Pivotal | IT大咖说 知识共享平台

Pivotal

Greenplum on PKS

Containerized MPP Database

翁岩青
Pivotal Greenplum研发经理
iweng@pivotal.io

Agenda

- **Greenplum** Architecture
- **Greenplum** Data Platform
- Kubernetes on PCF
- **Greenplum** on Kubernetes
- Q+A

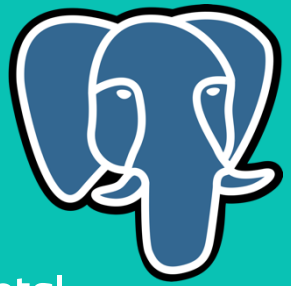
Pivotal Greenplum®

Pivotal Cloud Foundry®

Pivotal Greenplum®

WHAT IS GREENPLUM?

OPEN SOURCE
MASSIVELY PARALLEL PROCESSING
DATA WAREHOUSE



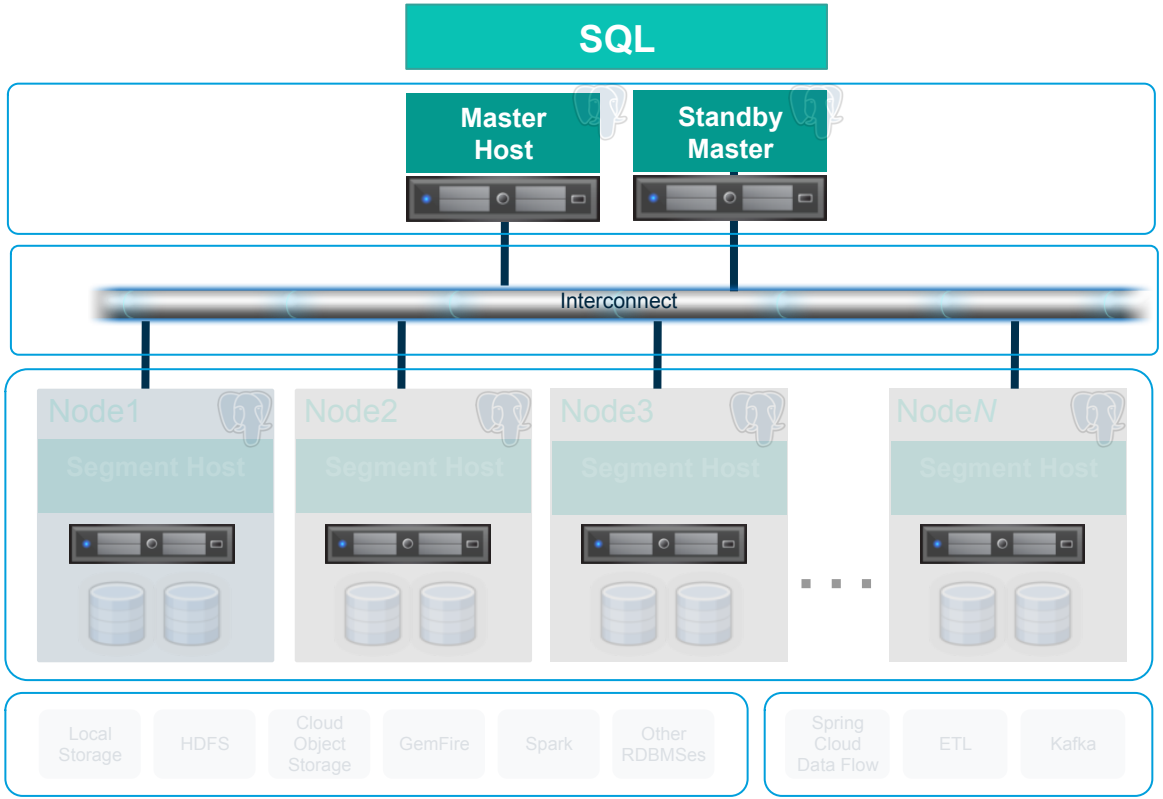
Greenplum = Massively Parallel Postgres for Analytics

Master Servers
Query planning and dispatch

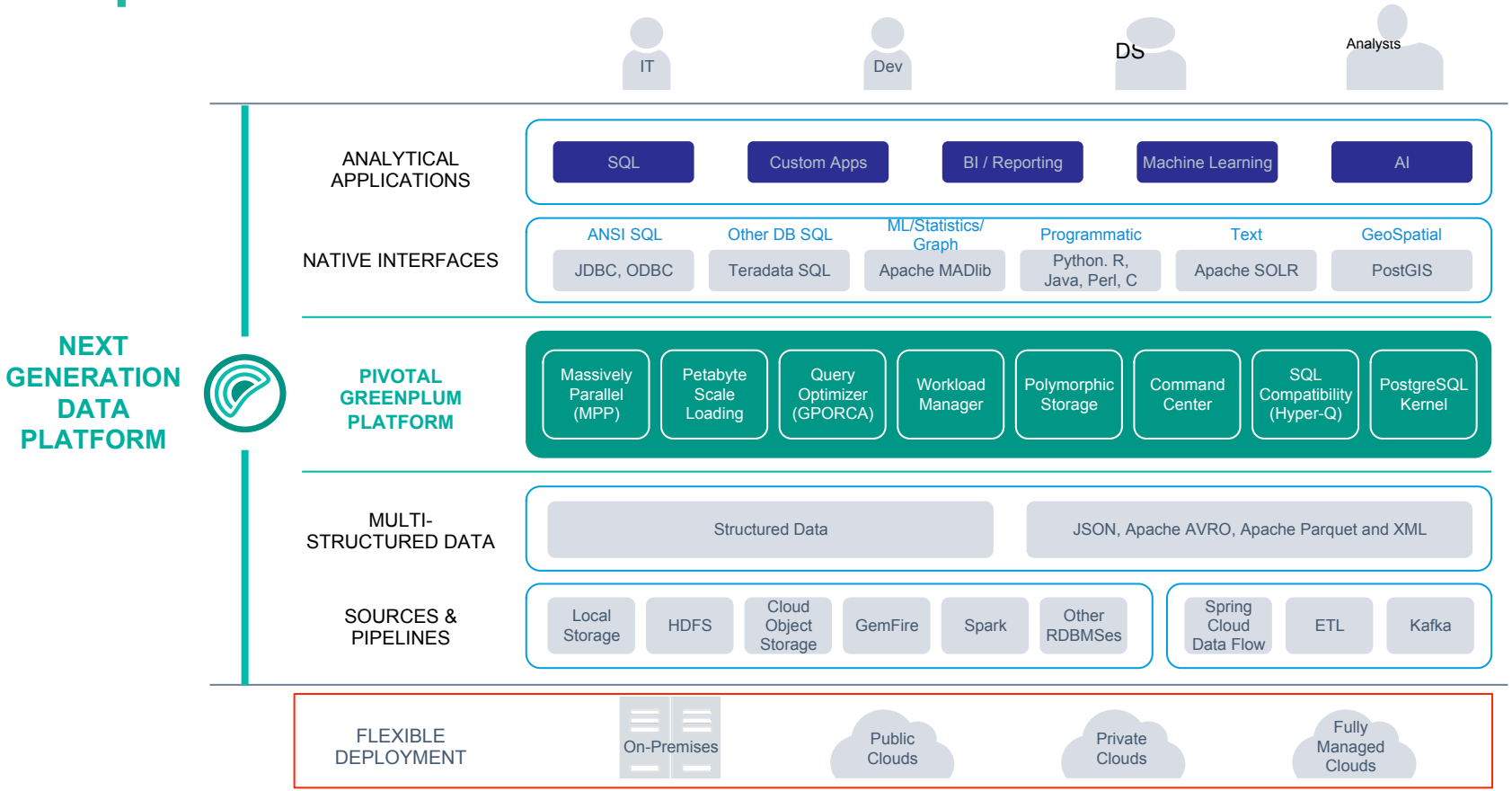
Interconnect

Segment Servers
Query processing and data storage

External Sources & Pipelines
Parallel loading and streaming



Greenplum Data Platform



Faster Deployments… How?

Infrastructure-Agnostic

Bare-Metal

Private Cloud

vmware vSphere

openstack.

Public Cloud

amazon web services

Microsoft Azure

Google Cloud Platform

Have you

Experienced these before with any database?

Have you...

- Ran out of disk space ?
- Been able to provision more than 100 postgres instances in few minutes?
- Faced issues in recovering failures ?
- Faced issues in expanding the database?

