

Demystifying Application Connectivity with Kubernetes in the Docker Platform

Session #154945



Karthik Prabhakar

Dir. Solution Architecture, Tigera
@worldhopper



Nicola Kabar

Solution Architect, Docker
@nicolakabar

Quick Poll #1

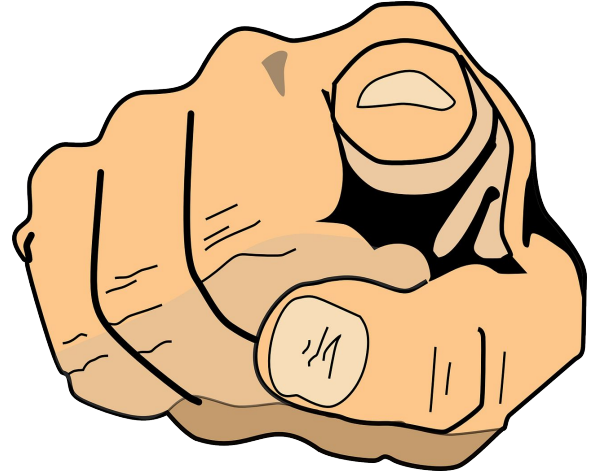
Who is designing a Kubernetes Container Platform ?

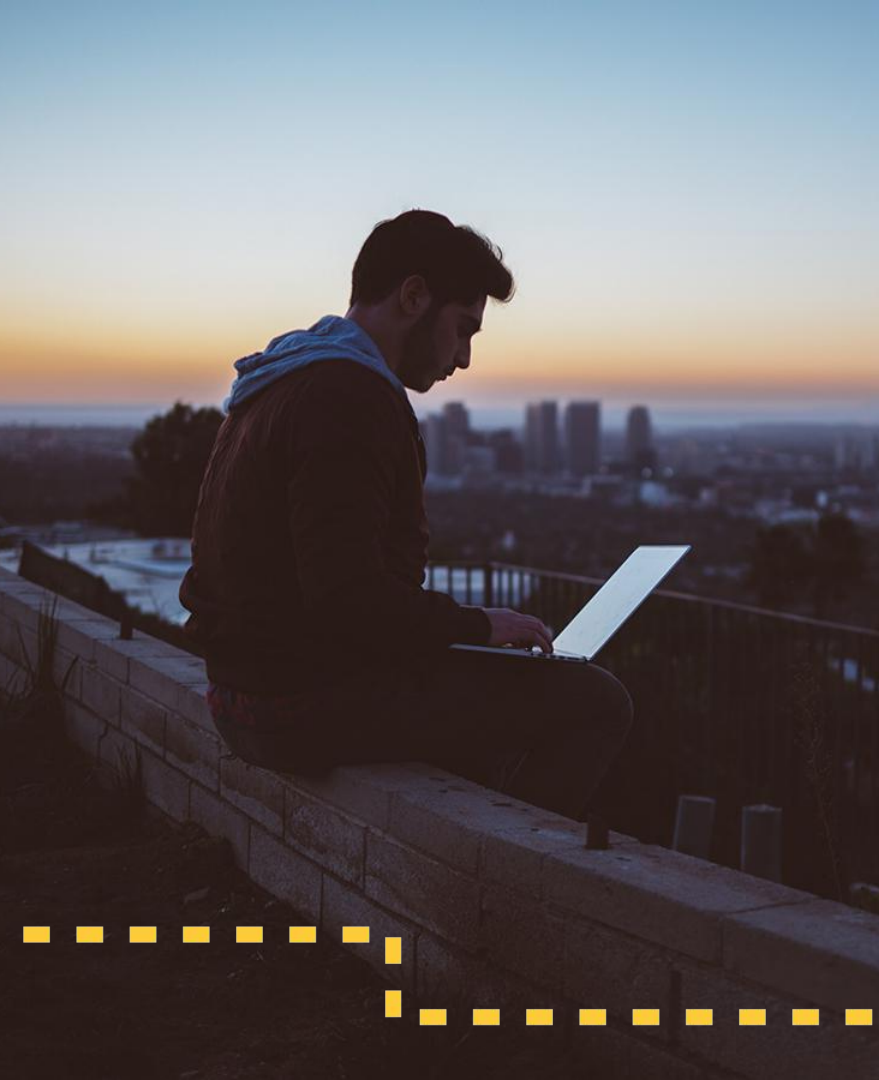
Quick Poll #2

**Who is intimidated by Kubernetes
Networking?**

...Then This Session is for You!

