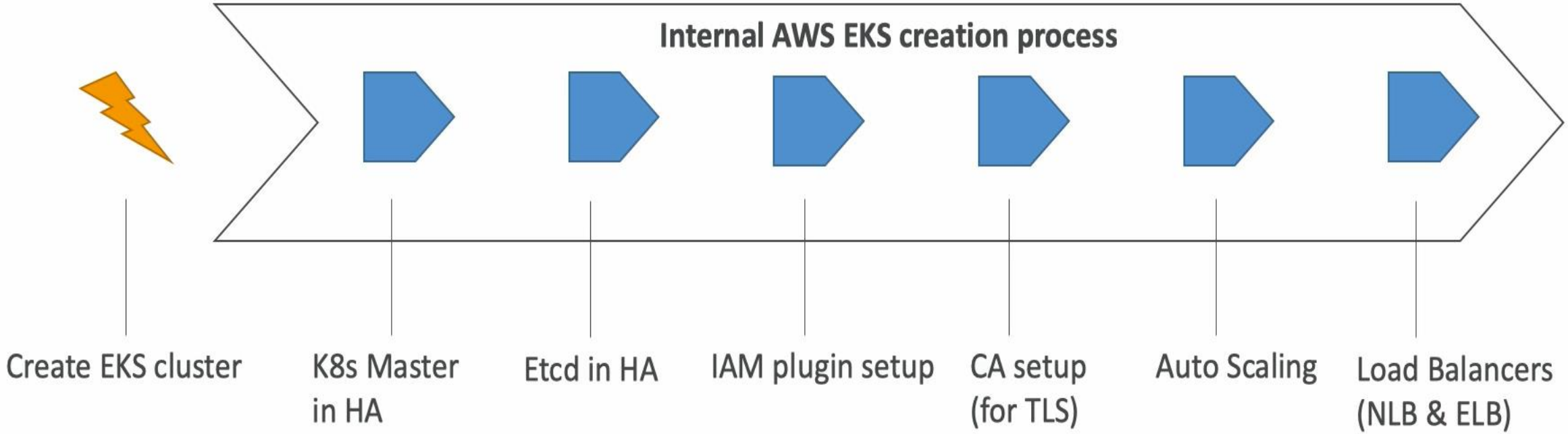
Enter Amazon EKS

Amazon EKS

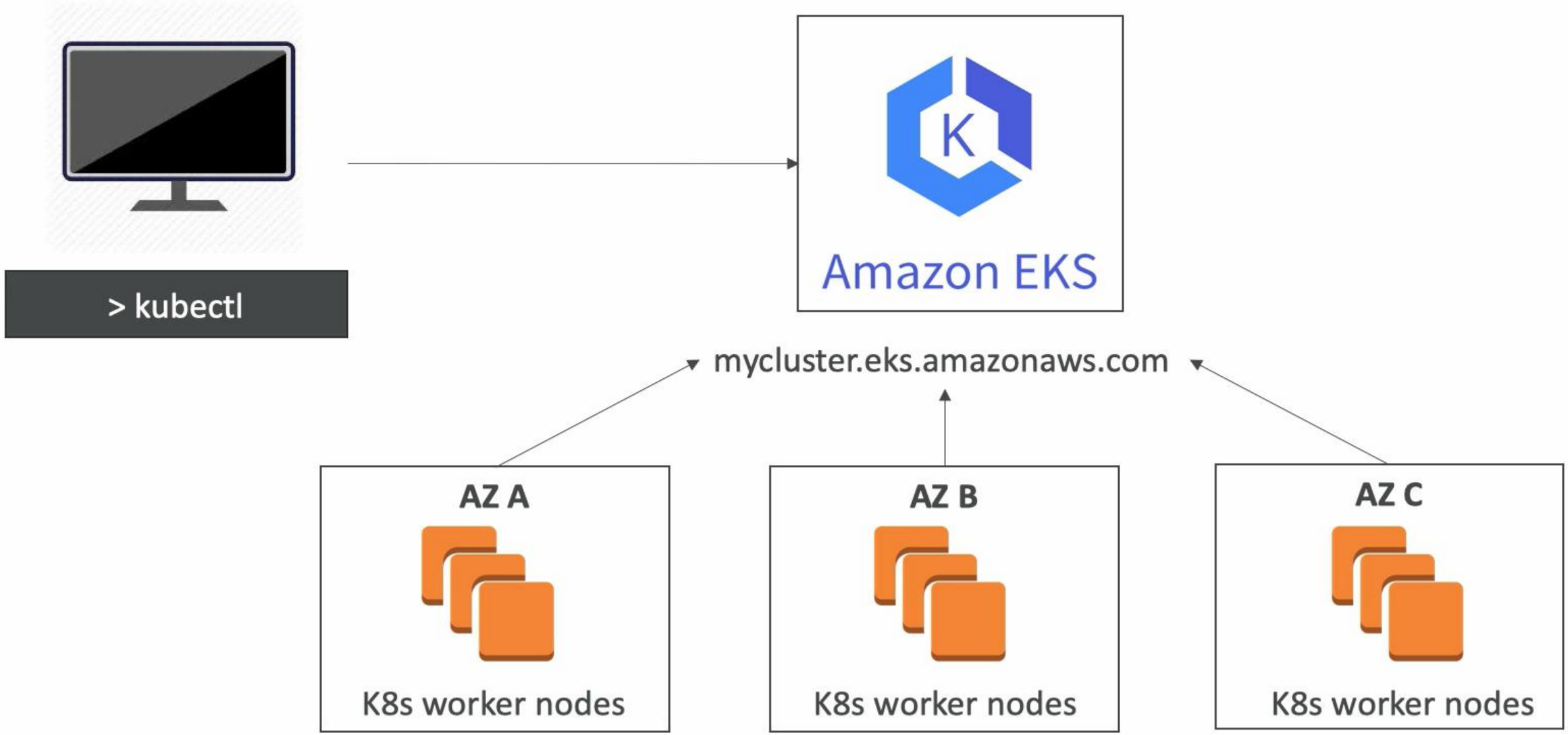
- All the hard Kubernetes setup is handled for you:
  - Master nodes in HA (High Availability)
  - Etcd ensemble in HA
  - API Server
  - KubeDNS
  - Scheduler
  - Controller Manager
  - Cloud Controller
- And you just need to deploy the Worker nodes and your applications!