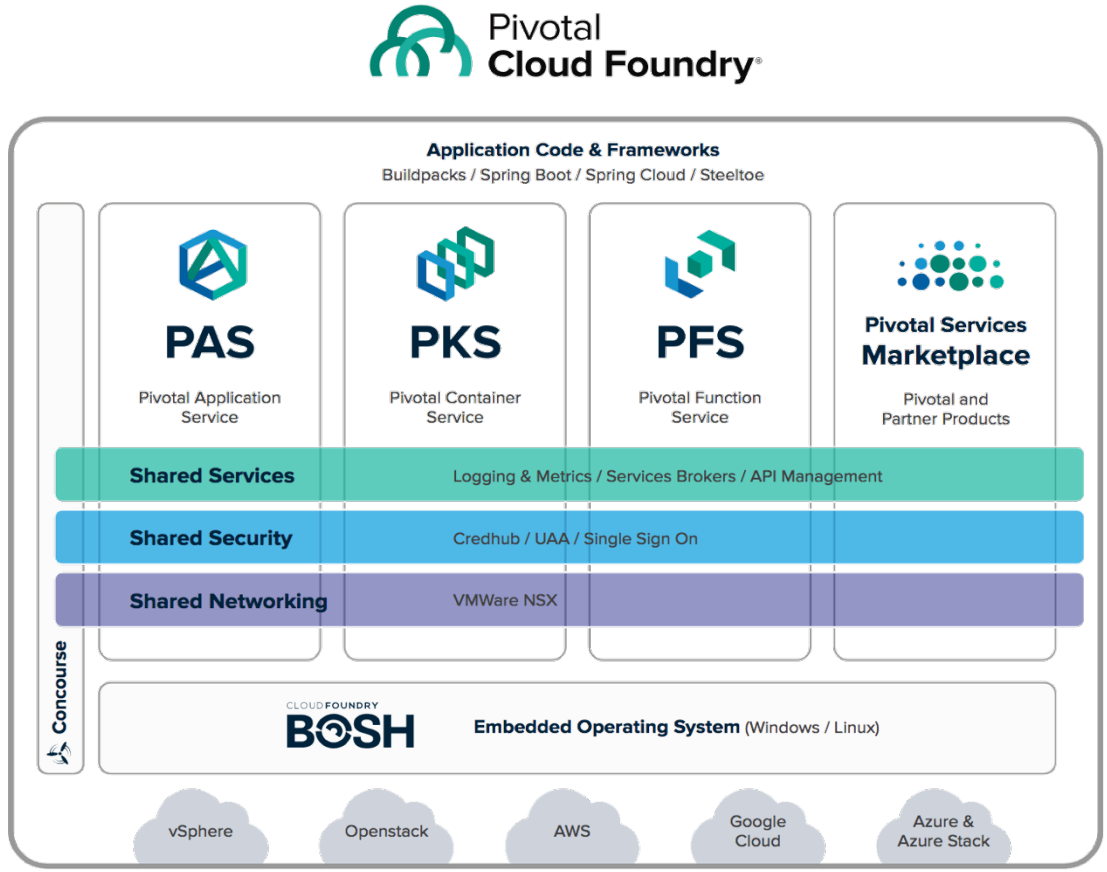
Pivotal
Container Service™

WHAT IS PKS?

RELIABLY DEPLOY AND RUN CONTAINERIZED WORKLOADS.

Kubernetes on Pivotal Cloud Foundry

Continuously deliver any app to every major private and public cloud with a single platform.



Faster Deployments… How?

Infrastructure-Agnostic

Bare-Metal

Private Cloud

vmware vSphere

openstack.

