https://do.co/webinars

Janakiram + Associates

janakiram.com

Deploying & Managing Containerized Workloads in the Cloud

Janakiram MSV
Principal Analyst

Janakiram + Associates
janakiram.com

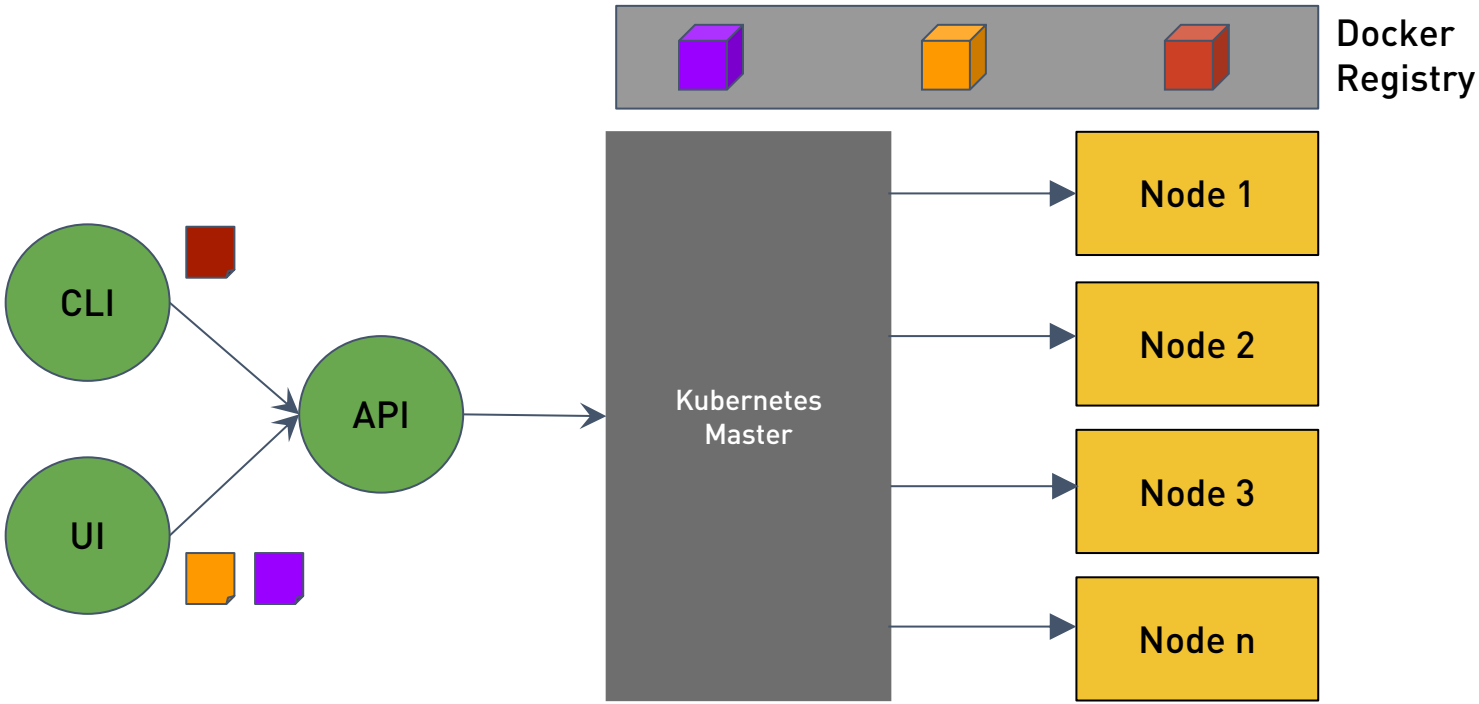
Session 4

A Closer Look at Kubernetes

Agenda

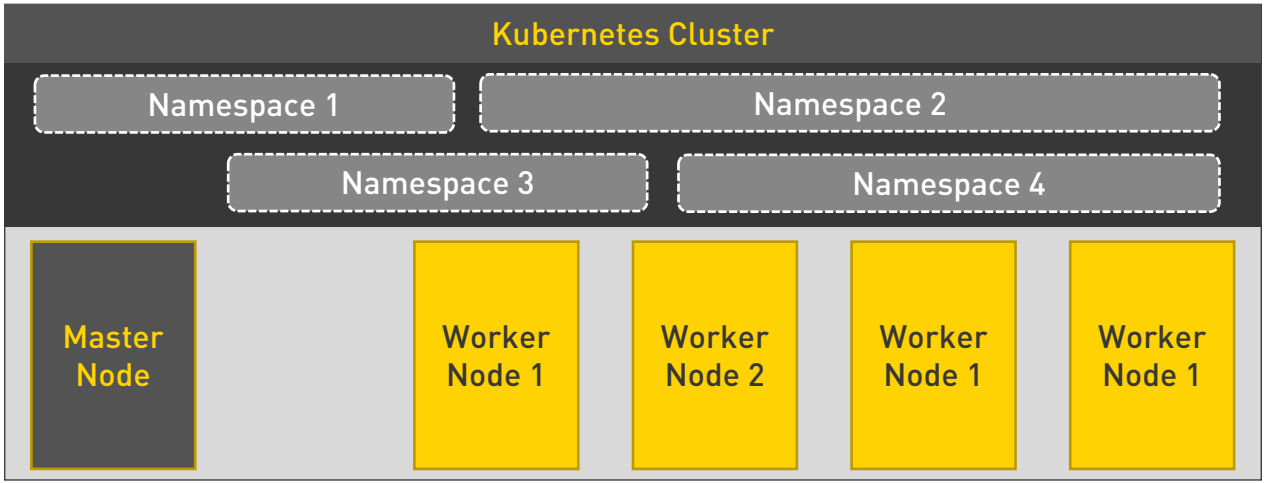
- Kubernetes Architecture
- Namespaces
- Nodes
- Pods
- Labels & Selectors
- Services
- Replica Sets
- Deployments

