

容器云平台在企业中的运维管理及场景实践

—— 时速云 王磊

manager

20



innovation

management

leadership

20%

teamwork



Agenda

- Why Enterprise need Private PaaS?
- What Enterprise Cares about
 - Network
 - Storage
 - CI/CD
 - Application Package and Deployment
 - Configuration Management
 - Secret Data
 - High Availability
- How to Manage Your PaaS

Why Enterprise Need Private PaaS?

Hybrid Cloud



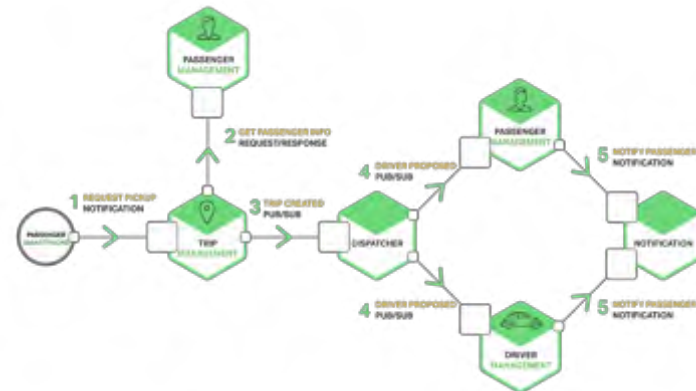
Cloud Enablement of Existing and New Applications



Private Cloud Strategy



Microservices and APIs

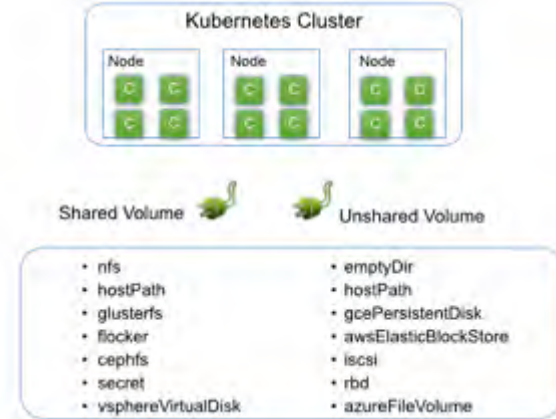


Requirements for Private PaaS?