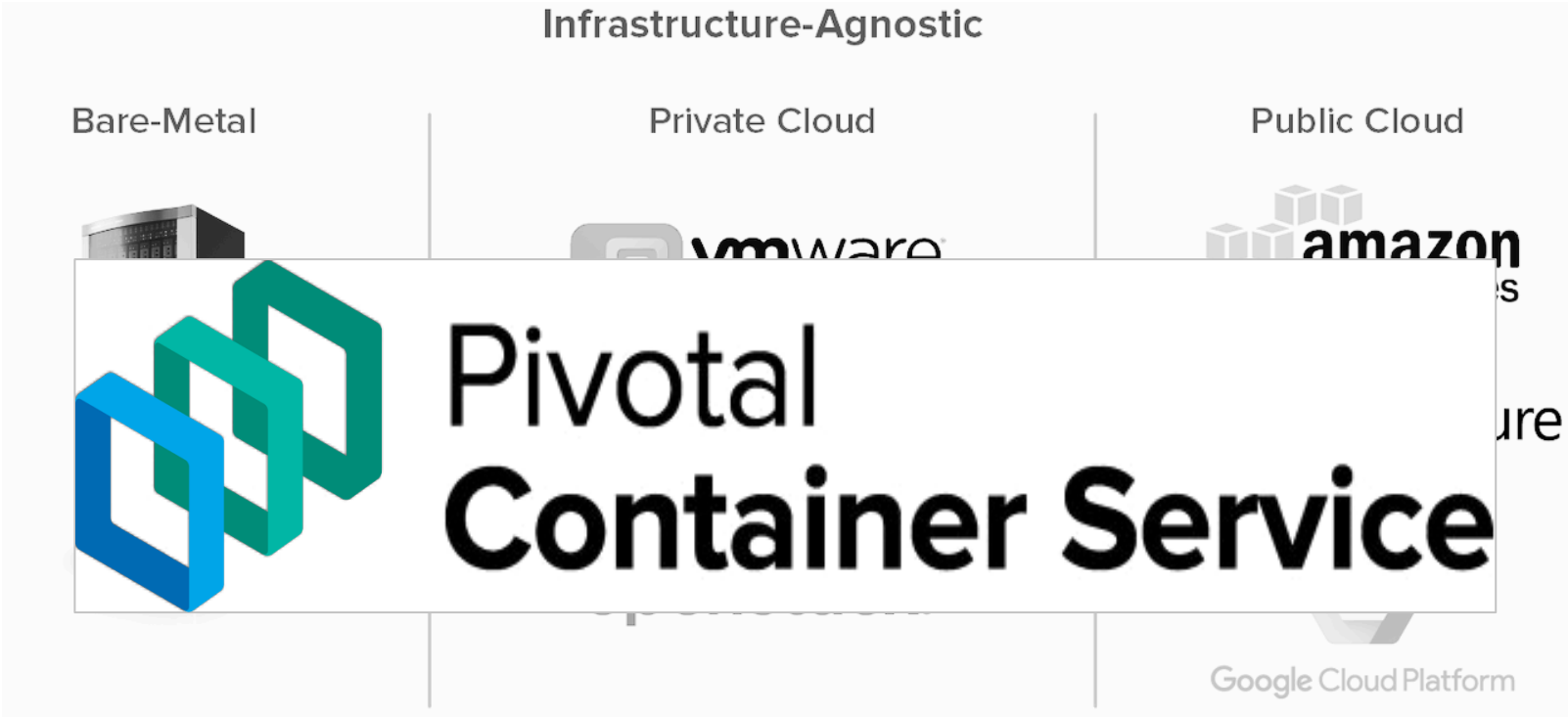
Public Cloud

amazon web services

Microsoft Azure

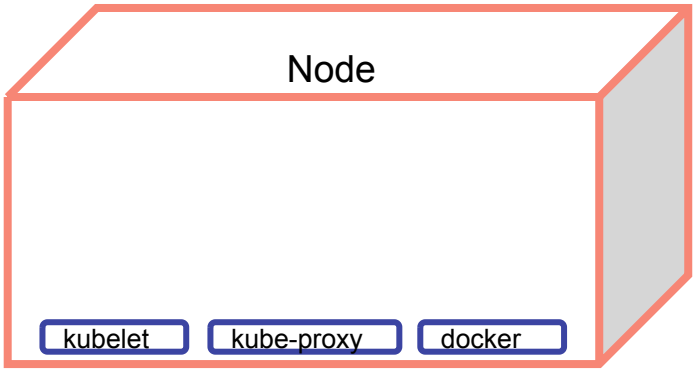
Google Cloud Platform

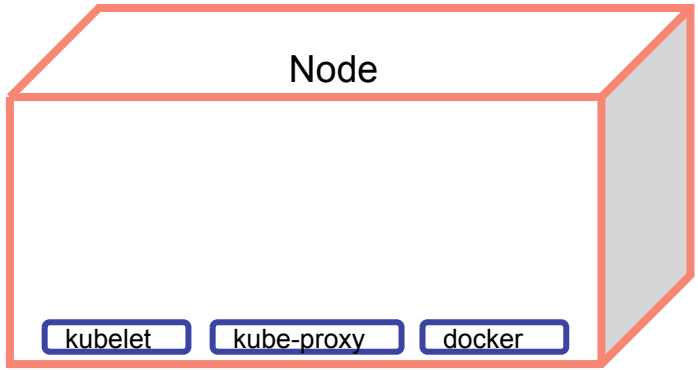
Greenplum Data Platform + PKS



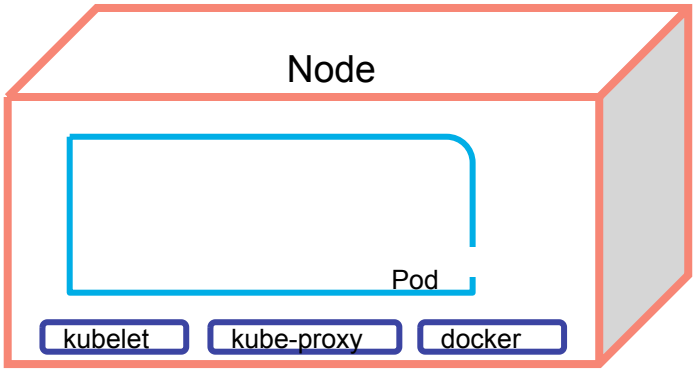
Kubernetes 101

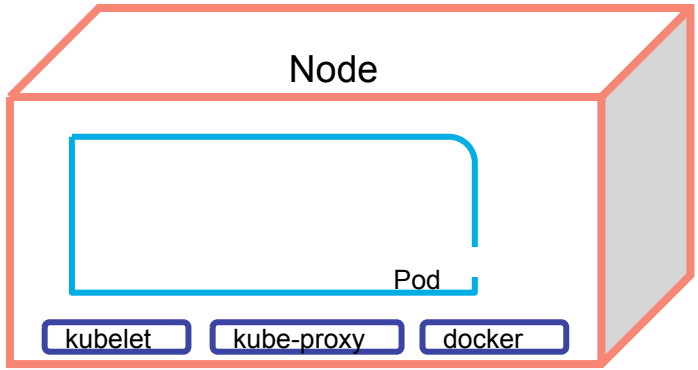
Kubernetes 101



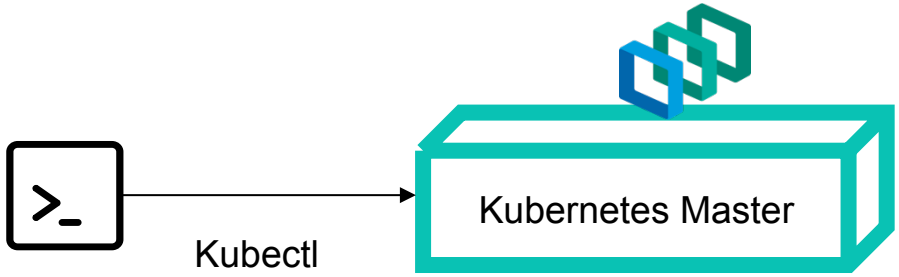


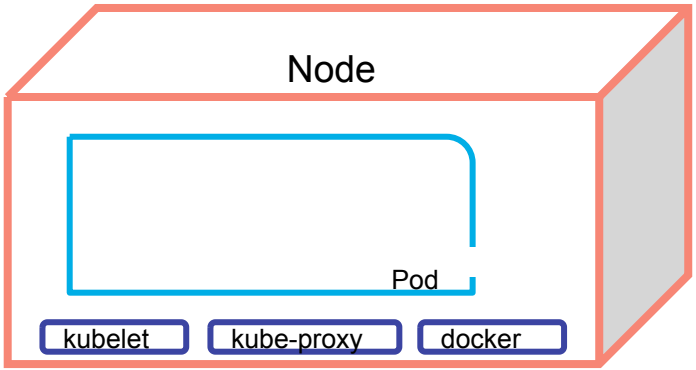
Kubernetes 101

