Google Cloud Platform Enterprise Services

Janakiram MSV

Learning Objectives

- Google Cloud IoT
- API Management
- Hybrid and Multi-cloud
- Migration

Google Cloud IoT

Google Cloud IoT

Cloud IoT Core

Edge TPU

Google Cloud API Management

API Management

Apigee API Platform

API Analytics

Cloud Endpoints

Google Cloud Hybrid & Multi-Cloud Services

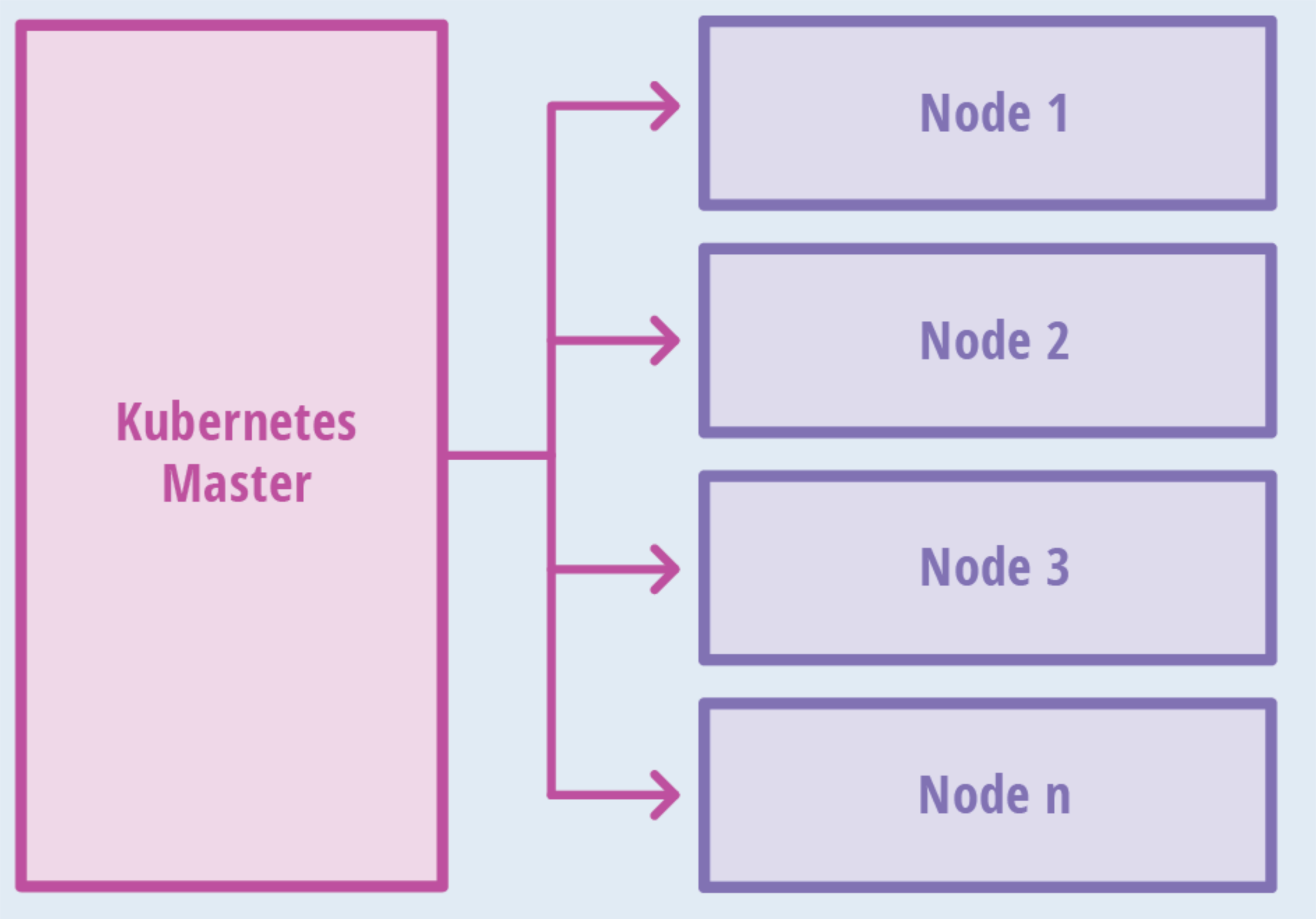
Hybrid & Multi-Cloud

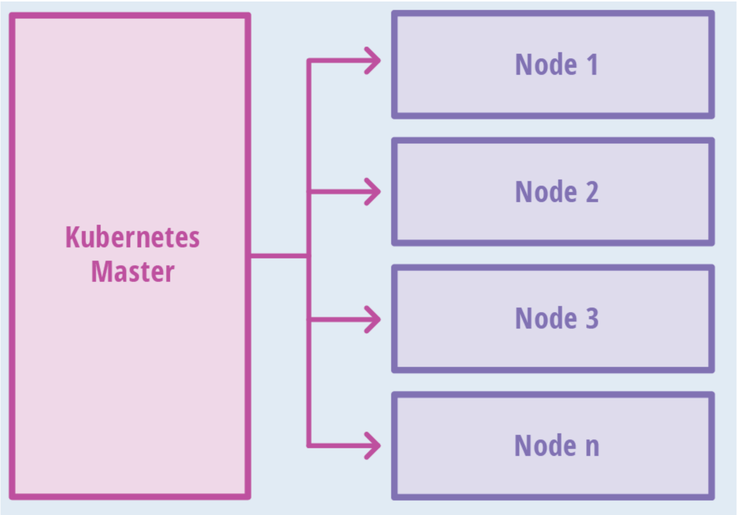
Traffic Director

Stackdriver

GKE On-Prem

Anthos


Kubernetes Master
Node 1
Node 2