Kubernetes Architecture



Cluster, Nodes & Namespaces

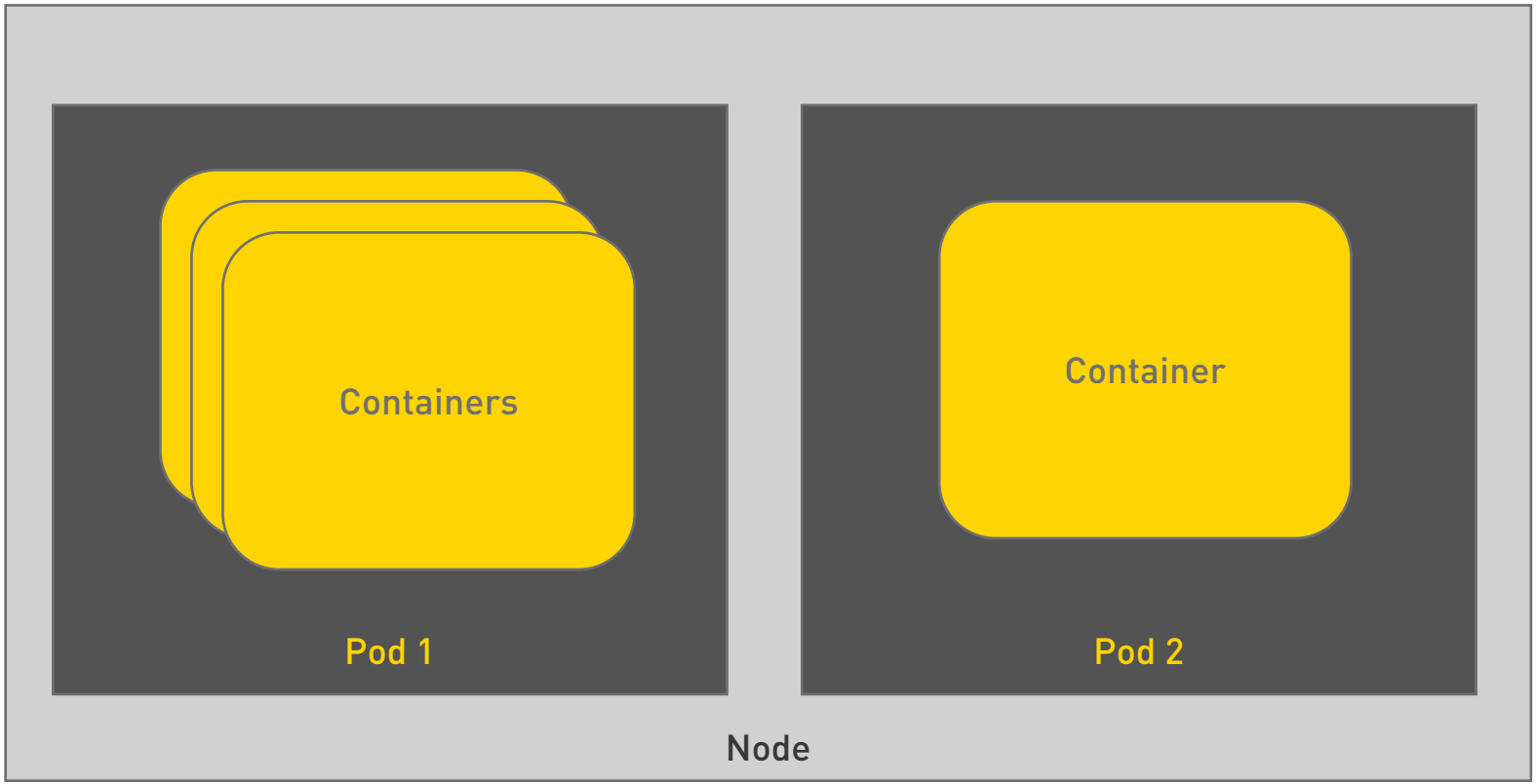
- Cluster represents a Kubernetes deployment
- Nodes are the physical or virtual machines that run the Cluster
- Namespaces are logical boundaries for applications deployed within the cluster



What are Pods?