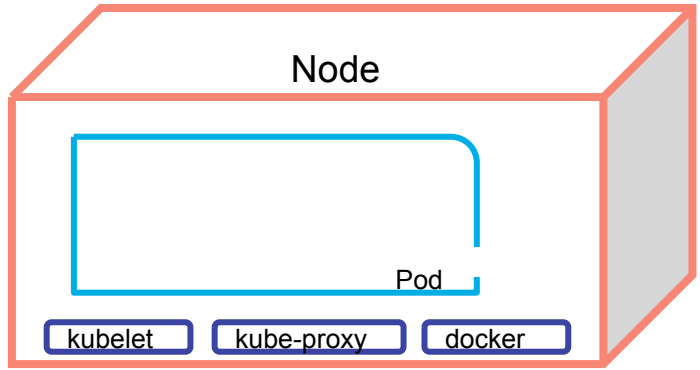




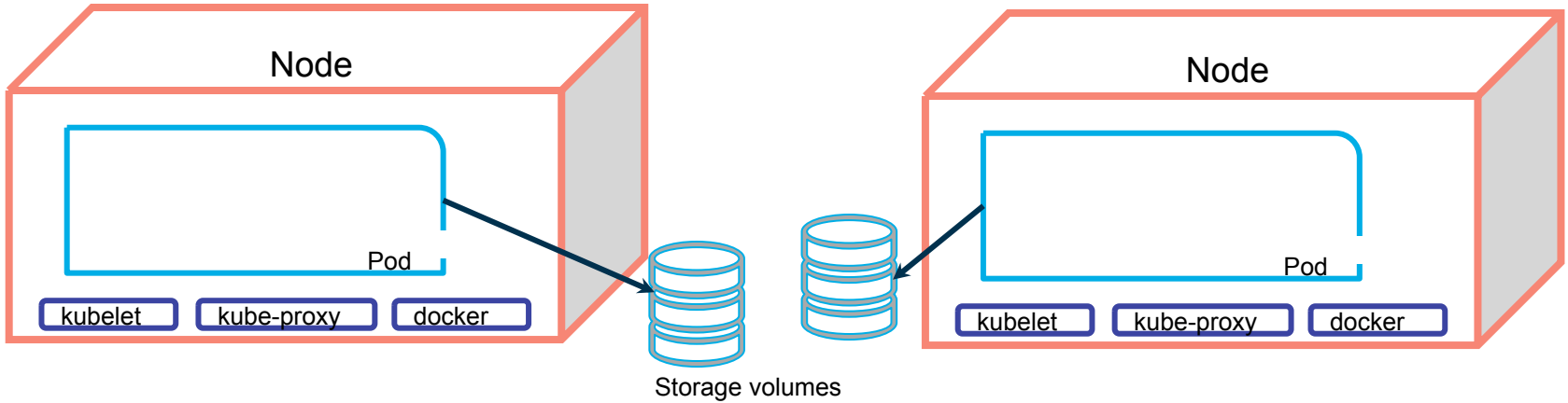
Kubernetes 101



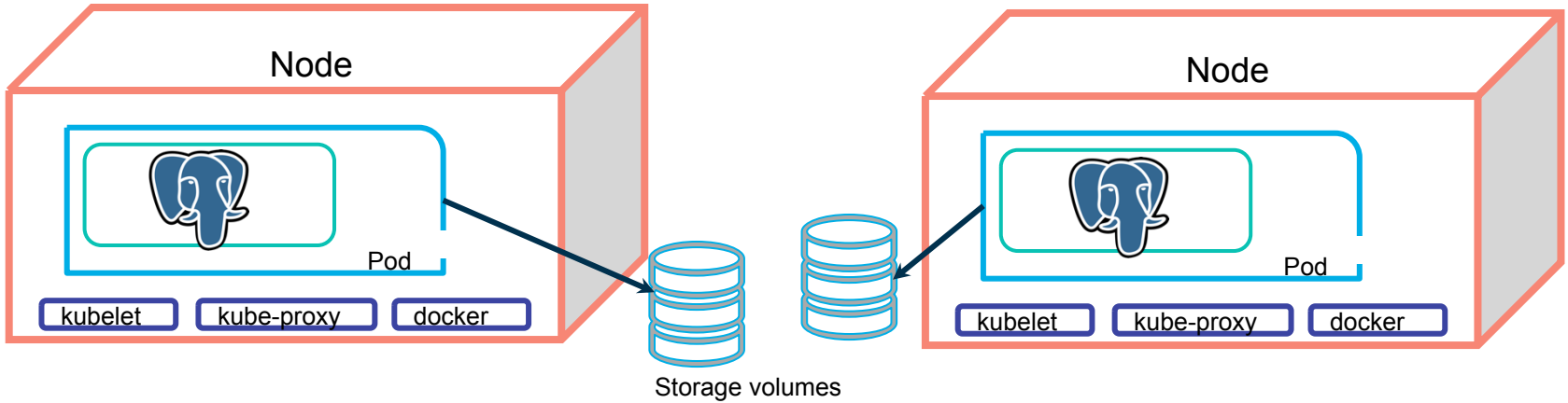




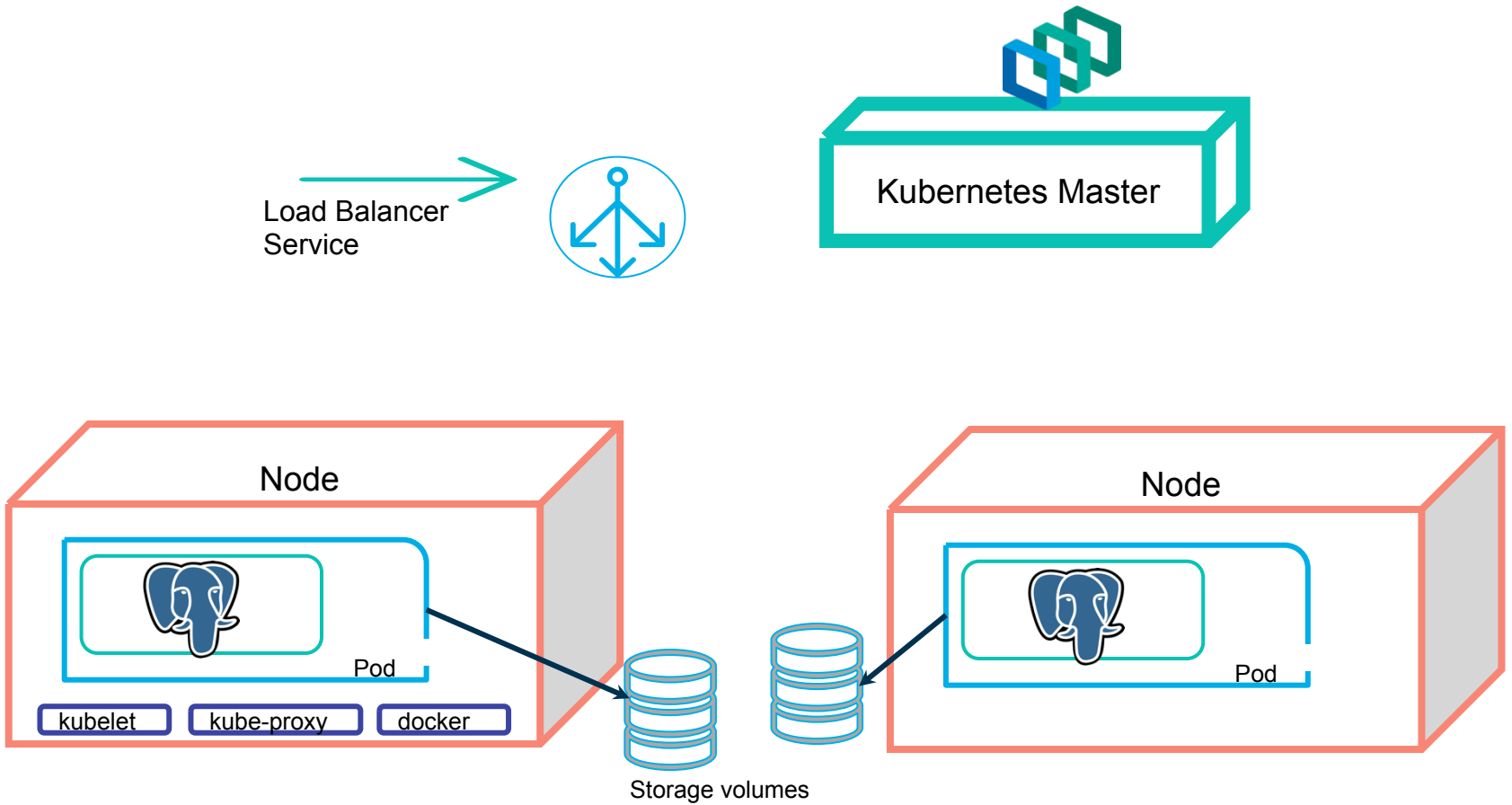
Kubernetes 101



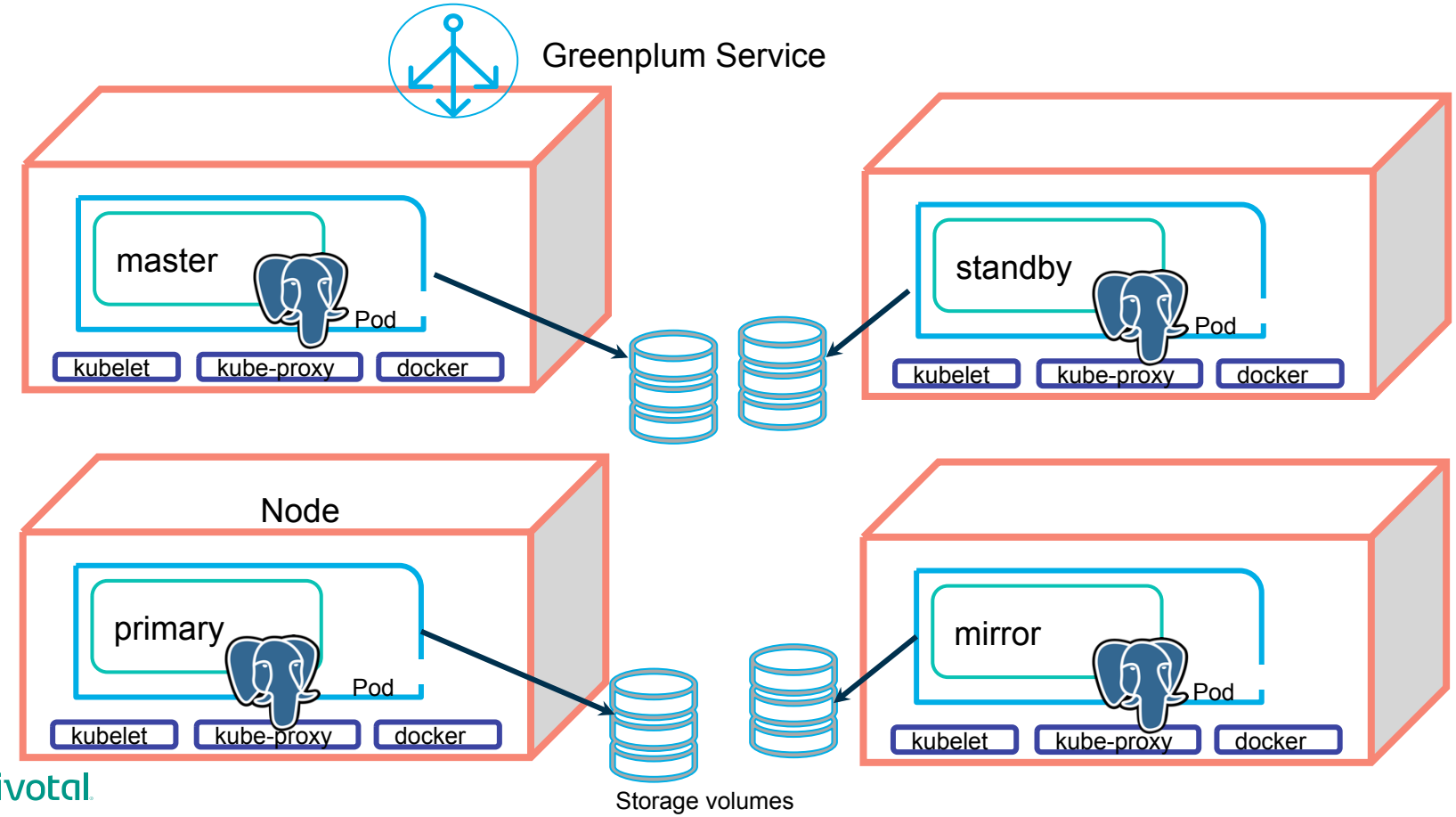
Kubernetes 101



Kubernetes 101



Greenplum on Kubernetes



Greenplum on PKS

Benefits

1. On Demand Cluster Provisioning

Radar

PKS

1. On Demand Cluster Provisioning

Radar

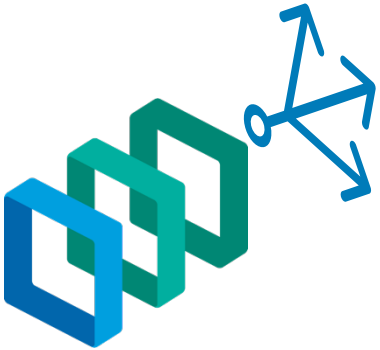