- **Networking Design Considerations**
- **Calico CNI + Basics of K8S Networking**
- **Docker EE + Calico Integration Overview**
- **Application Deployment with Docker EE**
- **Takeaways**
- **Q&A**





Design Considerations

- **Heterogeneous Infrastructure**
- **Diverse Application Portfolio**
- **Zero-Trust Security**

Let's Start with K8S Connectivity Basics

Kubernetes Connectivity Concepts

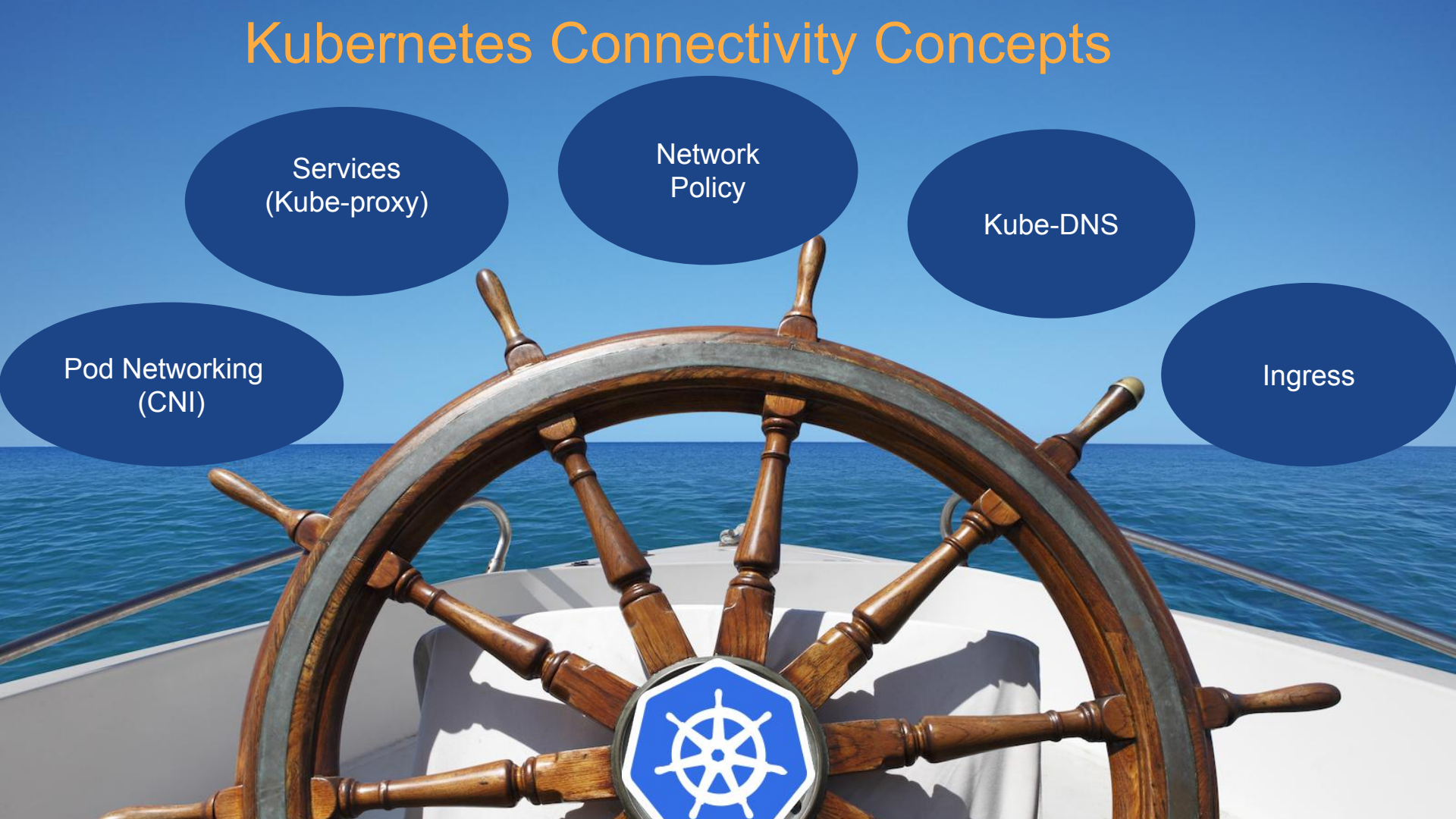
Services
(Kube-proxy)