- A group of containers that are always co-located and co-scheduled
- Containers within a Pod share the same context
  - IP address
  - Ports
  - Hostname
  - Storage
- Pods are scheduled in one or more Nodes
- Fundamental unit of deployment in Kubernetes

Kubernetes Pod



Demo

Exploring Nodes & Namespaces
Creating our First Pod

Labels & Selectors

- Every API object in Kubernetes may contain metadata in the form of key/value pairs
- Labels are the key/value pairs associated with Kubernetes objects
  - Node
  - Pod
  - Service
  - Replica Sets
  - ...
- Selectors are used for querying and matching a set of objects that meet specific criterion

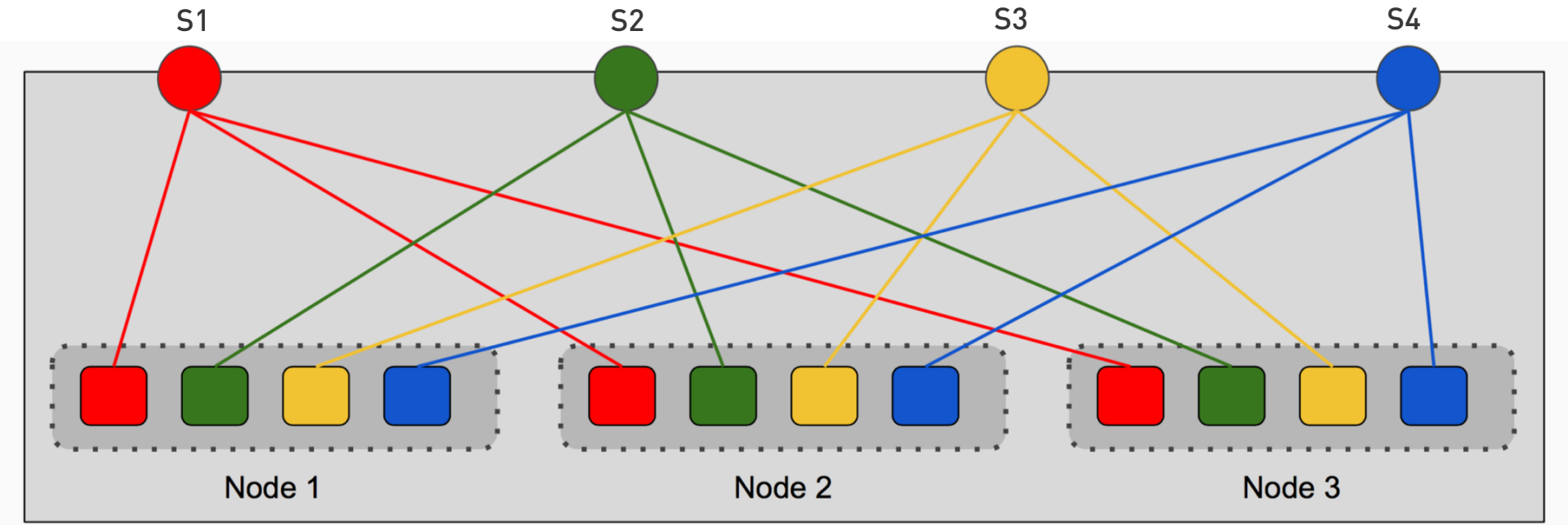
Kubernetes Service

- A service is an abstraction of a logical set of Pods defined by a policy
- It acts as an intermediary for Pods to talk to each other
- Selectors are used for accessing all the Pods that match a Label
- Service is an object in Kubernetes – Similar to Pods and Nodes
- Each Service exposes one or more *Ports* and *targetPorts*
- The *targetPort* is mapped to the port exposed by the matching Pods
- Services support TCP and UDP protocols

Demo

Exposing Pods through Services

Understanding Kubernetes Services



Replica Sets

- Ensures that a Pod or homogeneous set of Pods are always up and available
- Always maintains desired number of Pods
- If there are excess Pods, they get killed
- New pods are launched when they fail, get deleted, or terminated
- Creating a Replica Set with a count of 1 ensures that a Pod is always available
- Replica Sets and Pods are associated through Labels

Demo

Scaling Pods through Replica Set

Deployments

- Kubernetes Deployments are a combination of Pods and Replica Sets in one declaration unit
- Deployments are used to maintain desired number of Replica Sets
- Useful for performing CI/CD operations on applications
- Deployments provide PaaS-like capabilities to Kubernetes clusters

Demo

Exploring Deployments

Summary of Kubernetes Objects

- **Clusters**
  - Pool of compute, storage, and network resources
- **Nodes**
  - Host machines running within the cluster
- **Namespaces**
  - Logical partitions of a cluster
- **Pods**
  - Units of deployment
- **Labels**
  - Key-Value pairs for identification and service discovery
- **Services**
  - Collection of Pods belonging to the same application
- **Replica Set**
  - Ensures availability and scalability
- **Deployment**
  - Manages application lifecycle

Key Kubernetes Commands to Explore

- `kubectl create`
- `kubectl get`
- `kubectl describe`
- `kubectl scale`
- `kubectl delete`

Thank You!

Janakiram + Associates

janakiram.com