Give me a Greenplum Cluster

PKS

1. On Demand Cluster Provisioning

Radar

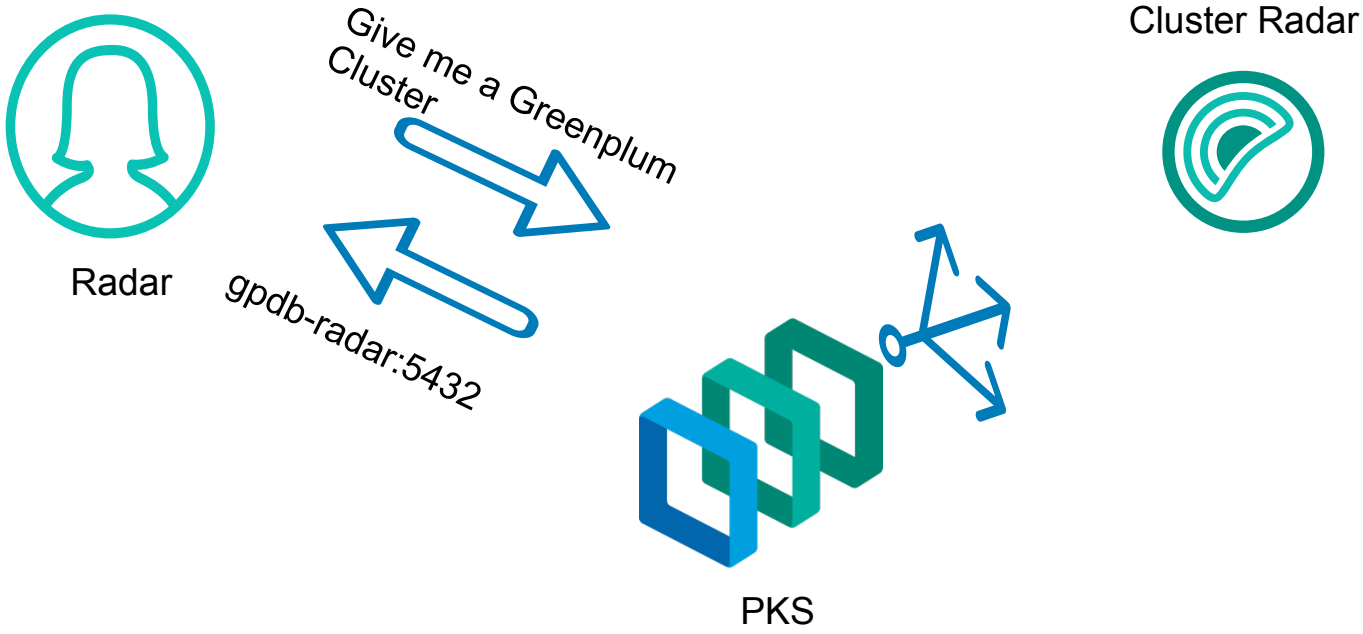
Give me a Greenplum Cluster



PKS

Cluster Radar

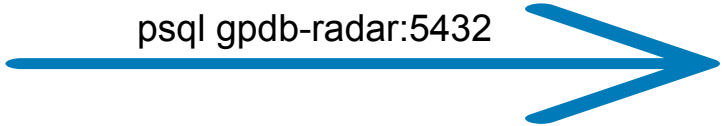
1. On Demand Cluster Provisioning



1. On Demand Cluster Provisioning

Radar

psql gpdb-radar:5432



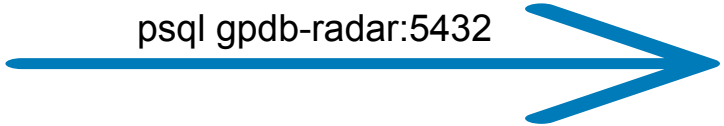
Cluster Radar

PKS

1. On Demand Cluster Provisioning

Radar

psql gpdb-radar:5432

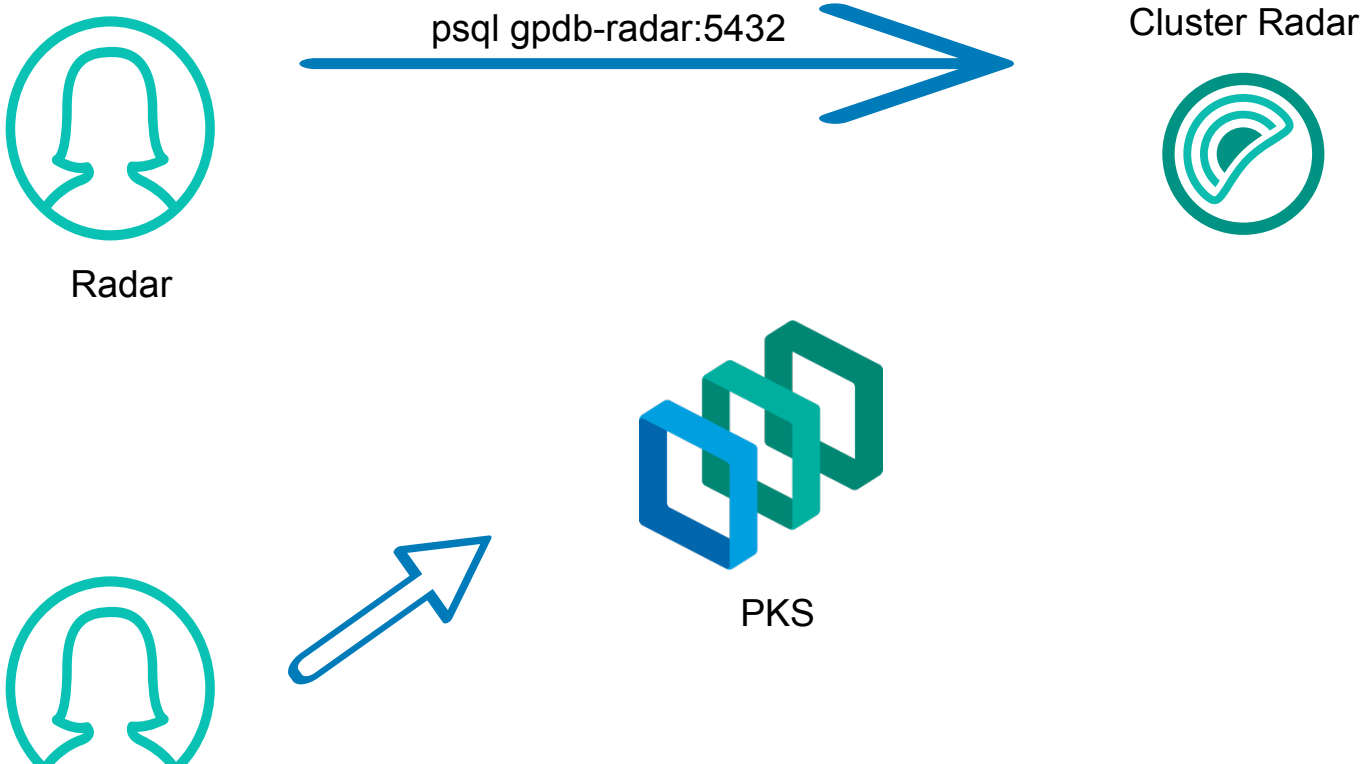