Node 3
Node n



Kubernetes Master

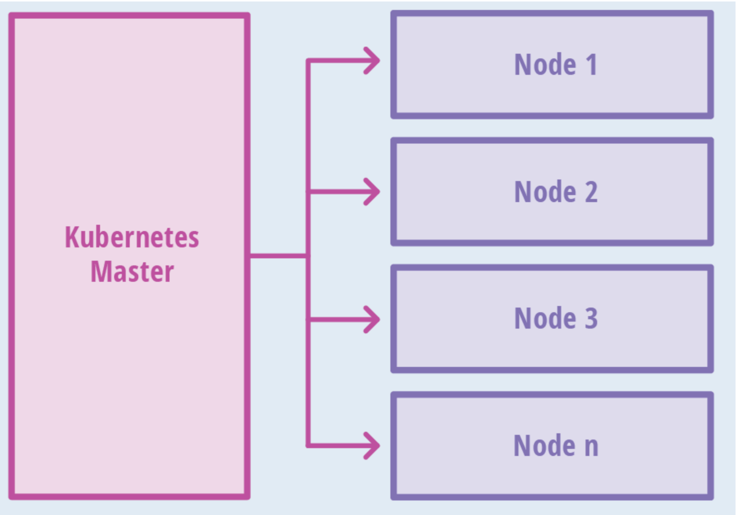
Node 1

Node 2

Node 3

Node n

Dev/Test - Datacenter



Kubernetes Master

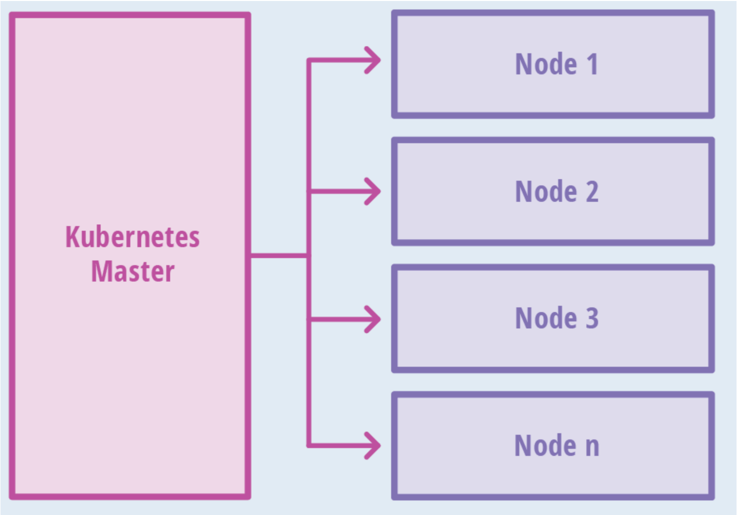
Node 1

Node 2

Node 3

Node n

Staging - AWS



Kubernetes Master