Network
Policy

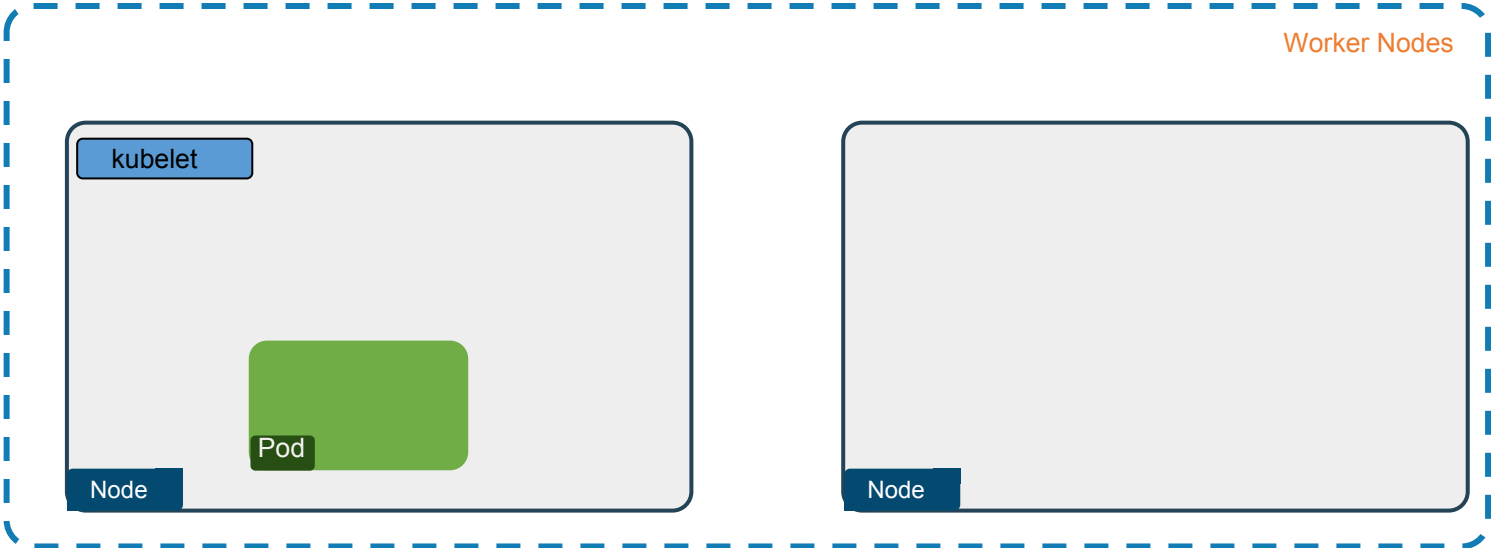
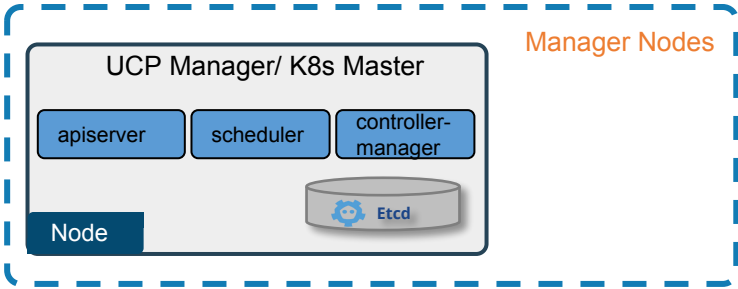
Kube-DNS

Ingress

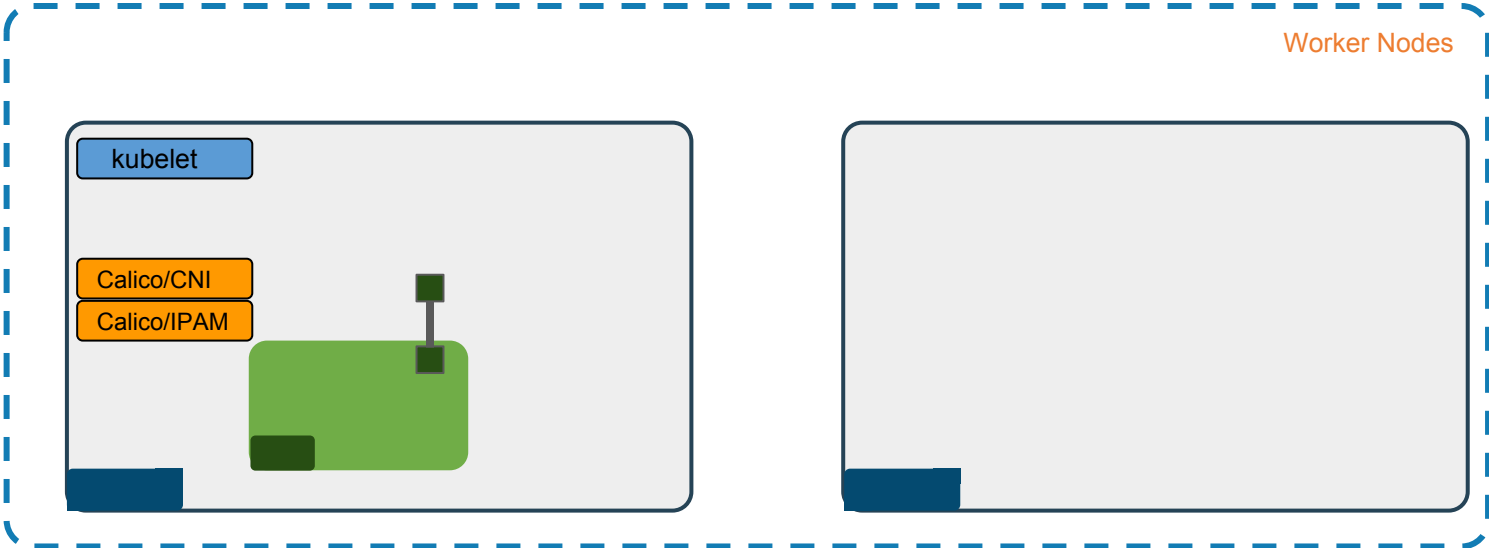
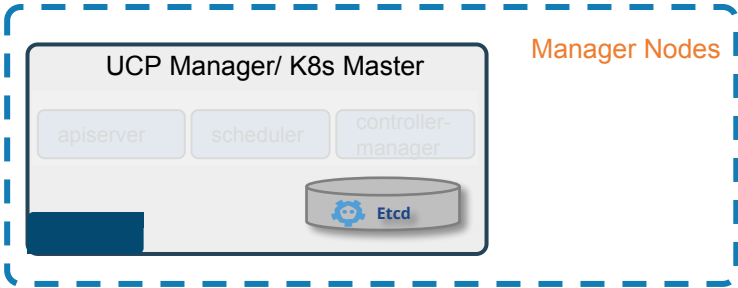
Pod Networking
(CNI)