Ease of deployment and management



Data layer separation



Integration with Legacy Systems



Extensibility



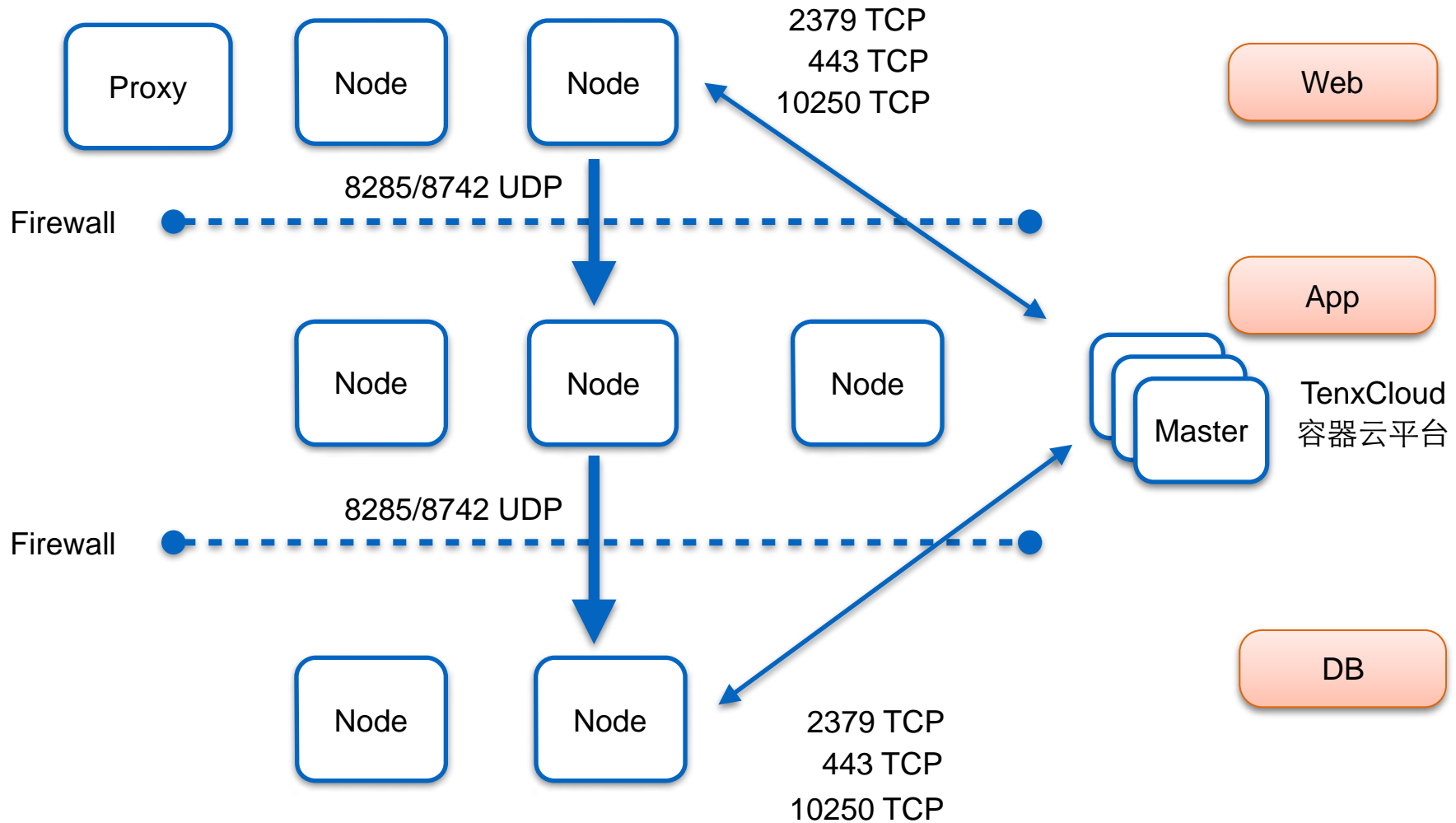
Portability through Standards



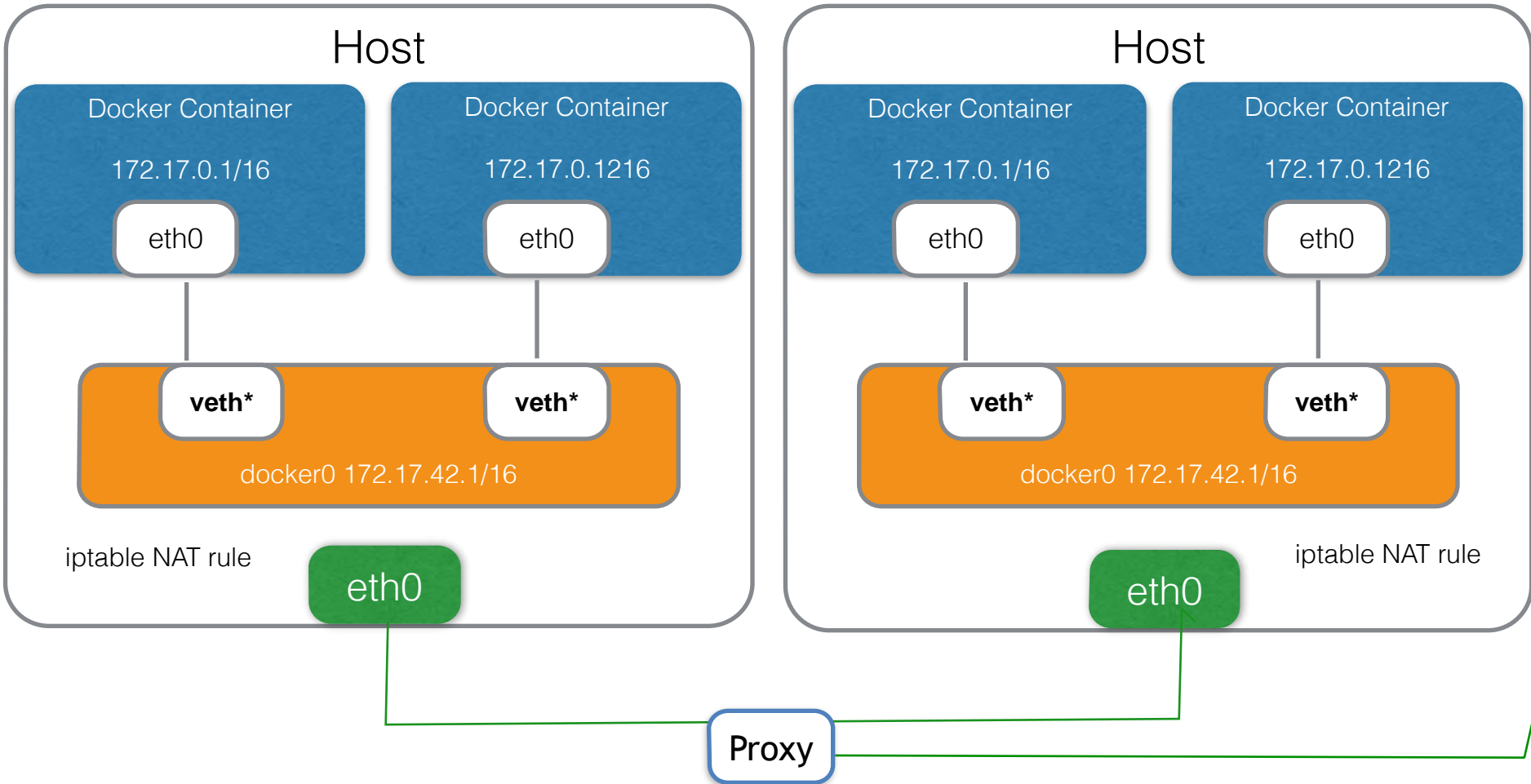
What Enterprise Cares about?