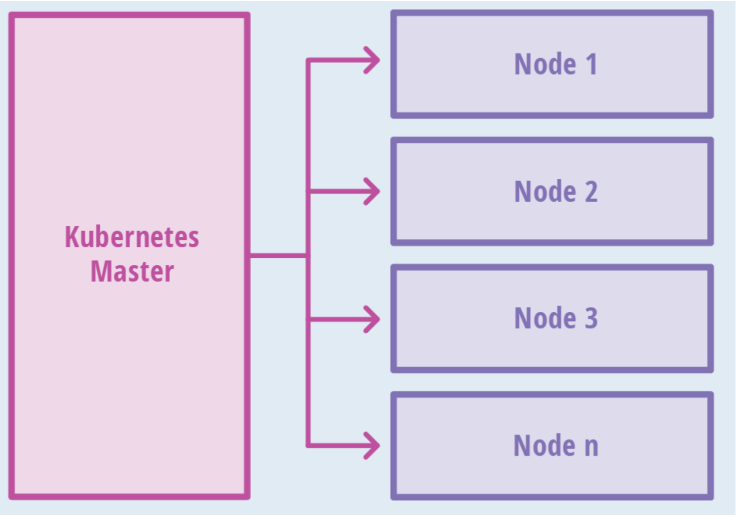
Node 1

Node 2

Node 3

Node n

UAT - Azure



Kubernetes Master

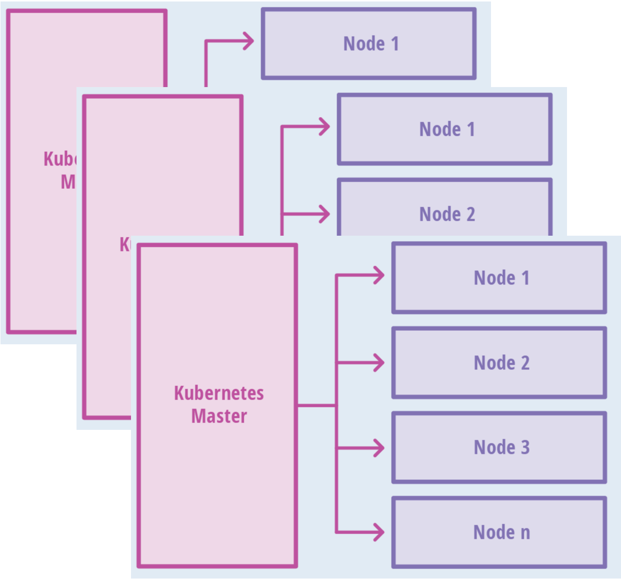
Node 1

Node 2

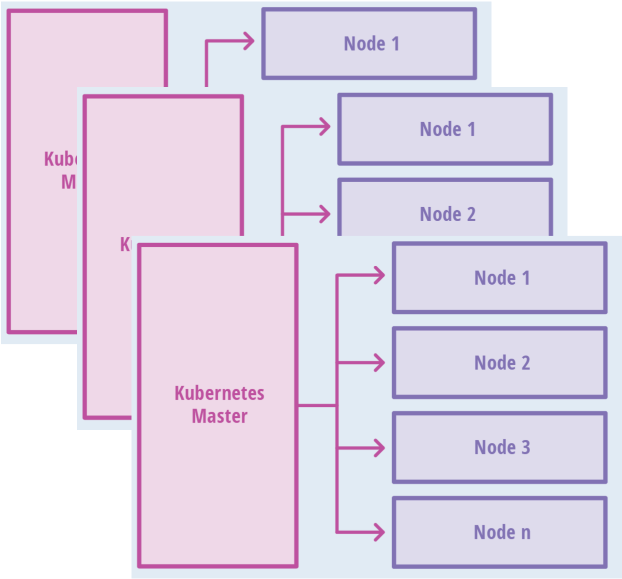
Node 3

Node n

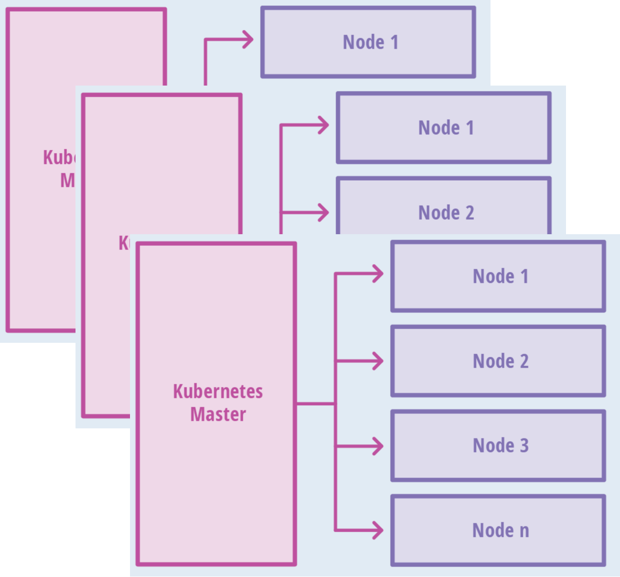
Prod - GCP