EKS Setup Process (under the hood)

- EKS will setup and manage our Kubernetes masters:



In short, you get:



Deep integration with AWS

- API calls can be audited in CloudTrail
- Authentication is through IAM while authorization through RBAC
- CloudFormation templates to manage your clusters
- You can customize the AMI for your nodes
- Load Balancers, EBS Volumes, EFS, etc…
- Container registries on ECR
- Networking is handled with a per-pod IP address with attached ENI
- CLI integration

- But it's still the Kubernetes Open Source project!

Amazon EKS use cases

- You can now create a cluster per use case, easily!

- Microservices in containers
- Platform as a Service / Website
- Migrate from on-premise cluster to cloud Kubernetes cluster
- Machine learning cluster (support for GPU instances!)
- Your imagination!

Course activities

- Deploy your EKS cluster using CloudFormation
- Scale your Kubernetes cluster
- Setup kubectl properly to access your cluster
- Learn how EKS works under the hood and its integrations with AWS
- Setup administration using the Kubernetes Dashboard
- Deploy a stateless application on EKS and expose it with a public ELB
- Deploy a stateful application on EKS and bind it with EBS volumes
- Deploy a stateful application (such as Wordpress) with EFS network drives
- Manage your Kubernetes cluster using the AWS CLI and eksctl CLI

Important: Pre-requisites

You need to know the basics of Kubernetes

- Kubernetes API Server
- Kubelet
- Master & Worker Node, Pods
- Deployments, Services
- Networking within Kubernetes

Important: Pre-requisites

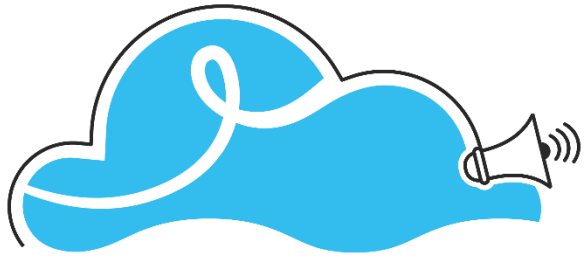
You need to know the basics of Kubernetes

- Kubernetes API Server
- Kubelet
- Master & Worker Node, Pods
- Deployments, Services
- Networking within Kubernetes