- Network
- Storage
- CI/CD
- Application Package and Deployment
- Configuration Management
- Secret Data
- Monitor and Alert
- High Availability

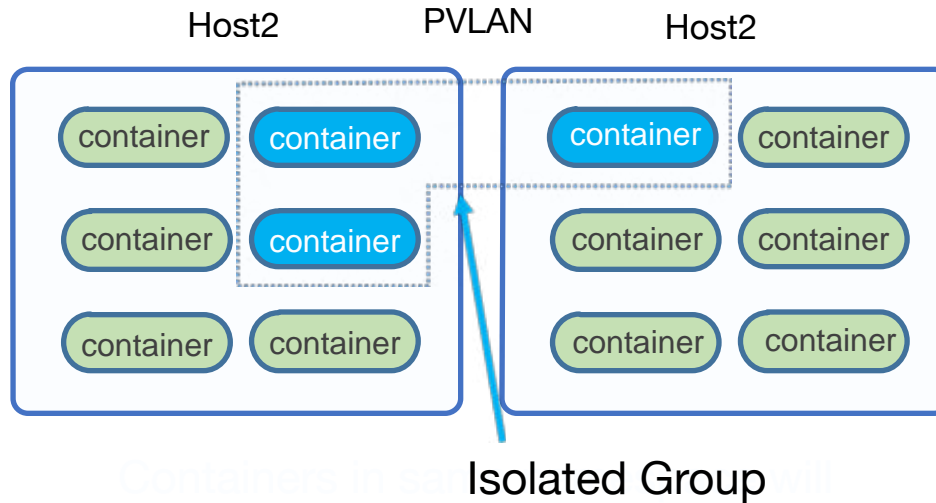
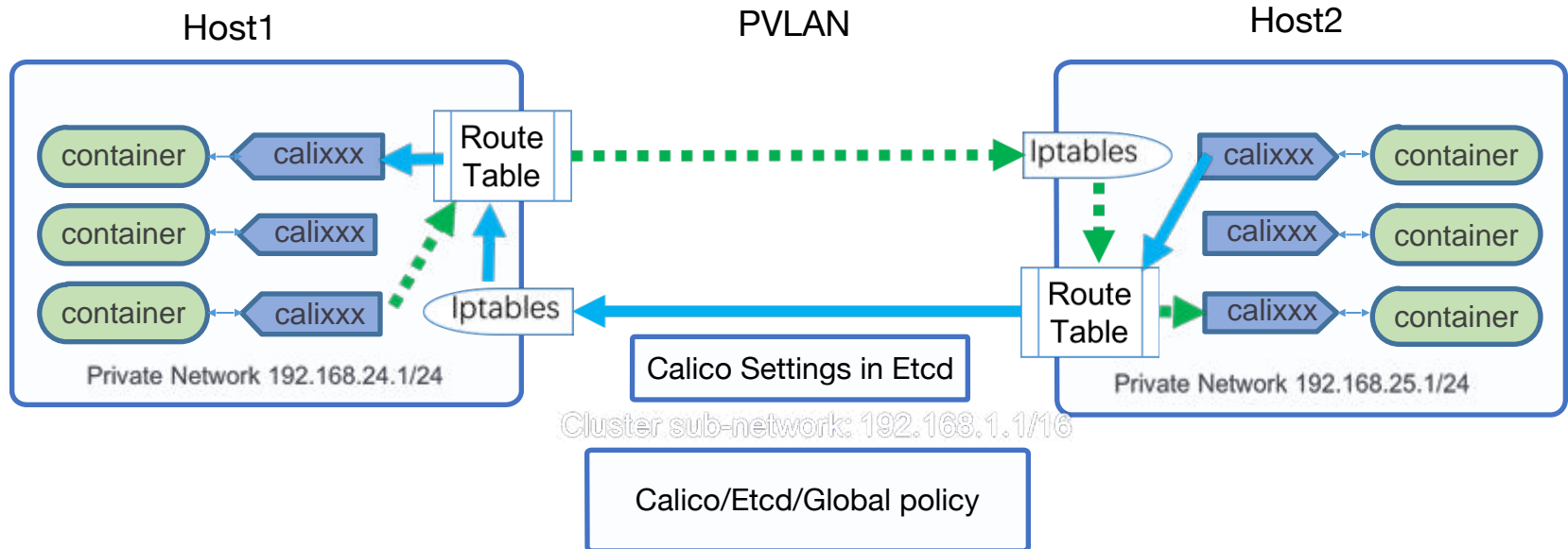
Network - Flannel