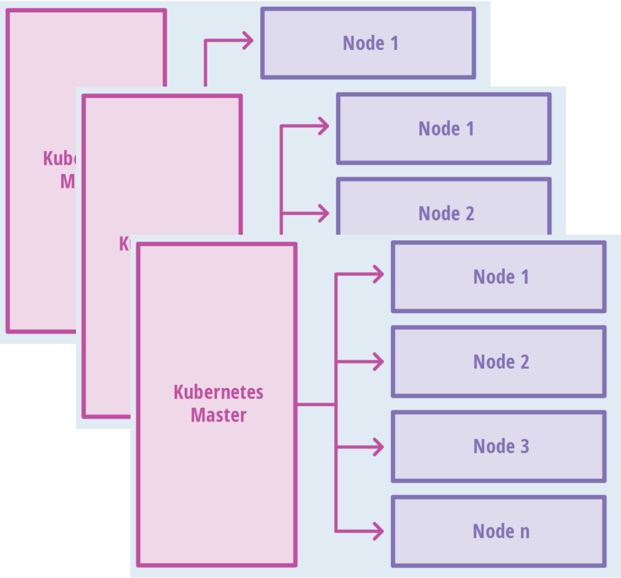
Finance: On-prem



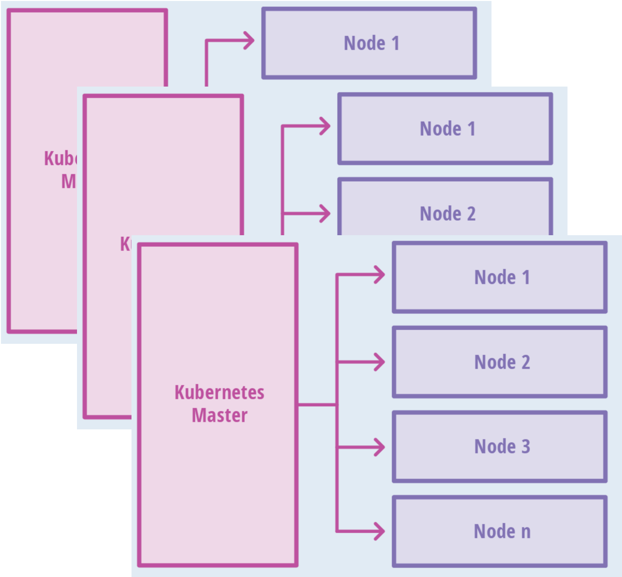
Operations: On-prem + AWS



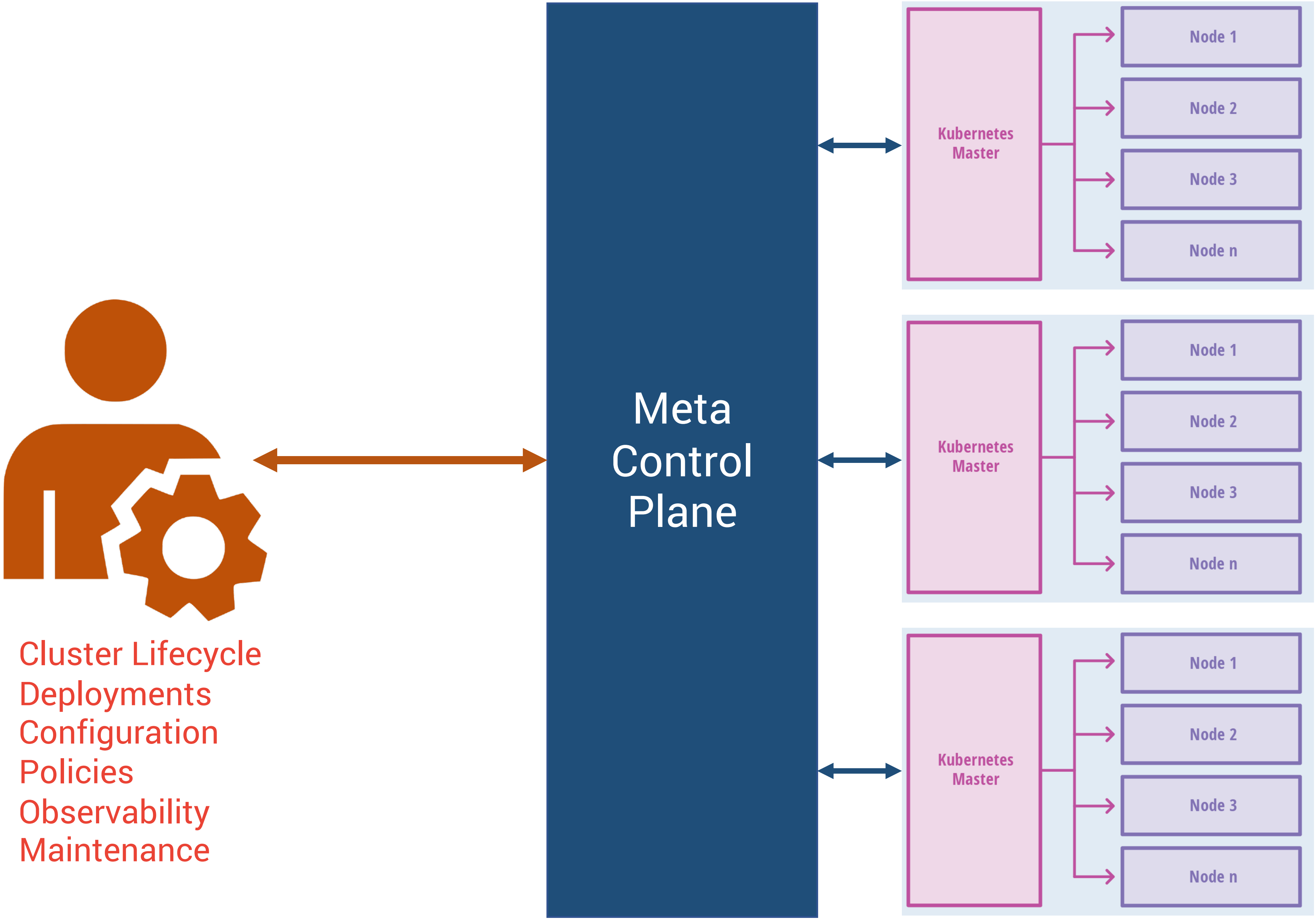
Legal: On-prem



HR: On-prem



IT: On-prem + GKE



Meta Control Plane

Kubernetes Master

Node 1