You need to know the basics of AWS and have an AWS account ready

- VPC, Subnets
- IAM
- EC2, EBS
- Load Balancers
- Security Groups

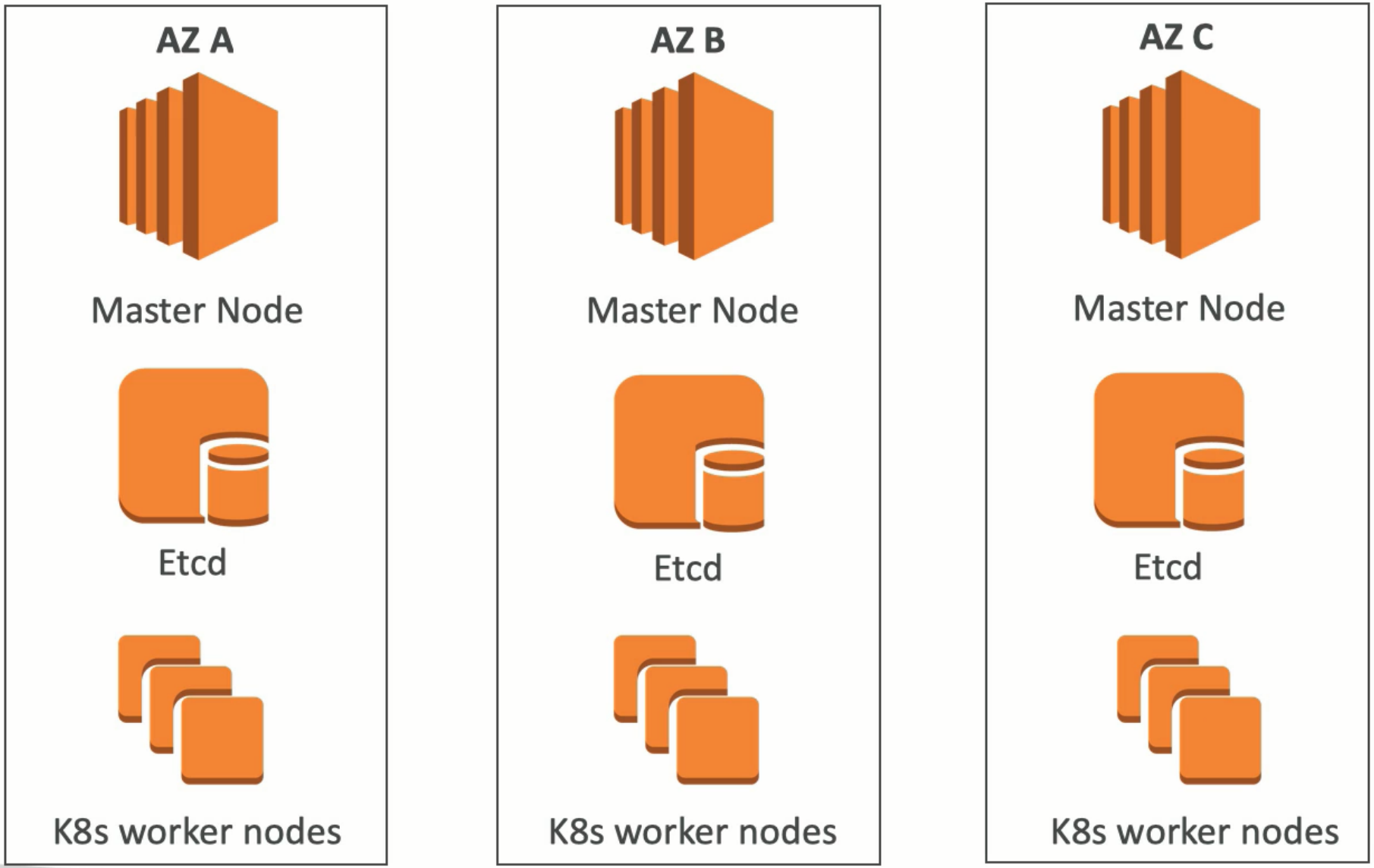
AWS EKS Cluster Setup



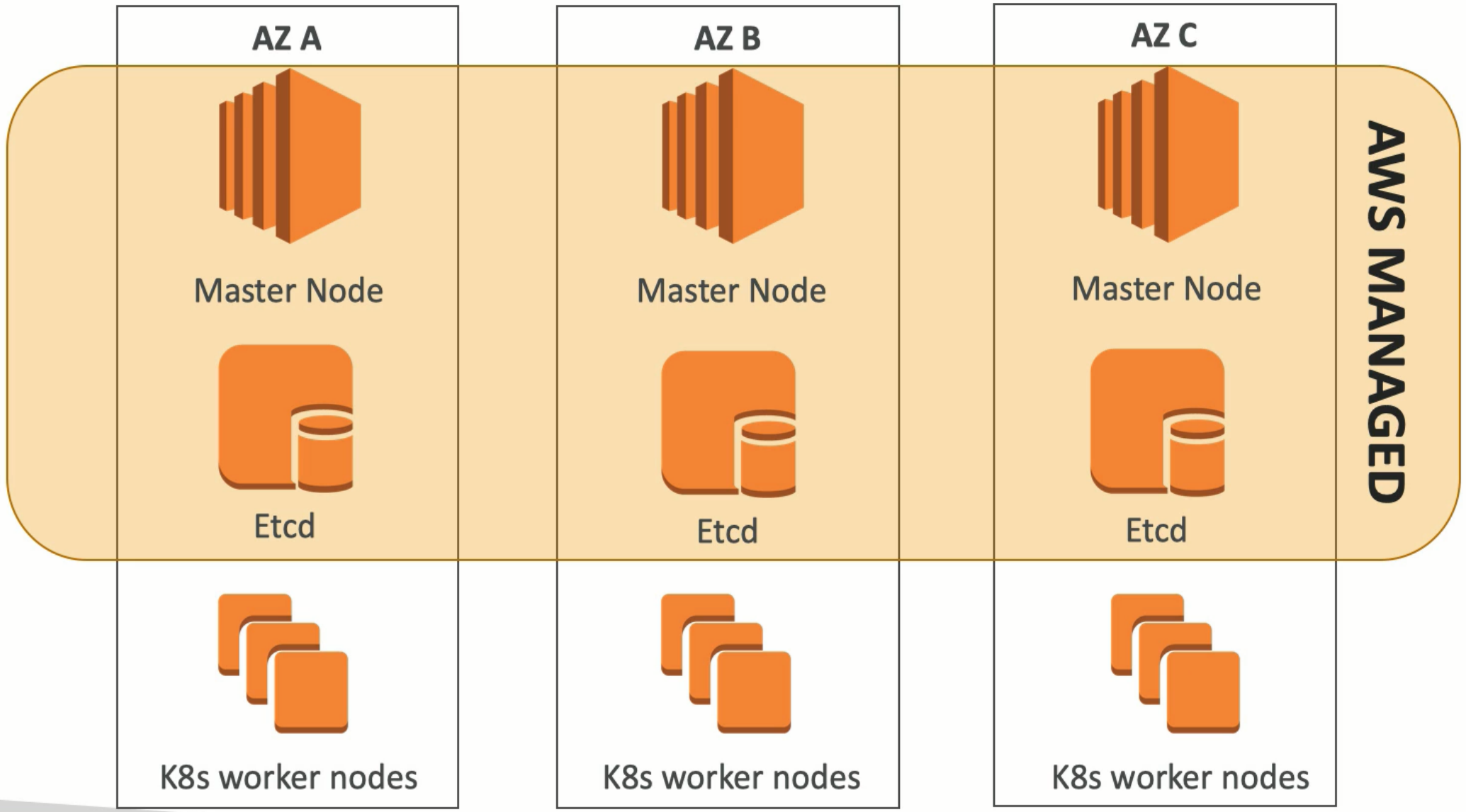