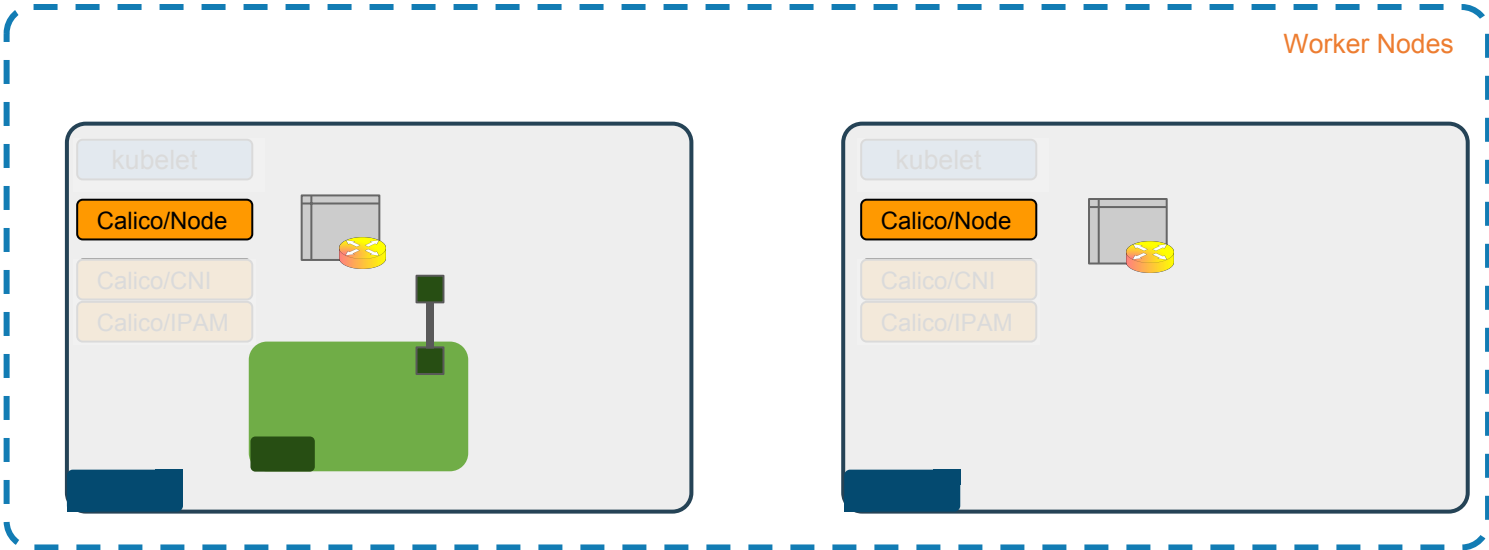
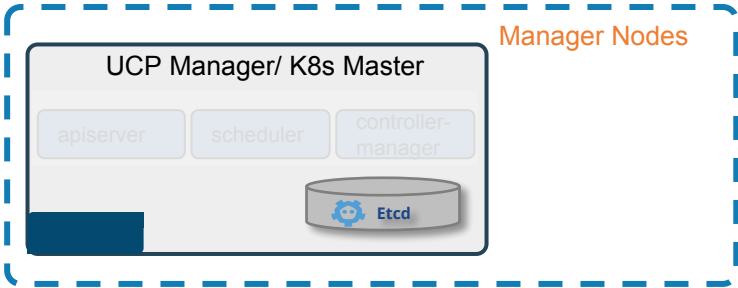
Pod Creation: Kubelet



Pod Network: Calico/CNI



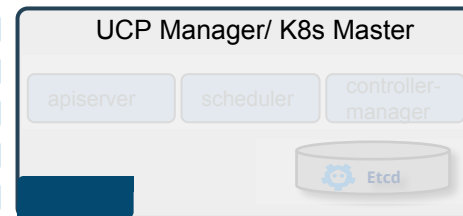
Pod Network: Calico/Node



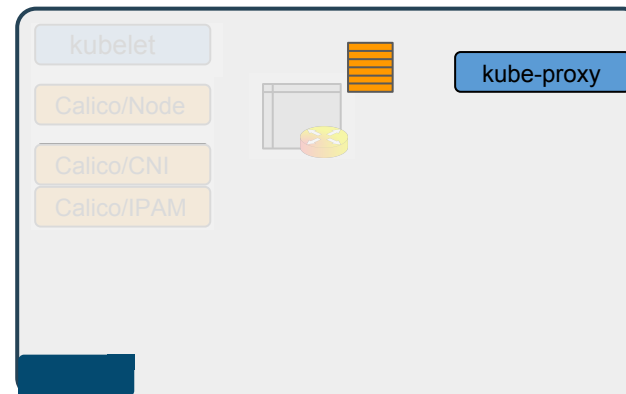
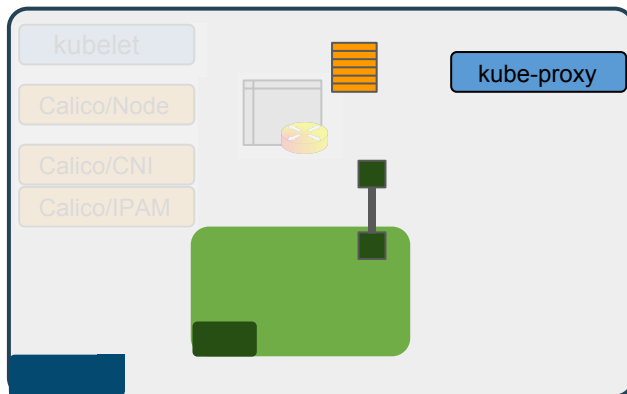
Kubernetes Services: Kube-proxy