Cluster Radar

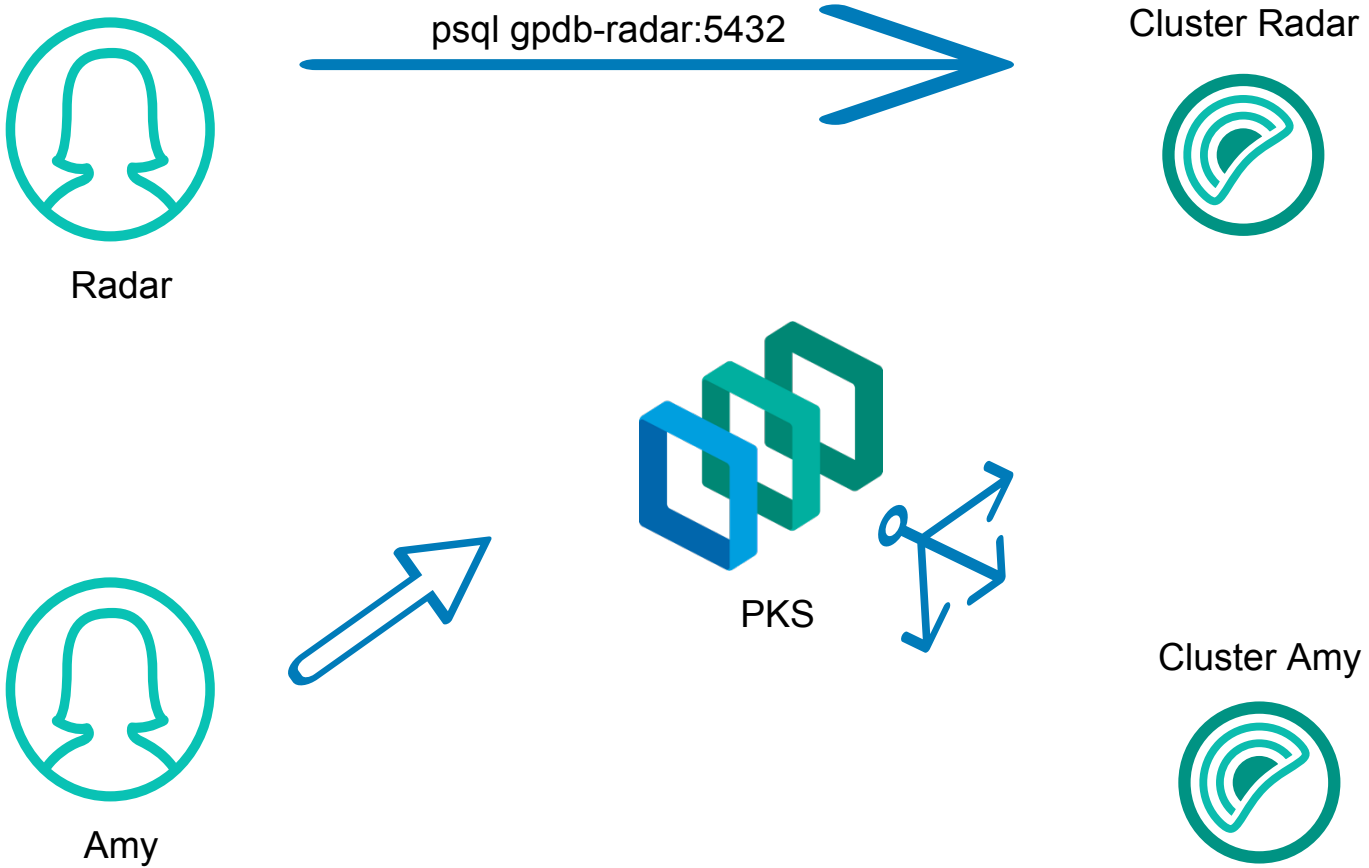
PKS

Amy

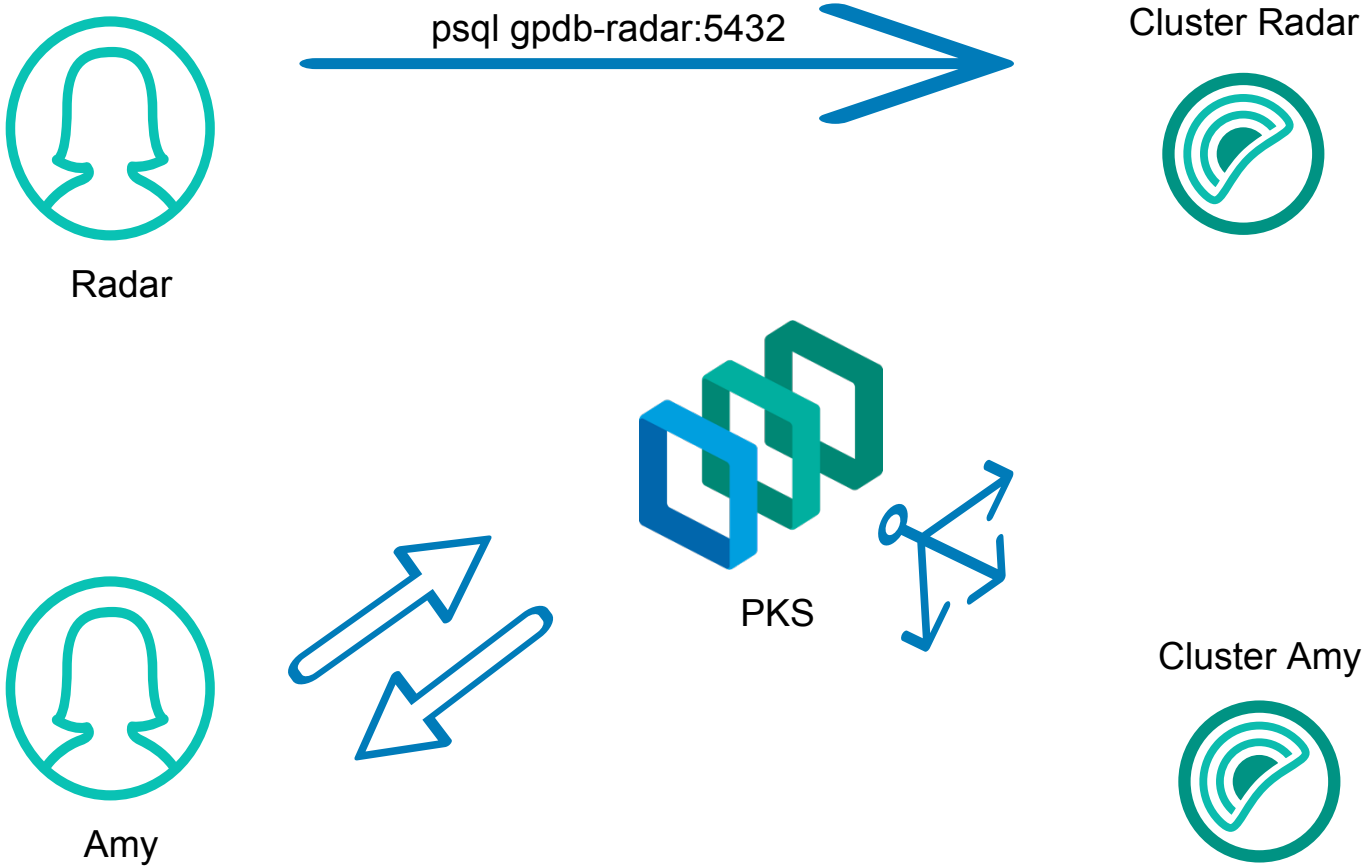
1. On Demand Cluster Provisioning



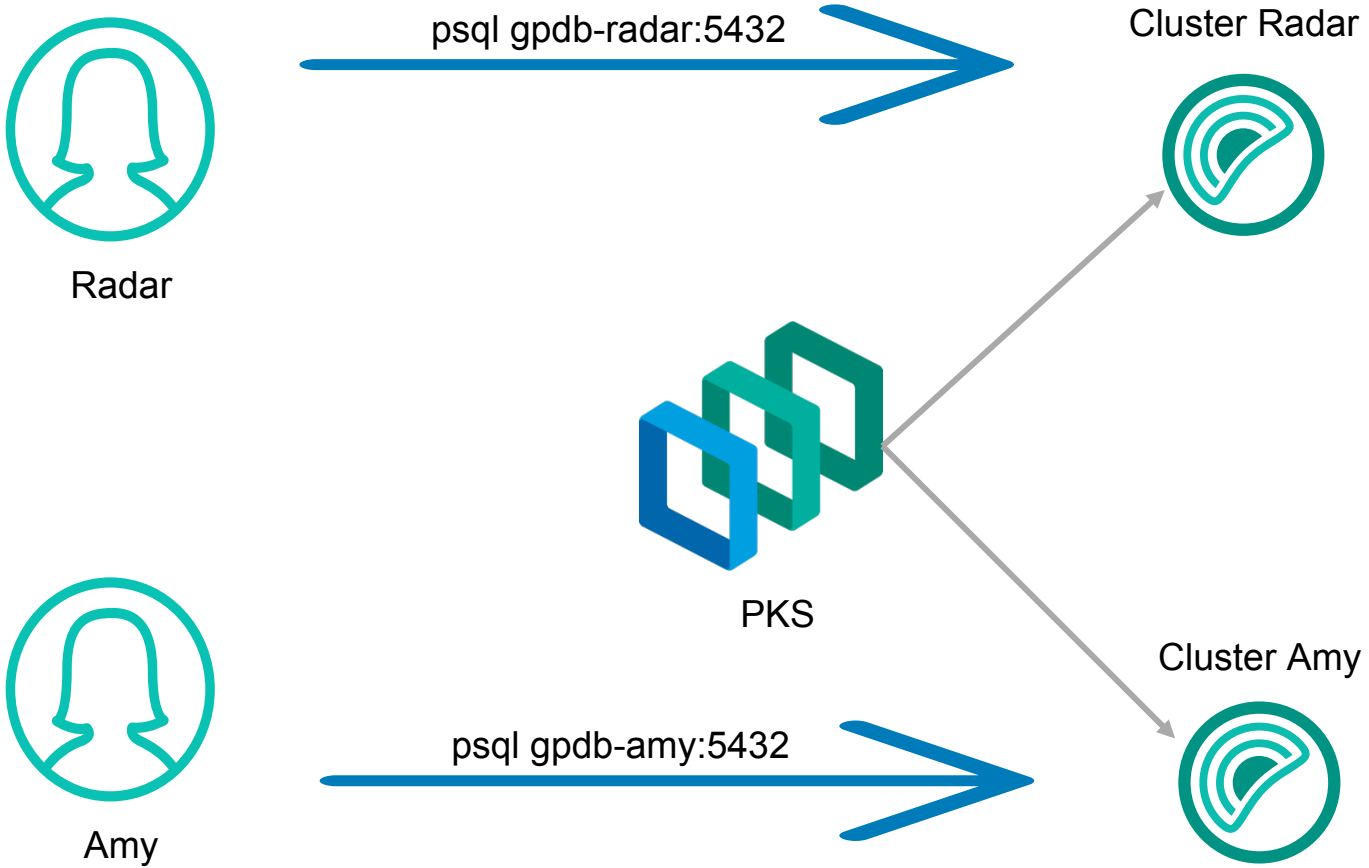
1. On Demand Cluster Provisioning



1. On Demand Cluster Provisioning



1. On Demand Cluster Provisioning

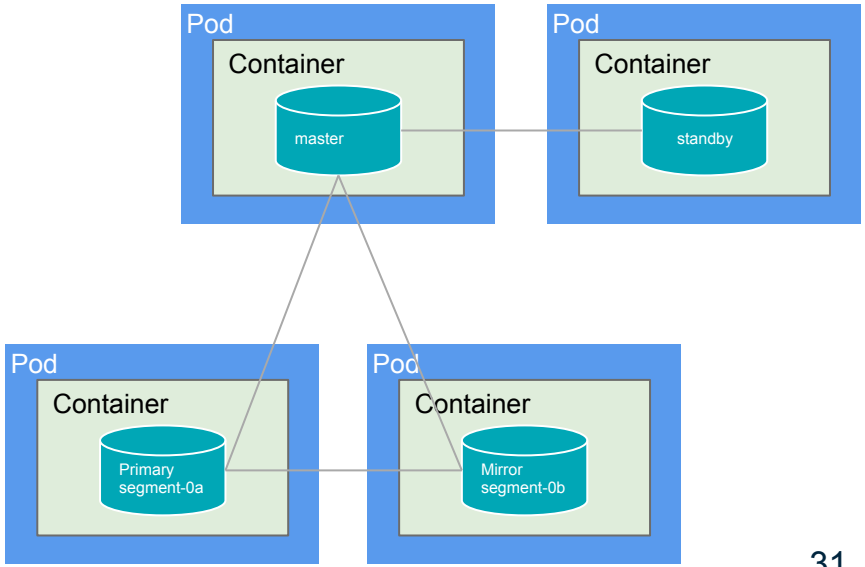


2. Service Discovery

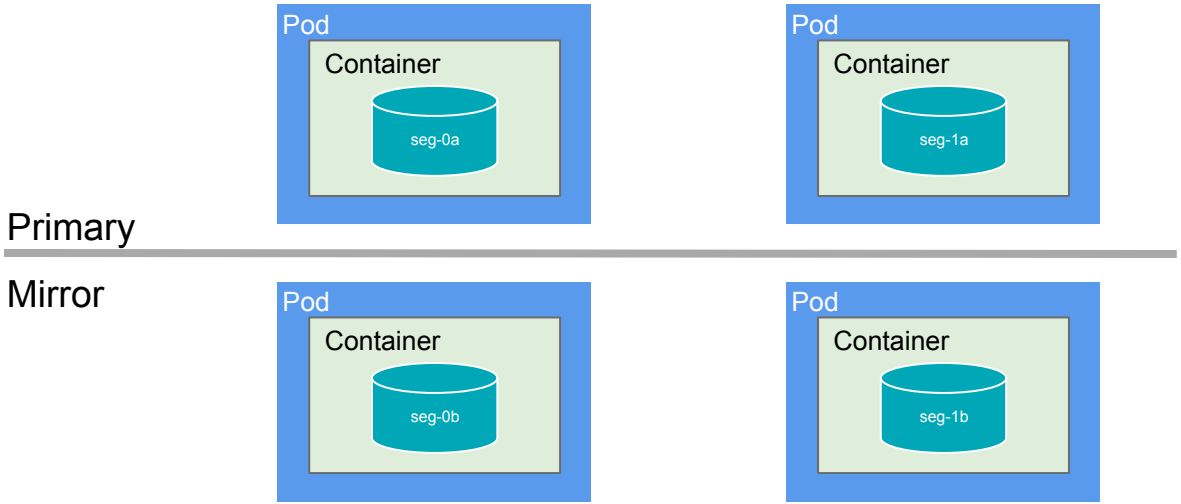
We can always discover a container by DNS.