https://www.youtube.com/c/CloudnLoud

Fail More **Learn More** **Grow More** **Save Cancer Children**

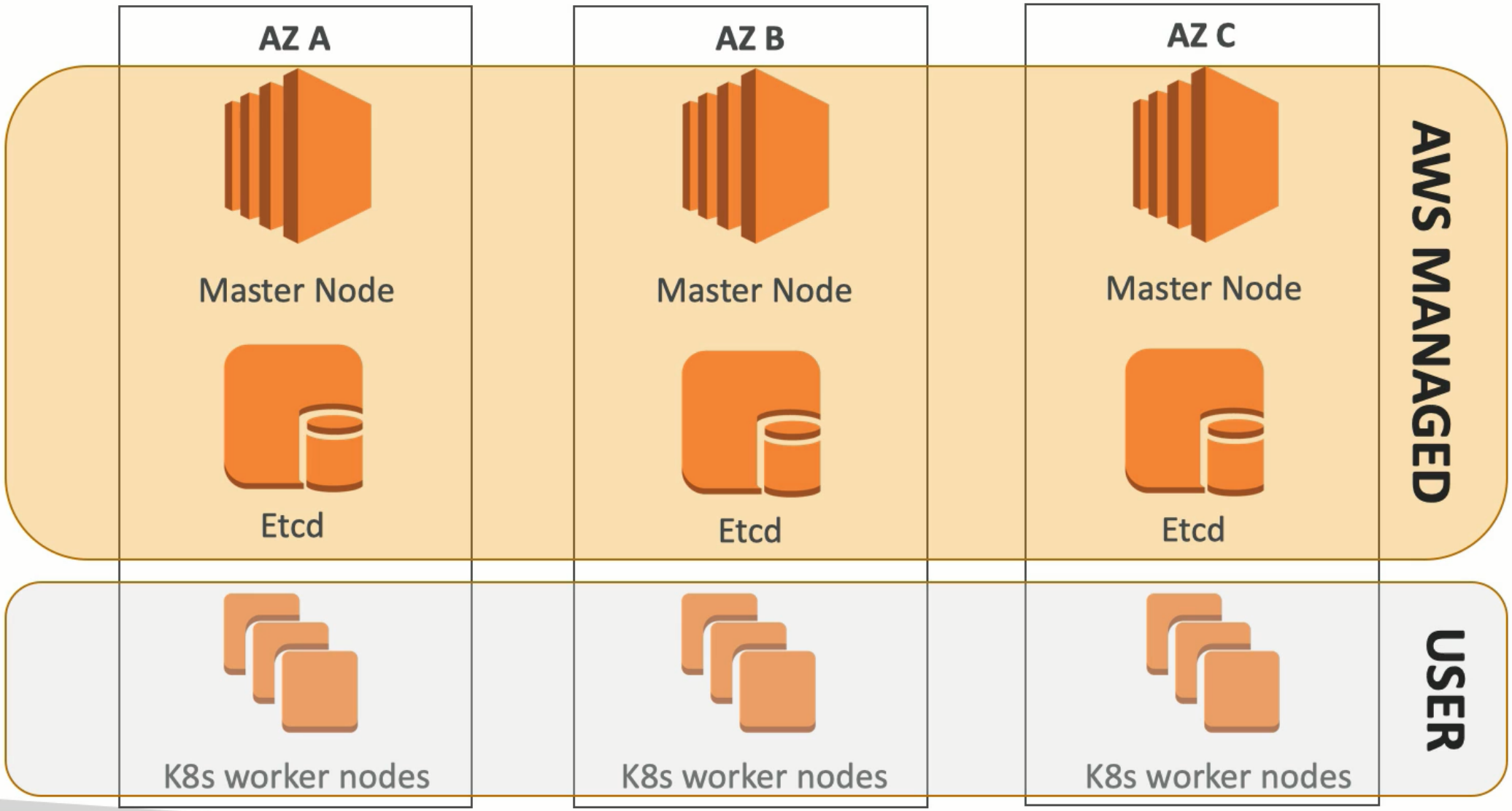
Amazon EKS Architecture



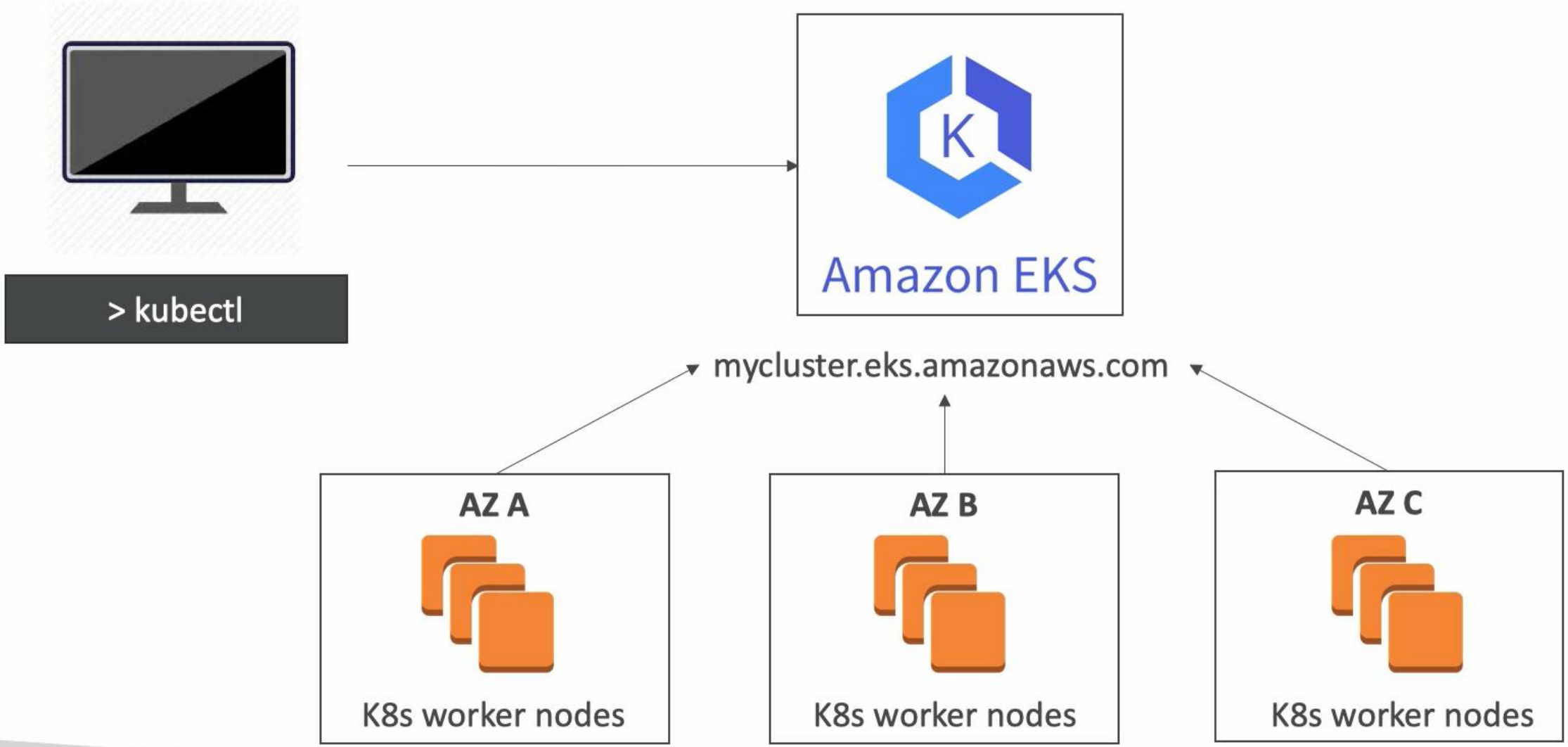
Amazon EKS Architecture



Amazon EKS Architecture



Amazon EKS Architecture



EKS Pricing - Warning

Important:

- Each EKS cluster costs $0.20 USD per hour
- That's $144 USD per month!
- There's no free tier for EKS
- On top of it, you will pay for the worker nodes EC2 instances, EBS volumes, Load Balancers, etc.