- > Cluster IP
- > Node Port
- > Load Balancer

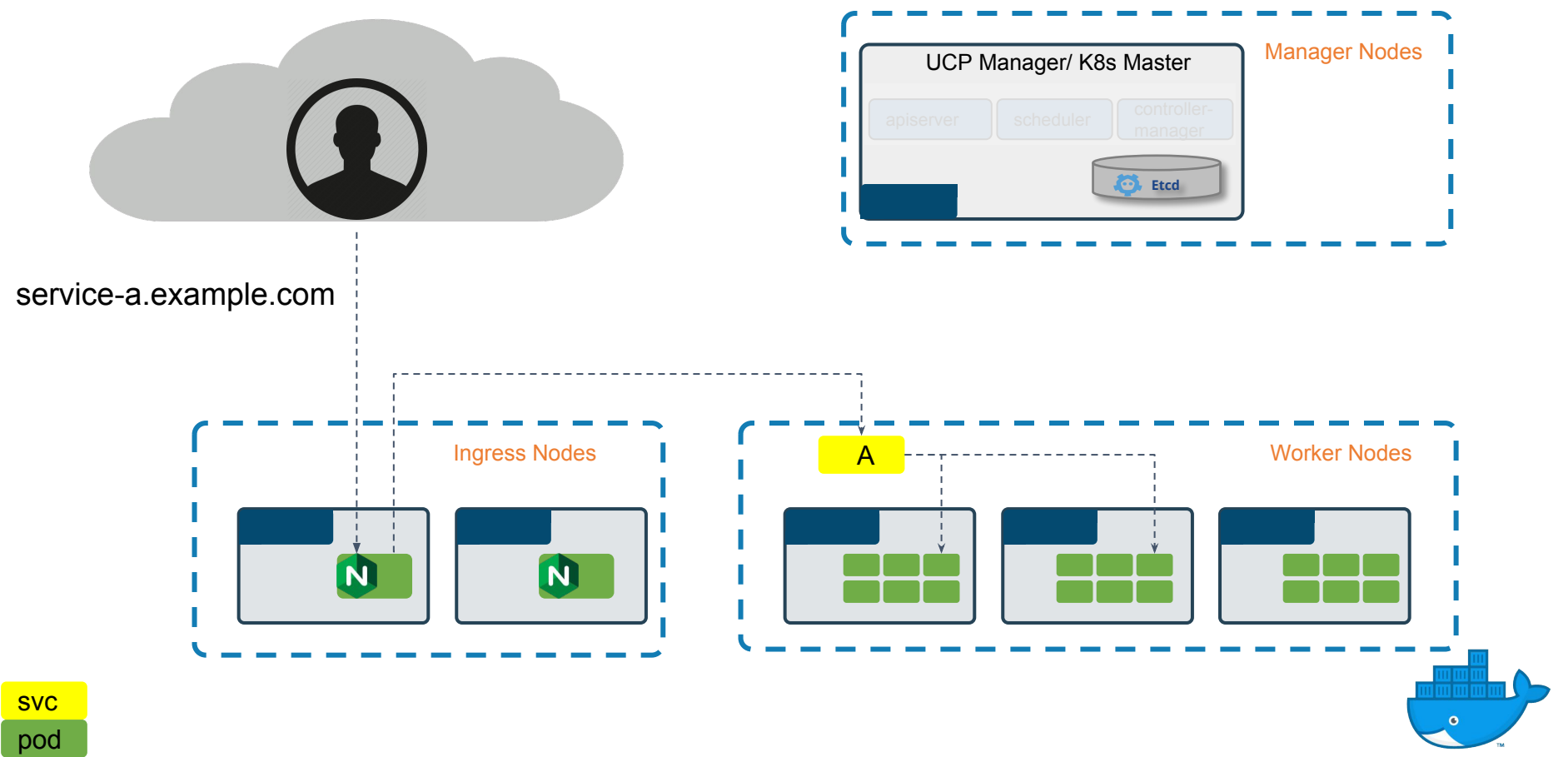
Manager Nodes



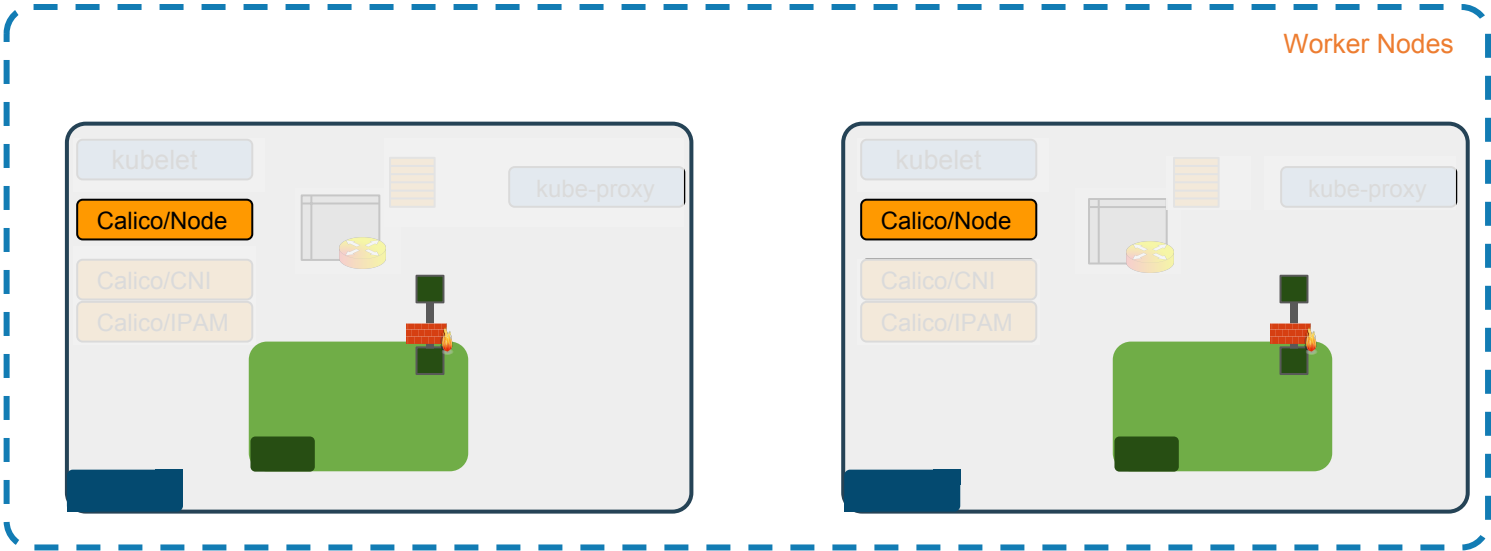
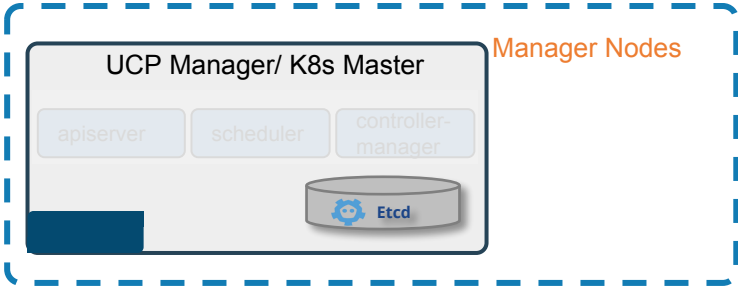
Worker Nodes



Kubernetes Services: Ingress



Network Policy



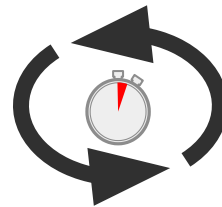
Kubernetes/Calico Network Policy



Label-based



Declarative



Dynamic

TIGERA CALICO: WHY IT'S AWESOME