Node 2

Node 3

Node n

Kubernetes Master

Node 1

Node 2

Node 3

Node n

Kubernetes Master

Node 1

Node 2

Node 3

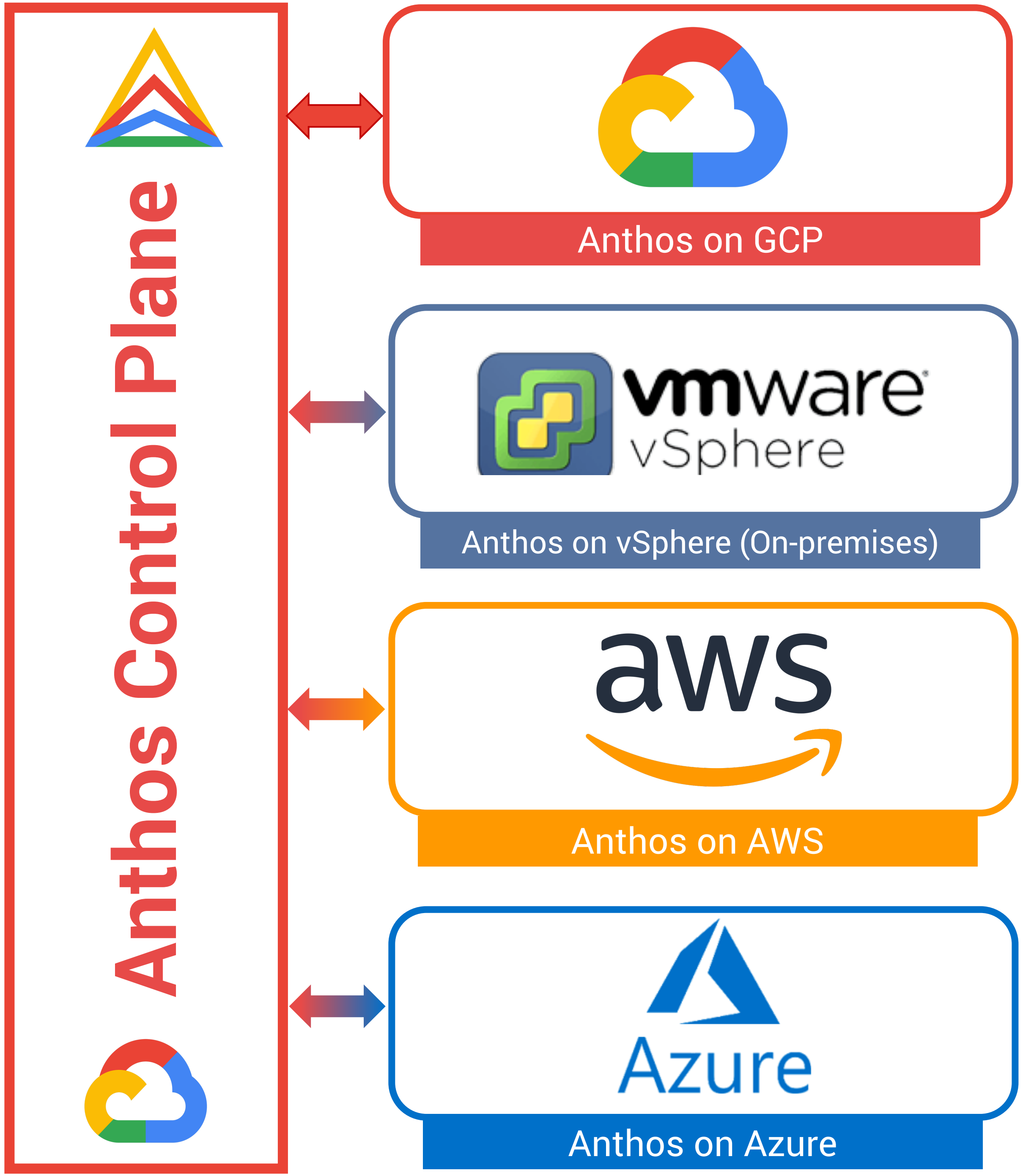
Node n

- Cluster Lifecycle
- Deployments
- Configuration
- Policies
- Observability
- Maintenance

What is Anthos?