Bottom line:

- If you do this tutorial, try to do it very quickly to limit your cost
- Otherwise, just watch the videos but you can't practice

Create EKS cluster via eksctl (https://eksctl.io/)

- **eksctl**
  - created by Weaveworks, then promoted to official AWS EKS cli (announcement summer 2019)
  - creation & management of EKS cluster

- **initial cluster**
  - region: us-east-1
  - one nodegroup
  - 3 worker nodes, type t2.small
  - ssh access

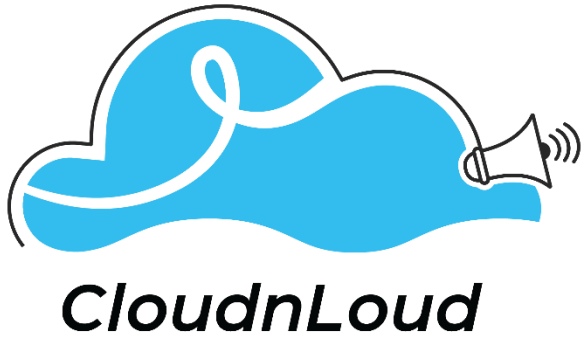
Create EKS cluster via eksctl (https://eksctl.io/)

- **eksctl**
  - created by Weaveworks, then promoted to official AWS EKS cli (announcement summer 2019)
  - creation & management of EKS cluster

- **initial cluster**
  - region: us-east-1
  - one nodegroup
  - 3 worker nodes, type t2.small
  - ssh access

eksctl will create
1.) a VPC
2.) 2 subnets in 2 AZs
in our setup

Deploy & Inspect a Stateless app



https://www.youtube.com/c/CloudnLoud

Fail More **Learn More** **Grow More** **Save Cancer Children**

Deploy & inspect a stateless app

- We will deploy the guestbook app at: https://github.com/kubernetes/examples/tree/master/guestbook

Deploy & inspect a stateless app

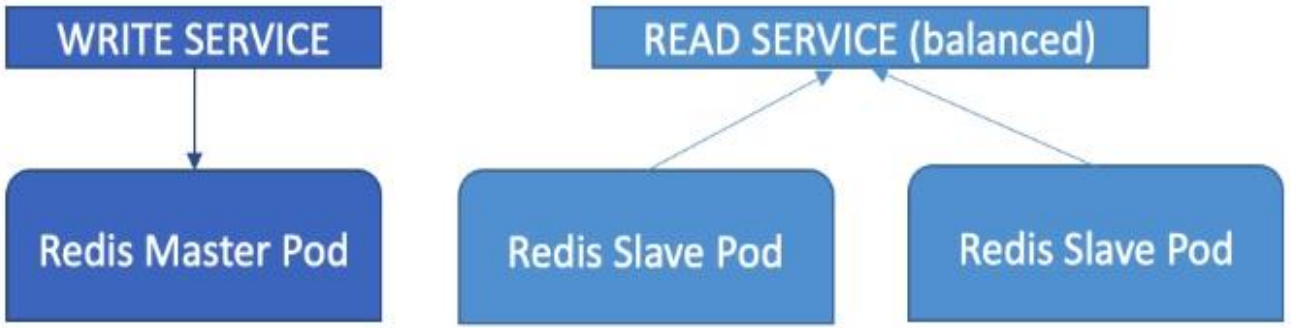
- We will deploy the guestbook app at: https://github.com/kubernetes/examples/tree/master/guestbook

- In this section we will…
- Deploy backend resources
- Deploy frontend resources
- Scaling Pods up/down
- Perform some *chaos* testing

- Our application will be accessible from an AWS Load Balancer!