Secure networking for the cloud-native era

Open source, maintained by Tigera with hundreds of third party contributors

Batteries-included Container networking for Docker EE
Kubernetes

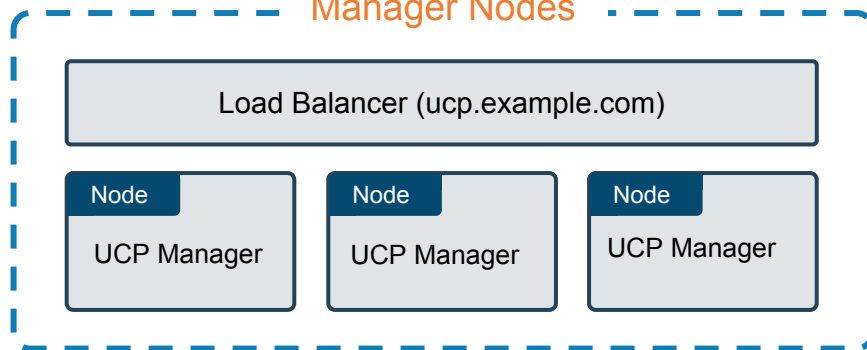
- > **Scalable, distributed control plane**
- > **Policy-driven network security**
- > **No overlay required**
- > **Integrated with all major cloud platforms**
- > **Widely deployed, proven at scale**