Network - Docker Bridge

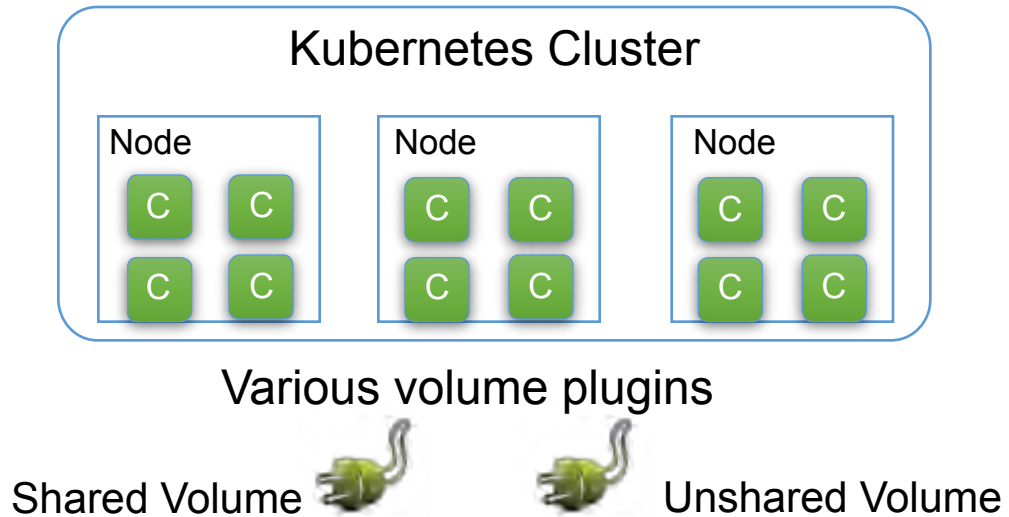


Network - Calico



Storage - Extensible Volume Support

- Easy to add customized plugin to use other storage backend following volume interface



- nfs
- hostPath
- glusterfs
- flocker
- cephfs
- secret
- vsphereVirtualDisk
- emptyDir
- hostPath
- gcePersistentDisk
- awsElasticBlockStore
- iscsi
- rbd
- azureFileVolume

Storage - Add your volume plugin

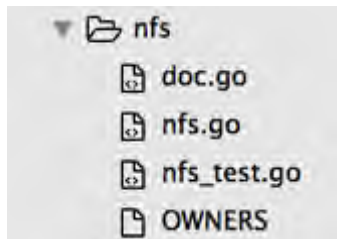
1. Register in kubelet entry

kubelet/app/plugins.go

```
allPlugins = append(allPlugins, customizedPlugin.ProbeVolumePlugins()...)
```

2. Implement it in the package below:

pkg/volume/<your_plugin>, interface can refer to volume.go



- ProbeVolumePlugins
- Init
- CanSupport
- SetUp
- TearDown
- ...

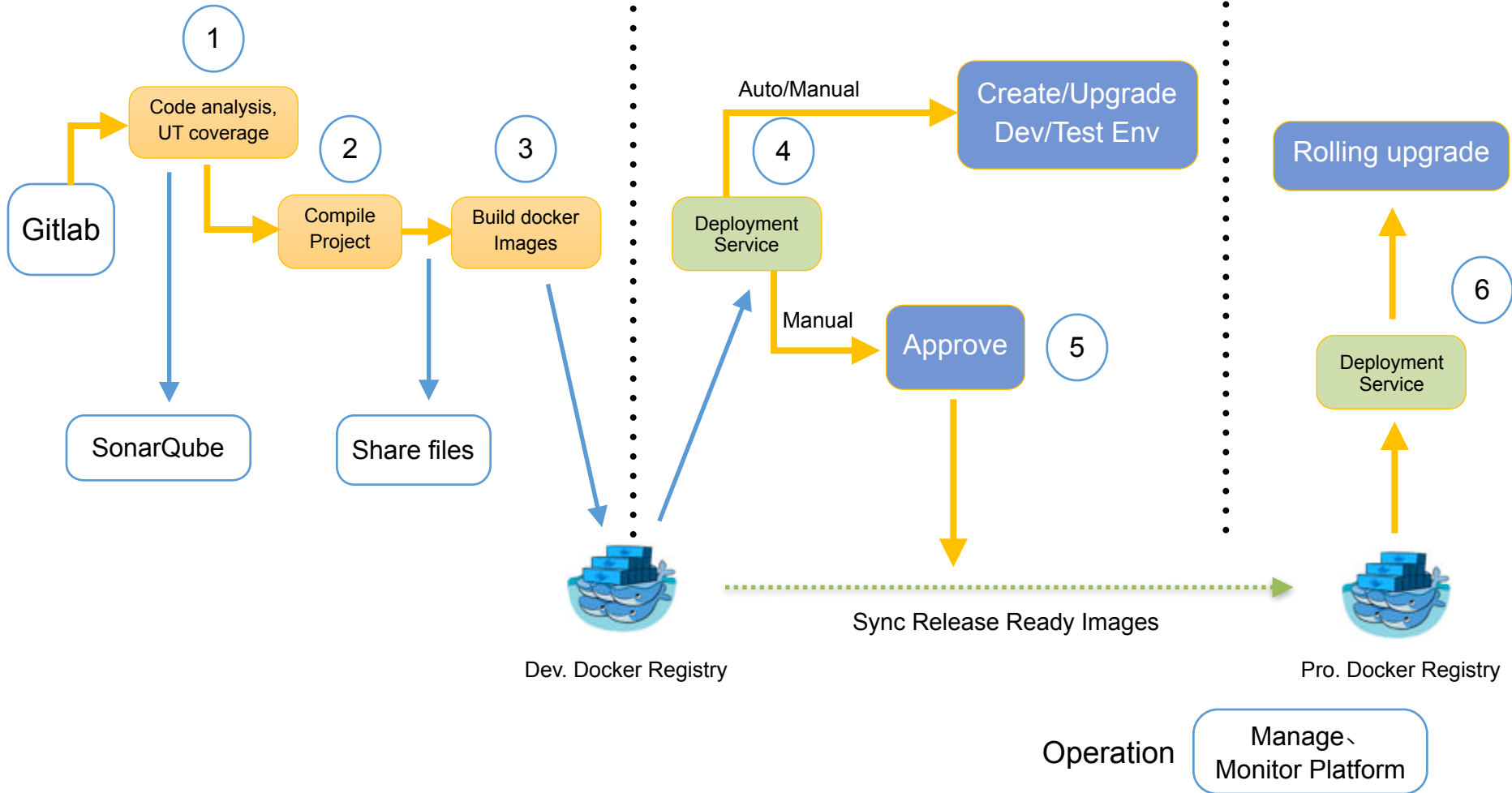
3. Update API Spec

CI/CD - Typical

Development Env.