- Google’s multi-cloud and hybrid cloud platform based on Kubernetes
- Enables customers to run managed Kubernetes service (GKE) in a variety of environments
- Anthos can be deployed in
  - Google Cloud
  - vSphere (on-premises)
  - Amazon Web Services
  - Microsoft Azure
- Non-GKE Kubernetes clusters can be attached to Anthos
- Delivers centralized management and operations for Kubernetes clusters running diverse environments

Anthos Managed Clusters



Anthos Control Plane

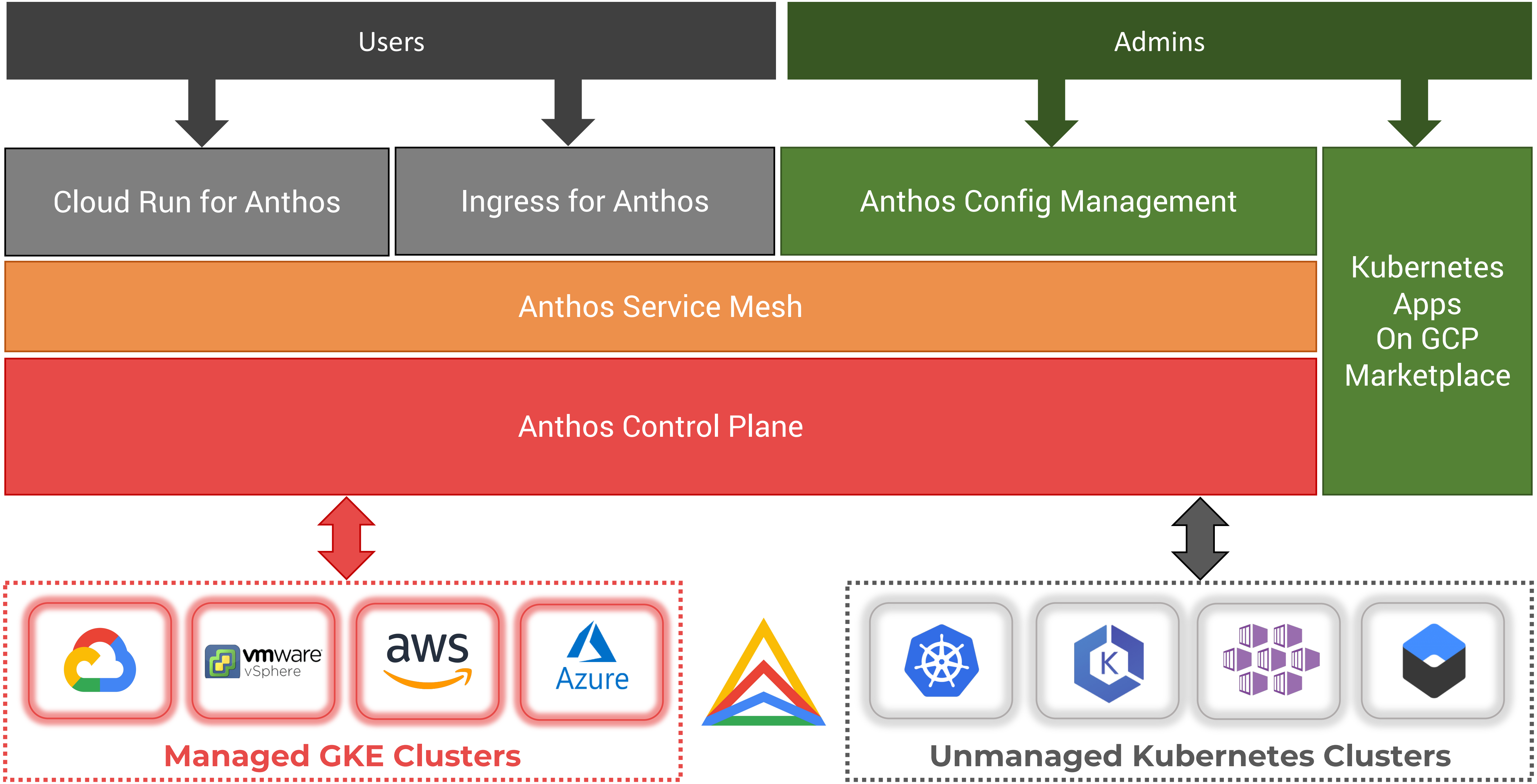
Anthos on GCP

Anthos on vSphere (On-premises)