Docker Enterprise Edition 2.0 + Calico Architecture Overview

Docker Enterprise Edition Overview

Manager Nodes



Add-ons



Logging



Monitoring



LDAP/AD

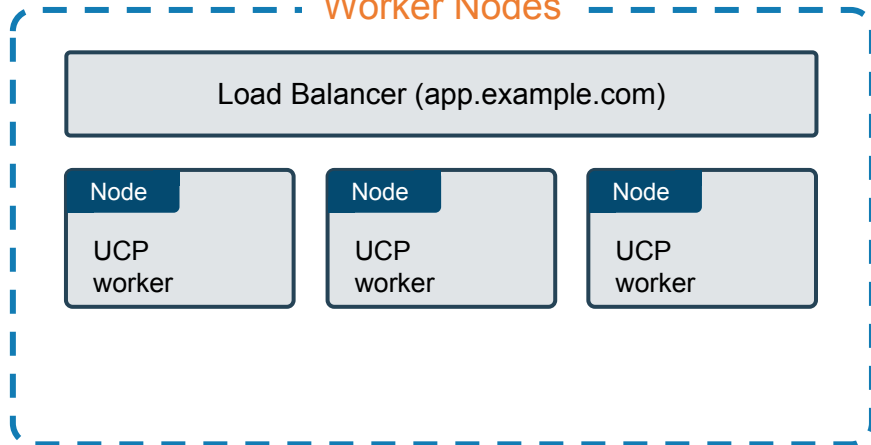


External CA

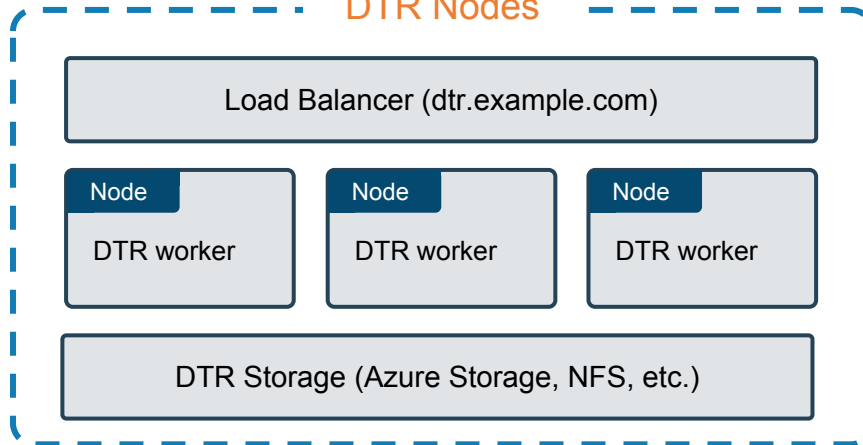


Image storage

Worker Nodes



DTR Nodes



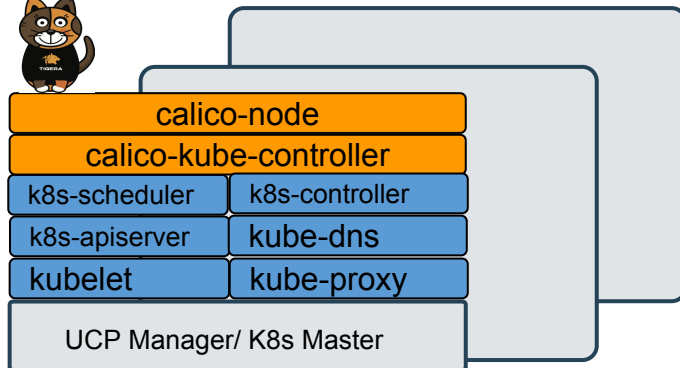
Docker Enterprise Edition 2.0 Calico Integration

certified

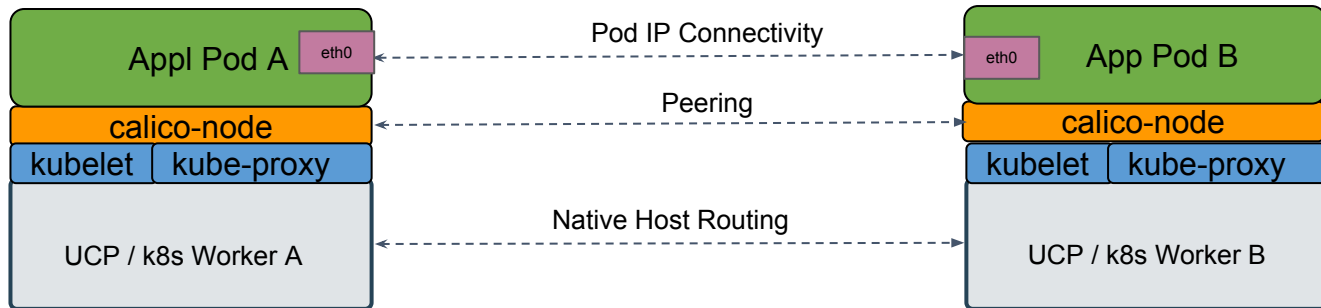


kubernetes

Manager Nodes



Worker Nodes



Demo 1: Docker EE + Calico Overview



Manager Nodes



Load Balancer (ucp.example.com)