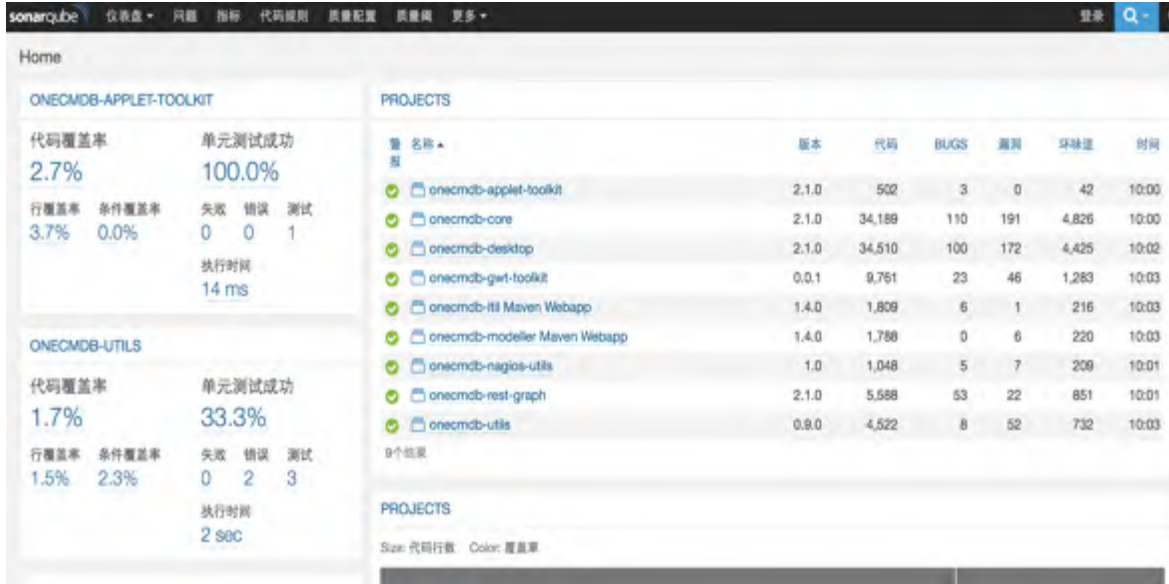
Testing Env.

Production Env.