For example, DNS address for different roles:

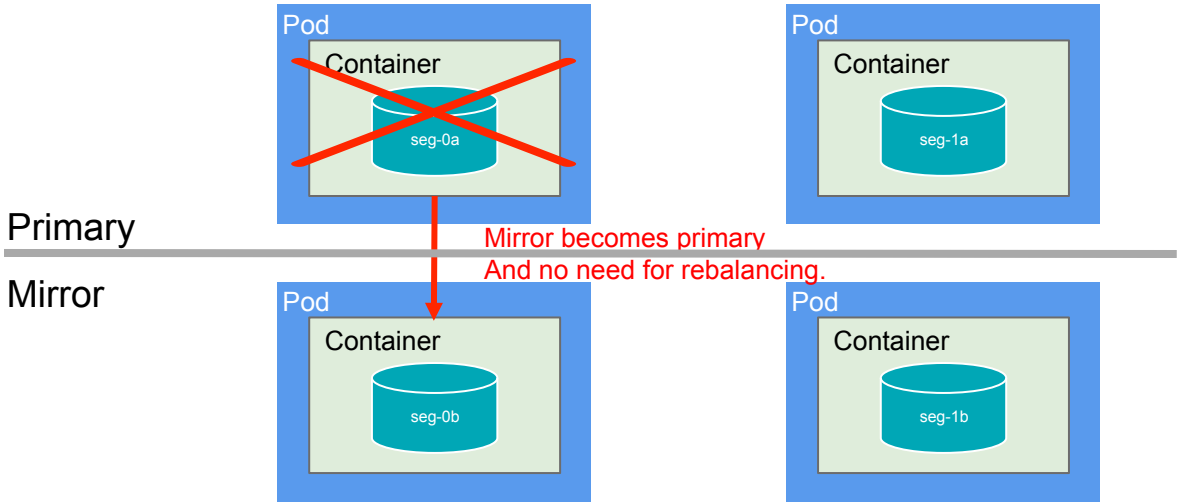
master.greenplum.svc.cluster.local
standby.greenplum.svc.cluster.local
segment-0a.greenplum.svc.cluster.local
segment-0b.greenplum.svc.cluster.local