Calico Node

UCP Manager/
K8s Master

Calico Node

UCP Manager/
K8s Master

Calico Node

UCP Manager/
K8s Master

Worker Nodes



App Pods

Calico Node

Node

UCP/K8S
worker

App Pods

Calico Node

Node

UCP/K8S
worker

App Pods

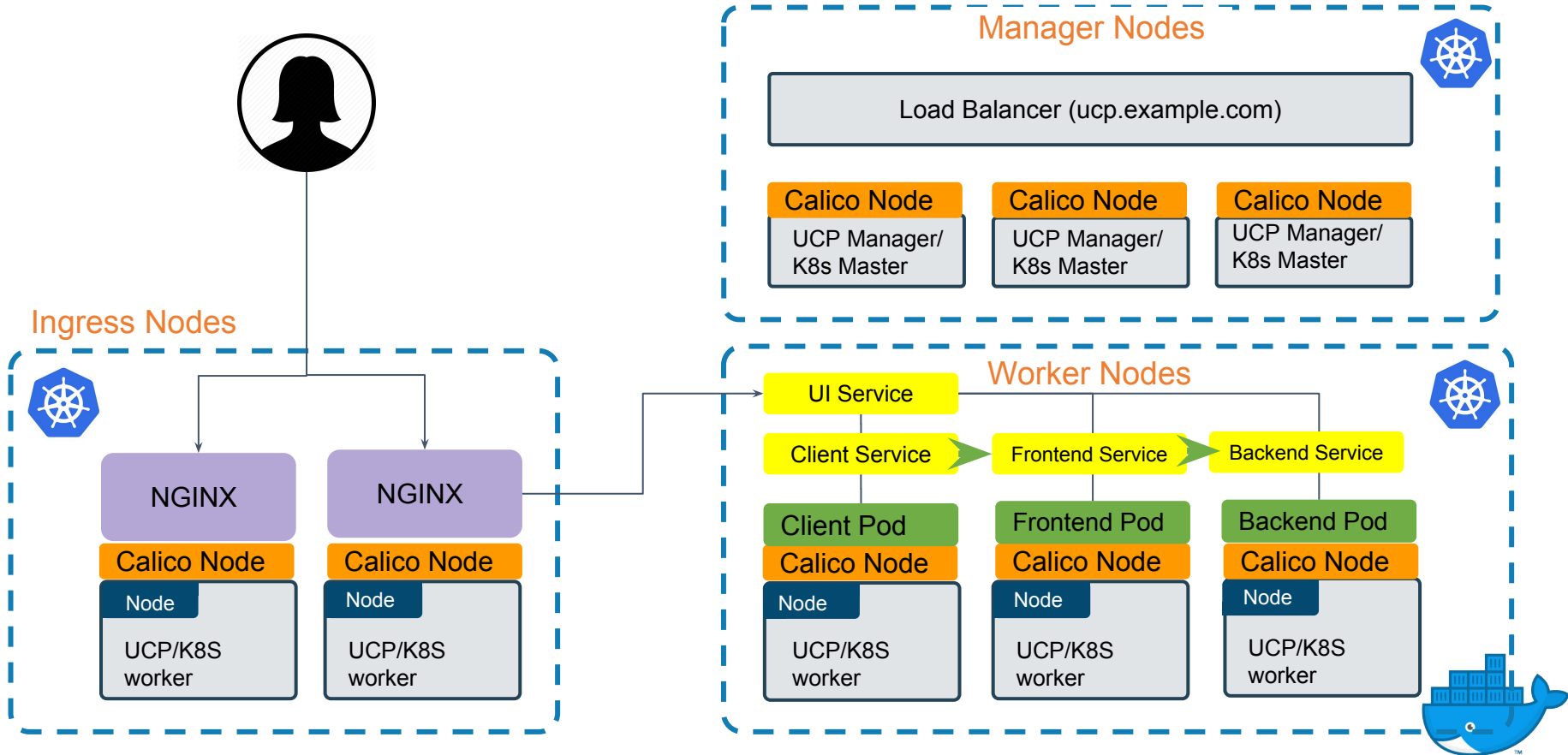
Calico Node

Node

UCP/K8S
worker



Demo 2: Let's Deploy a Sample Application



**Let's revisit our
Design Considerations.**

Heterogeneous Infrastructure

- **No Underlay Dependency = No Lock-In**
- **Simple Zero-Touch Provisioning**
- **Any Infrastructure, Any Cloud**
 - ◆ **On-Prem (VM, Bare)**
 - ◆ **Cloud (AWS, Azure, GCP)**
 - ◆ **Hybrid**



Google Cloud Platform



Diverse Application Portfolio

Connectivity Concept	Out-of-the-Box Solution with Docker EE 2.0
Pod - Pod	Calico CNI
Services	ClusterIP NodePort LoadBalancer
Ingress	NGINX Ingress Controller
DNS	kube-dns
K8s Network Policy	Calico



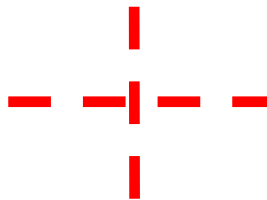
Zero-Trust Security



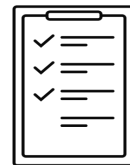
Stage/tier
separation



Tenant/namespace
isolation



Micro-
segmentation



Compliance



- **Declarative policy-driven isolation**
- **Fine-grained access control**
- **Dynamic, in lock step with Kubernetes**

Takeaway

Docker Enterprise Edition 2.0 seamlessly integrates Calico as the **default CNI providing **choice** of using native K8S connectivity abstractions on **any infrastructure** all while enforcing a dynamic policy-based microsegmentation.**

Thank you! Questions?

Try Docker EE + Calico: trial.docker.com

 community.docker.com

 slack.projectcalico.org

 @docker @projectcalico @tigeraio