CI/CD - Example

1



3



tenxcloud/tomcat-base:8.0
ADD *.war /tomcat/webapps/

Share war files to final image

Compile to generate war files

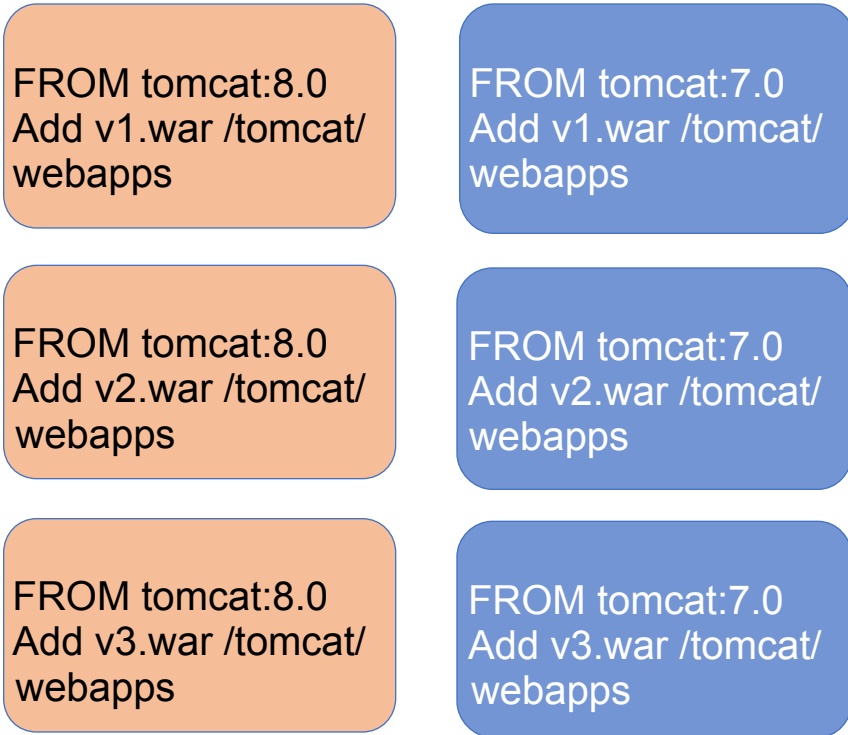
2

itil.war
onecmdb-iti-1.4.0.war
onecmdb-modeller.war
ROOT.war

No source code in final images
Only compiled code

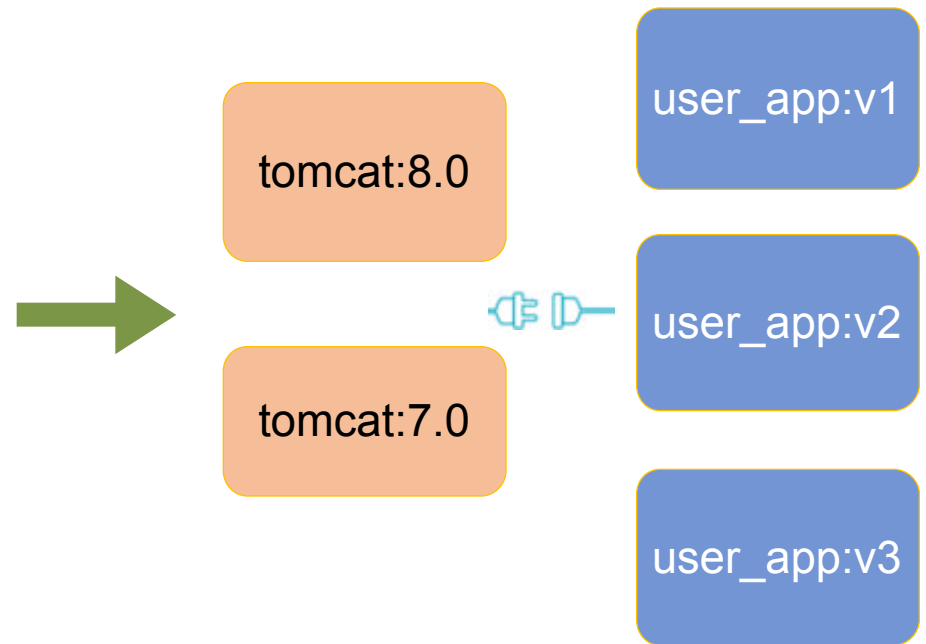
Application Package and Deployment

Common Deployment Approach



$M * N$ number of images

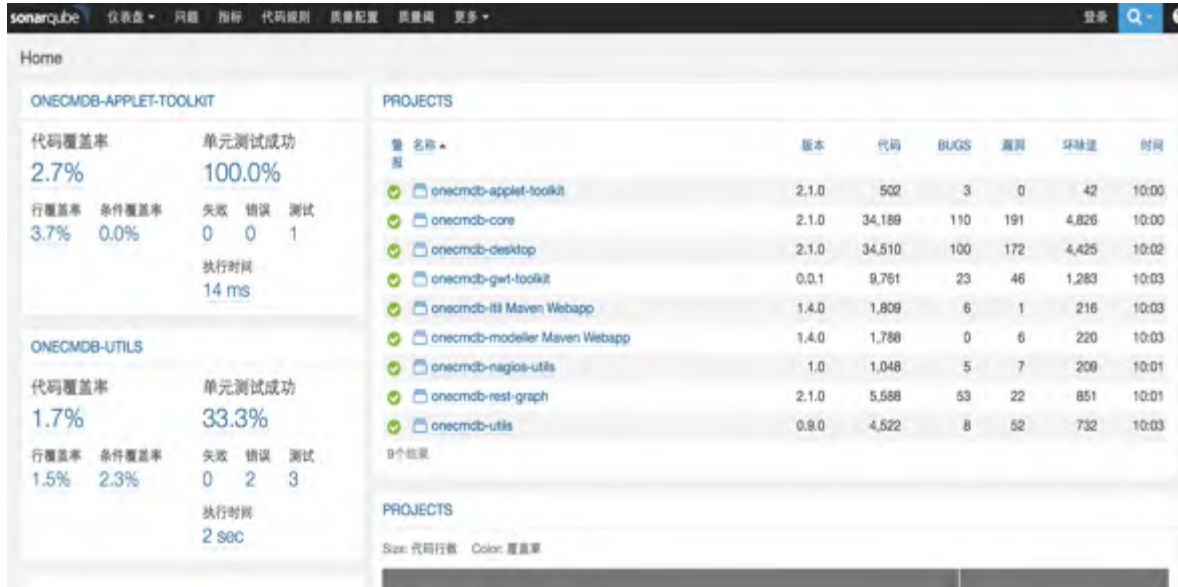