3. HA without Rebalancing

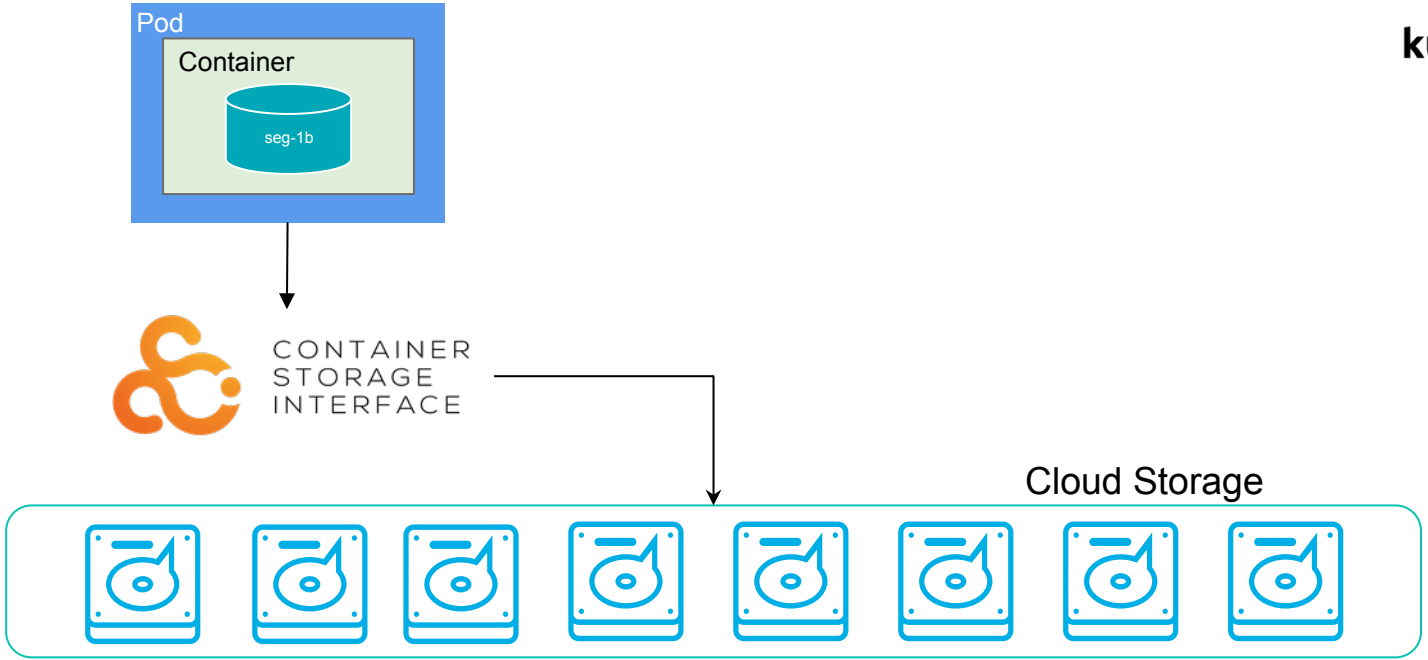


3. HA without Rebalancing

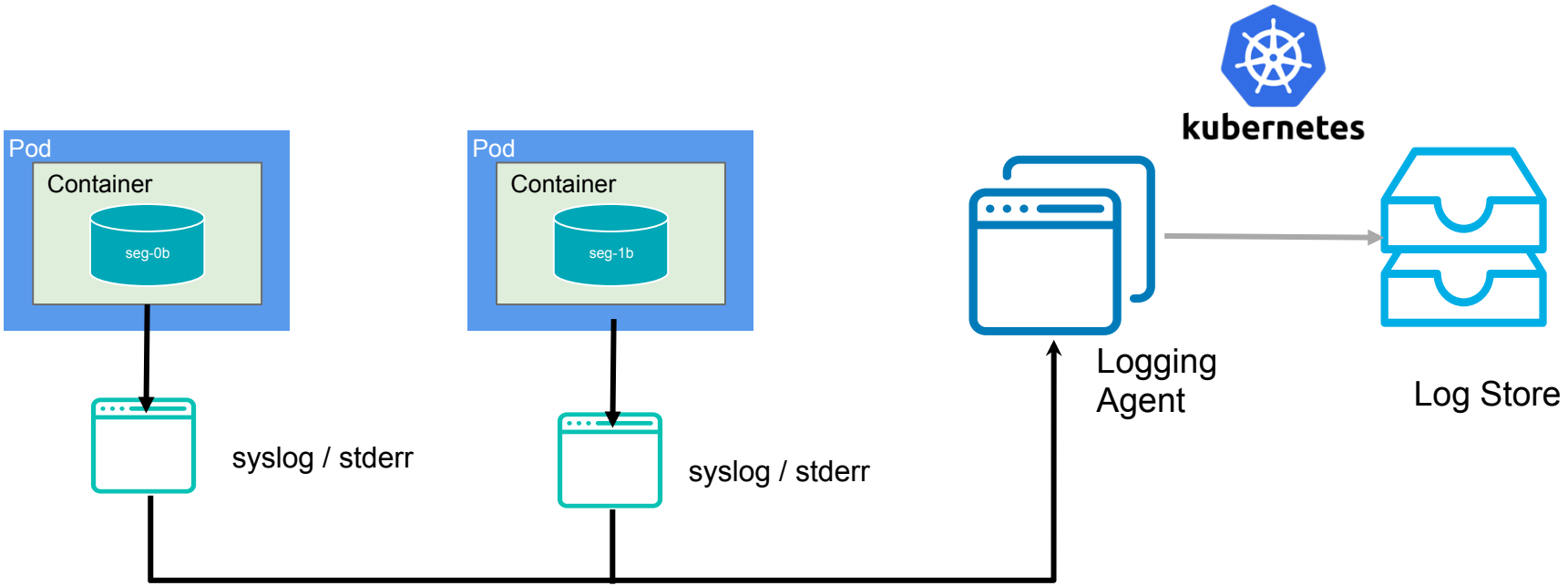


4. Kubernetes Plugins Support : Container Storage Interface

kubernetes



4. Kubernetes Plugins Support : Logging

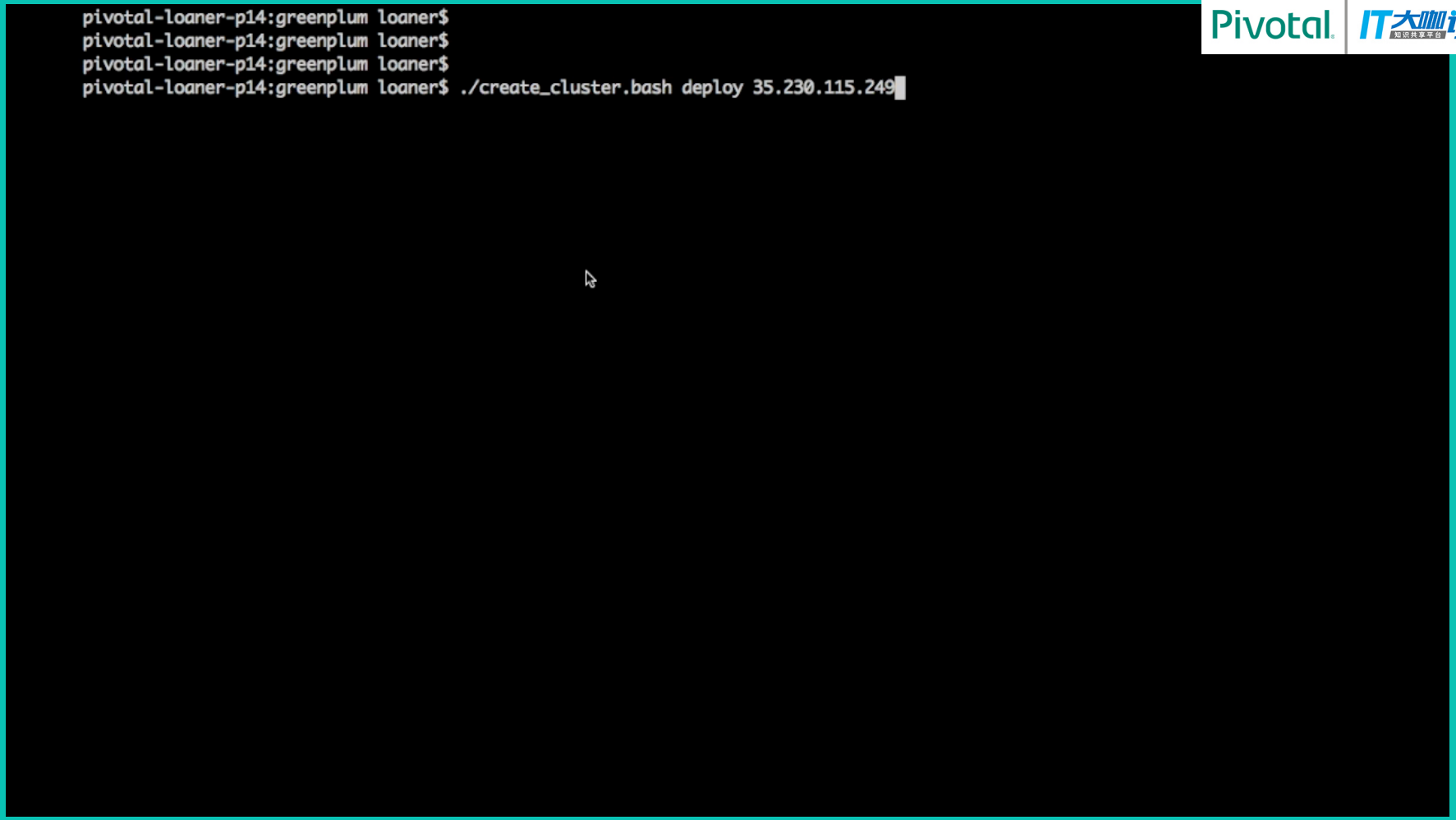


GREENPLUM ON PKS DEMO

HEY PKS! GIVE ME A GREENPLUM CLUSTER OF "N" SEGMENTS

Demo

Deploy Greenplum on PKS



Demo

Expand Greenplum on PKS

Demo

Greenplum Segment Failover


pivotal-loaner-p14:greenplum loaner$
pivotal-loaner-p14:greenplum loaner$
pivotal-loaner-p14:greenplum loaner$
pivotal-loaner-p14:greenplum loaner$

How to

Automate Greenplum operation on PKS

Pivotal

How to

- Auto segment failover
- Auto activate standby if master failed
- Auto expand cluster

Greenplum on PKS

Greenplum Operator on Kubernetes

Kubernetes Operator

Custom Resource Definitions (CRD)

- Provides API object framework for a new resource

- Used to implement new resource types
  - Data model for Controllers

Controllers

- Active reconciliation process
  - Current State -> Desired State

- Used to automate app administration
  - {add, extend, replace} cluster functionality

Kubernetes Operator

Custom Resource Definitions (CRD)

```yaml
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
metadata:
 name: greenplums.greenplum.pivotal.io
spec:
 group: greenplum.pivotal.io
 version: v1
 scope: Namespaced
 names:
   plural: greenplums
   kind: greenplum
 validation:
    # openAPIV3Schema is the schema for validating custom objects.
    openAPIV3Schema:
      properties:
        spec:
          properties:
            cidr:
              type: string
              # cidr regex
              pattern: '^([0-9]{1,3}\.){3}[0-9]{1,3}(\/([0-9]|[1-2][0-9]|3[0-2]))?$'
            segmentCount:
              type: integer
              minimum: 1
              maximum: 10
```

Greenplum Custom Resource and Usage

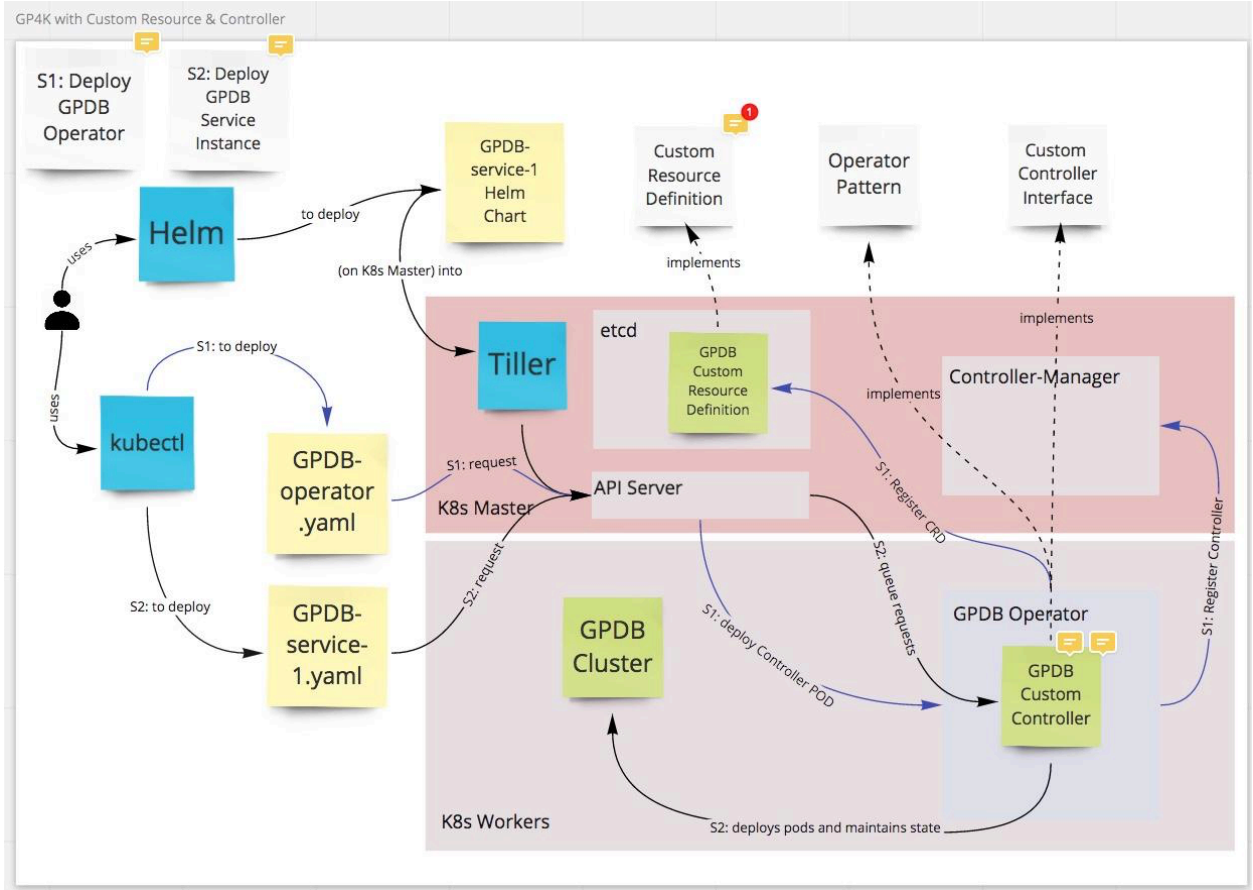
```yaml
apiVersion: greenplum.pivotal.io/v1
kind: greenplum
metadata:
 name: gpdb-service-1
spec:
 serviceAccountName: greenplum
 segmentCount: 1
 dataDir: /greenplum
 cidr: 0.0.0.0
 image: gcr.io/data-gpbosh/gpdb-ubuntu-kubernetes-oss:latest
 imagePullPolicy: Always
```

```
# S1: create operator inside K8s
kubectl create -f greenplum-operator.yaml

# S2: using greenplum operator to provision a service using greenplum custom resource
kubectl create -f gpdb-service-1.yaml
```

Greenplum on Kubernetes

How greenplum operator works on Kubernetes



Future Work

More Persistent Volumes
Custom Scheduler
Resource Management

QUESTIONS?

Transforming How The World Builds Software