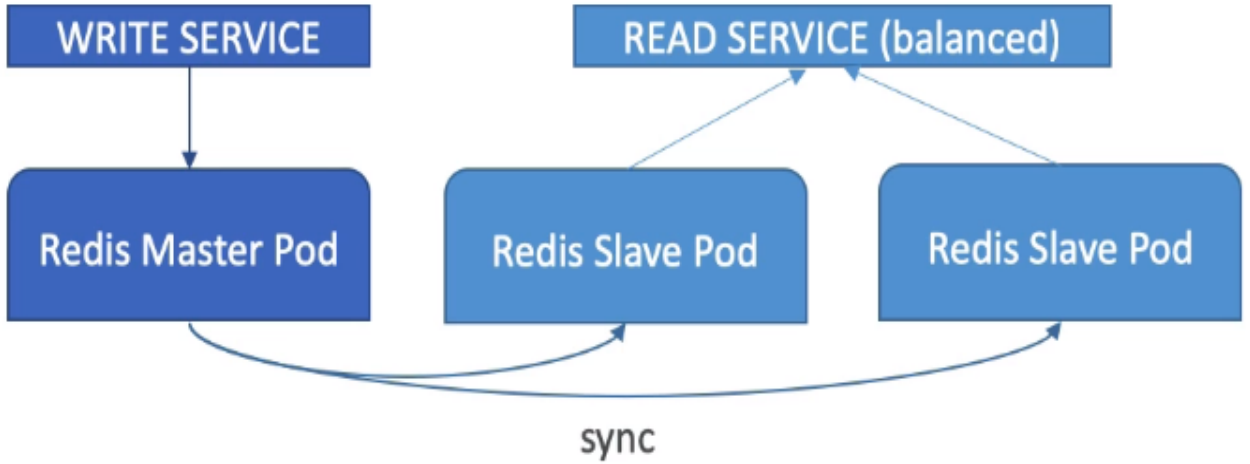
Deploy & inspect a stateless app

- Backend: Redis
  - Single Master (WRITEs)
  - Multi Slaves (READ)



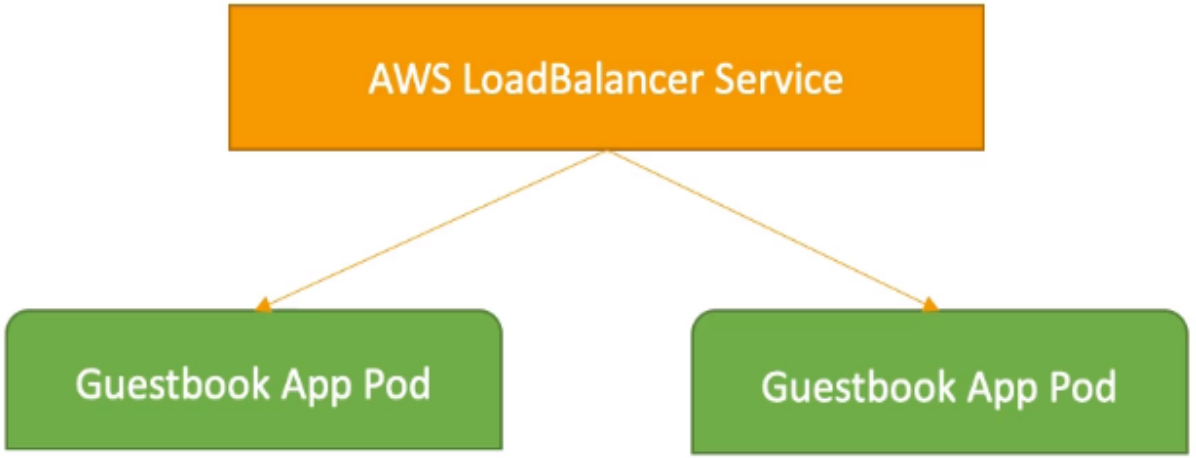
Deploy & inspect a stateless app

- Backend: Redis
  - Single Master (WRITEs)
  - Multi Slaves (READ)
  - Slaves sync continuously from Master