Anthos on AWS

Anthos on Azure

Anthos Big Picture



Google Cloud Platform Fundamentals

Lab Guide for Anthos

```
# Run the below commands in the macOS Terminal

export PROJECT_ID=<PROJECT ID> # Replace this with your GCP project ID
export REGION=<REGION ID> # Replace this with a valid GCP region

gcloud config set project $PROJECT_ID
gcloud config set compute/region $REGION

# Enable APIs
gcloud services enable \
     container.googleapis.com \
     gkeconnect.googleapis.com \
     gkehub.googleapis.com \
     cloudresourcemanager.googleapis.com

# Launch GKE Cluster
gcloud container clusters create cloud-cluster \
    --machine-type=n1-standard-1 \
    --num-nodes=1

# Launch Minikube. Refer to the docs at https://minikube.sigs.k8s.io/docs/
minikube start

# Create GCP Service Account
gcloud iam service-accounts create anthos-hub
```

Google Cloud Platform Fundamentals

```
# Add IAM Role to Service Account
gcloud projects add-iam-policy-binding $PROJECT_ID \
 --member="serviceAccount:anthos-hub@$PROJECT_ID.iam.gserviceaccount.com" \
 --role="roles/gkehub.connect"

# Download the Service Account JSON Key
gcloud iam service-accounts keys create "./anthos-hub-svc.json" \
  --iam-account="anthos-hub@$PROJECT_ID.iam.gserviceaccount.com" \
  --project=$PROJECT_ID

# Register cluster with Anthos
URI=` gcloud container clusters list --filter='name=cloud-cluster' --uri`

gcloud container hub memberships register cloud-cluster \
        --gke-uri=$URI \
        --service-account-key-file=./anthos-hub-svc.json

# List Membership
gcloud container hub memberships list

# Register Minikube with Anthos
gcloud container hub memberships register local-cluster \
      --service-account-key-file=./anthos-hub-svc.json \
      --kubeconfig=~/.kube/config \
      --context=minikube

# List Membership
gcloud container hub memberships list
```

```
# Create Kubernetes Role

kubectl config use-context minikube

cat <<EOF > cloud-console-reader.yaml
kind: ClusterRole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: cloud-console-reader
rules:
- apiGroups: [""]
  resources: ["nodes", "persistentvolumes"]
  verbs: ["get", "list", "watch"]
- apiGroups: ["storage.k8s.io"]
  resources: ["storageclasses"]
  verbs: ["get", "list", "watch"]
EOF
kubectl apply -f cloud-console-reader.yaml

# Create RoleBinding
kubectl create serviceaccount local-cluster

kubectl create clusterrolebinding local-cluster-anthos-view \
     --clusterrole view \
     --serviceaccount default:local-cluster

kubectl create clusterrolebinding cloud-console-reader-binding \
     --clusterrole cloud-console-reader \
     --serviceaccount default:local-cluster
```

Google Cloud Platform Fundamentals

```
# Get the Token
SECRET_NAME=$(kubectl get serviceaccount local-cluster -o jsonpath='{$.secrets[0].name}')

# Copy the secret and paste it in the console
kubectl get secret ${SECRET_NAME} -o jsonpath='{$.data.token}' | base64 --decode

# Delete Membership
gcloud container hub memberships delete cloud-cluster
gcloud container hub memberships delete local-cluster

# Clean up
gcloud container clusters delete cloud-cluster --project=${PROJECT_ID}
gcloud iam service-accounts delete anthos-hub@${PROJECT_ID}.iam.gserviceaccount.com
minikube delete
```

Google Cloud Migration Tools

Migration

Transfer Appliance

Migrate for Compute Engine

BigQuery Data Transfer Service

Google Cloud Products & Services

https://cloud.google.com/products

Google Cloud Platform Fundamentals

Resources for Google Cloud Enterprise Services

Key Links

- Google Cloud IoT Core
- Edge TPU
- Apigee API Platform
- Anthos
- Stackdriver

References

- Anthos: Making Hybrid Cloud a Reality
- Everything You Want To Know About Anthos
- How to build an end-to-end IoT Solution on GCP