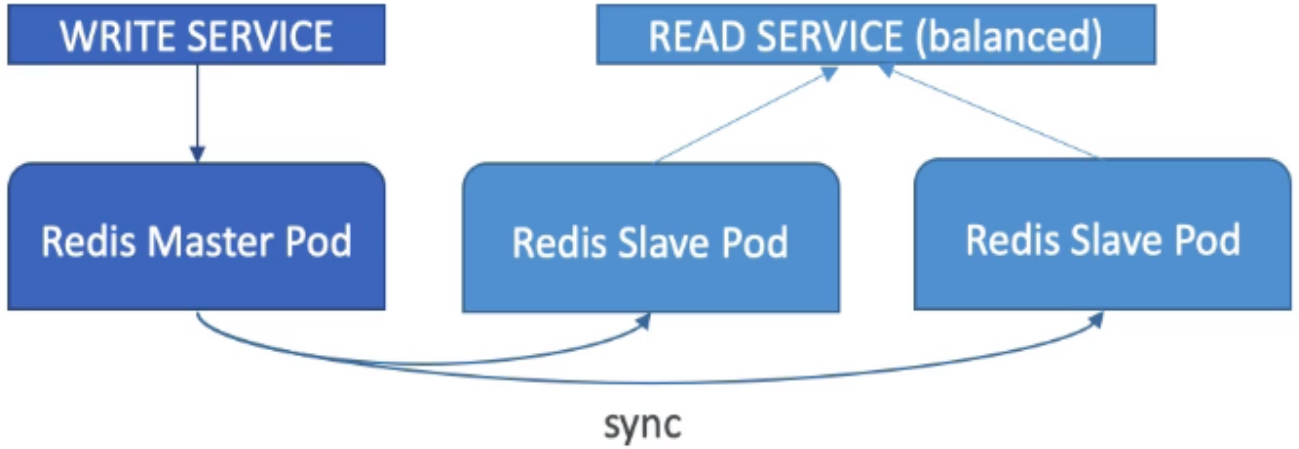


Deploy & inspect a stateless app

- Frontend: PHP App
  - Load balanced to public with ELB

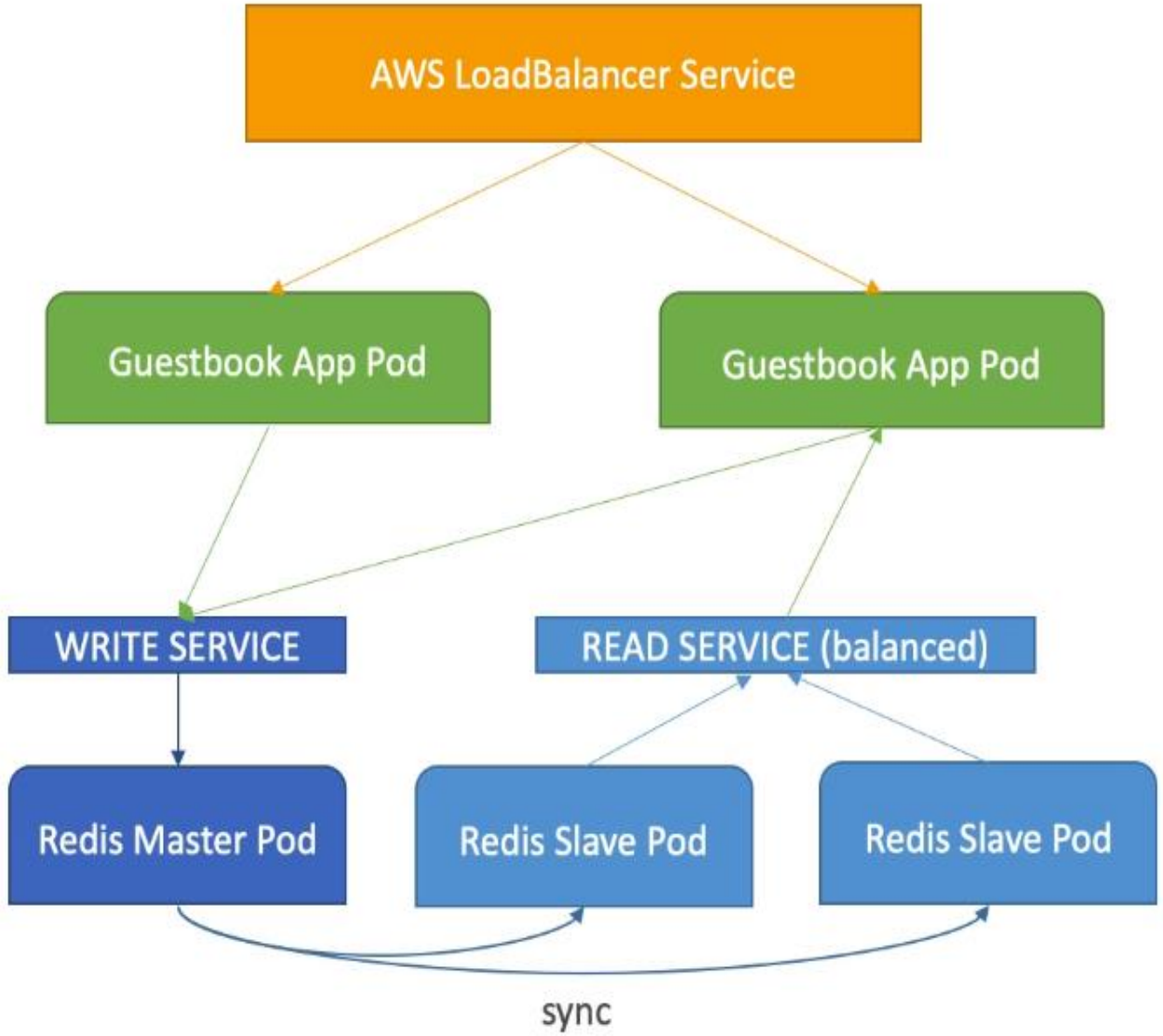


- Backend: Redis
  - Single Master (WRITEs)
  - Multi Slaves (READ)
  - Slaves sync continuously from Master



Deploy & inspect a stateless app

- Frontend: PHP App
  - Load balanced to public with ELB
  - *Read* balanced over multiple slave DBs
  - *Write* req to single Master DB
- Backend: Redis
  - Single Master (WRITEs)
  - Multi Slaves (READ)
  - Slaves sync continuously from Master



Backend Deployment

Let's create…

- Redis Master pod
- Redis Master service
- Redis Slave pods
- Redis Slave service

And…

- Inspect AWS networking (ENI)

Elastic Networking Interface