Deploy using Pod Orchestration Approach



$M + N$ number of images

CI/CD - Example

1



2

Compile to generate war files

- itil.war
- onecmdb-til-1.4.0.war
- onecmdb-modeller.war
- ROOT.war

3

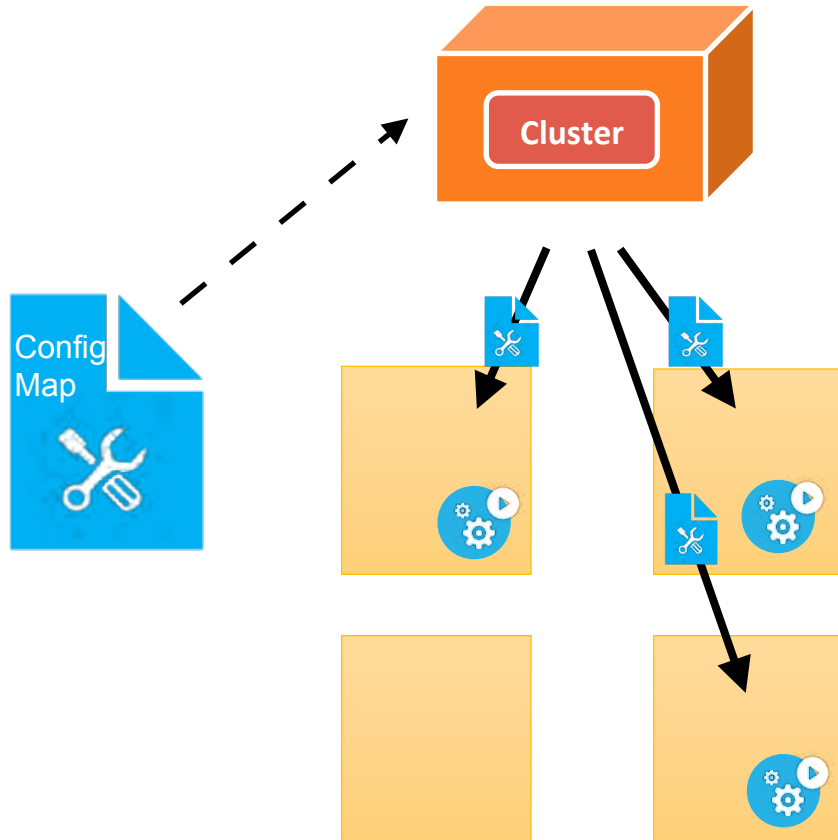


```
tenxcloud/busybox
ADD *.war /app_dir
```

Share war files to final image

No source code in final images
Only compiled code

Configuration Management

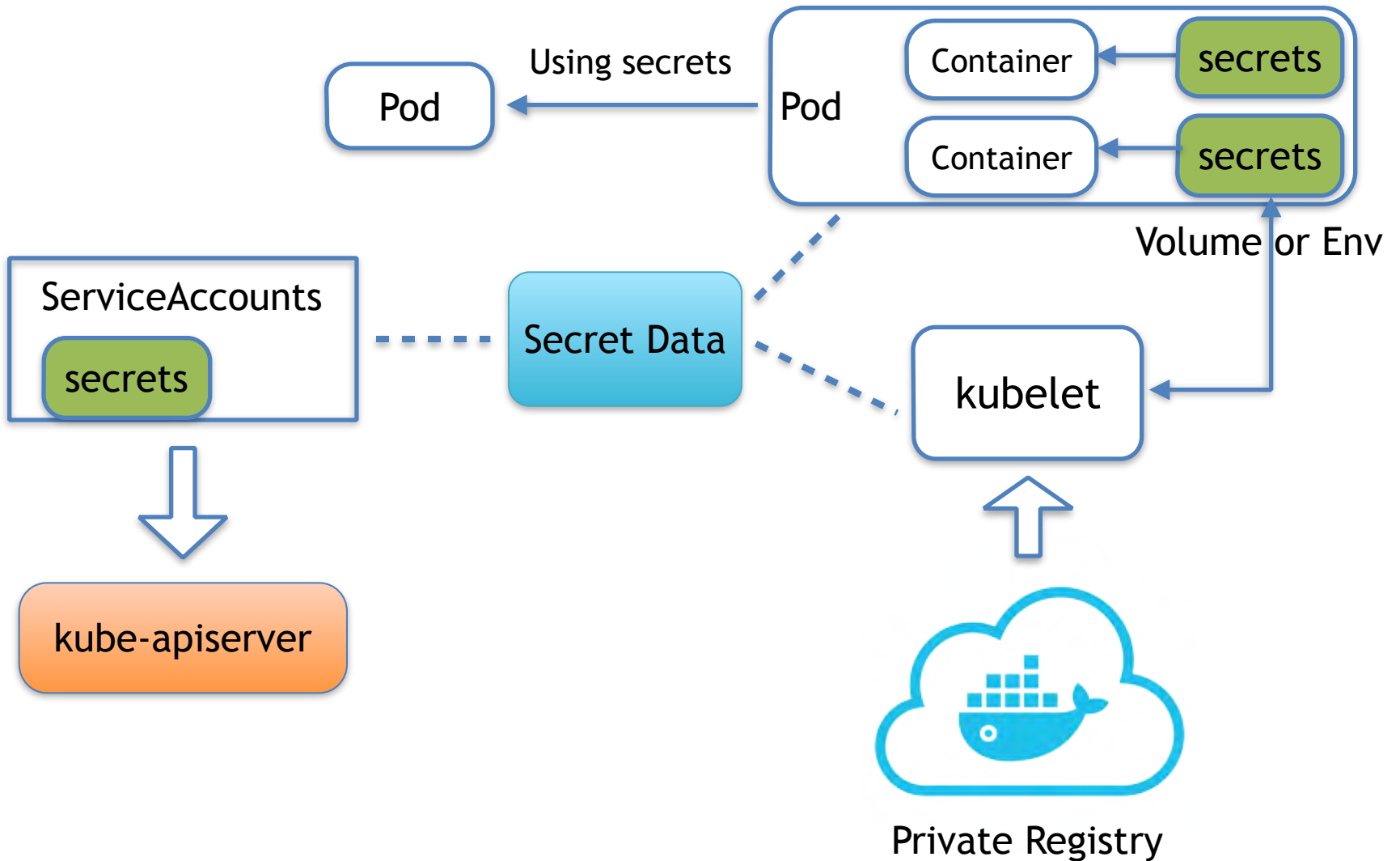


Centralized and cluster level configuration management

Features:

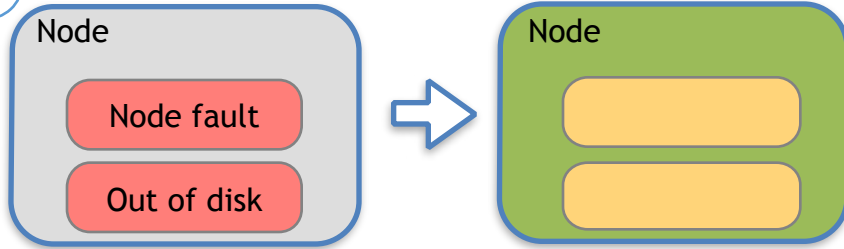
- ✓ • Decouple config from image content
- ✓ • Support environment variables, command-line arguments or as a volume
- ✓ • Update associated application node automatically with new config once updated

Secret Data



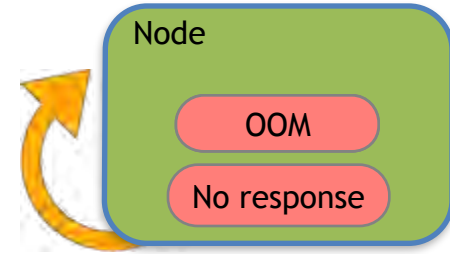
High Availability

1



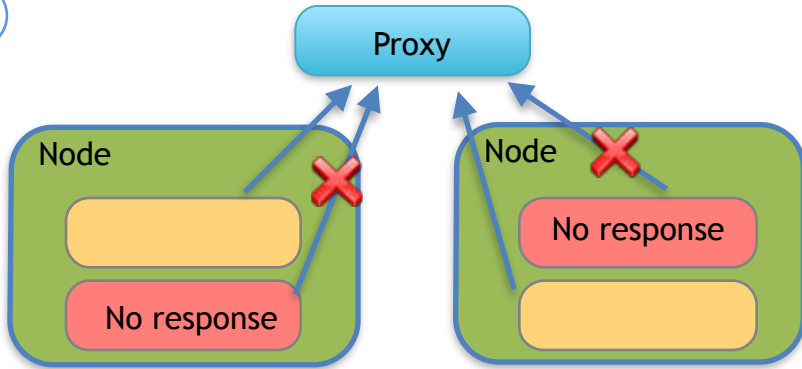
Application will be migrated to other node

2



Application will be restarted on the same node

3



Application will be removed from endpoints of proxy

4



Master HA

5

Multiple zones HA

6

Federation clusters

Agenda

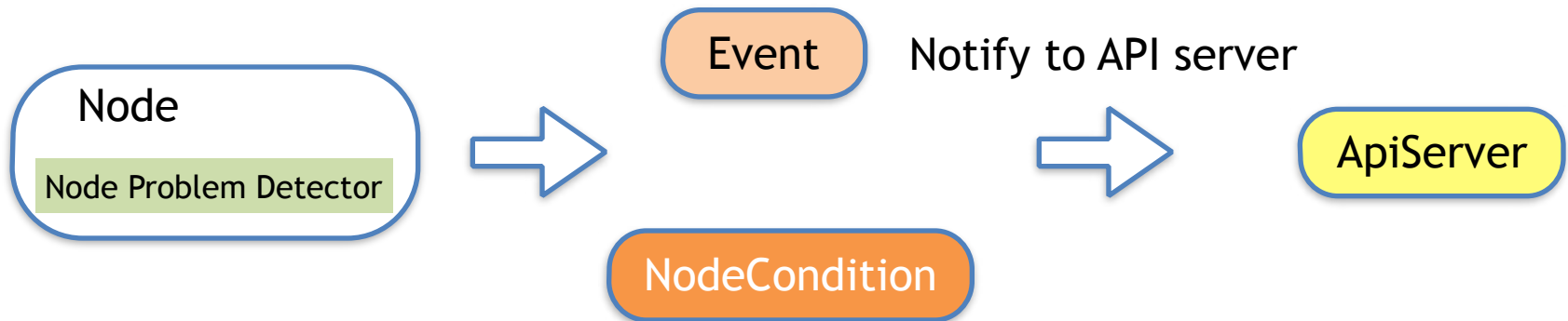
- Why Enterprise need Private PaaS?
- What Enterprise Cares about
 - Network
 - Storage
 - CI/CD
 - Application Package and Deployment
 - Configuration Management
 - Secret Data
 - Monitor and Alert
 - High Availability
- **How to Manage Your PaaS**

Monitor and Alert

- Platform
- Kubernetes Cluster- Components
- Node
- Service
- CPU
- Memory
- Disk
- QPS

Node Problem Detector

- A DaemonSet detects node problems and reports them to APIServer.



```
Events:
  FirstSeen    LastSeen    Count   From              SubobjectPath    Type      Reason
  ----
  Id           46m        5       (controllermanager-)  Normal      NodeNotReady      Node 192.168.1.189 status is now: NodeNotReady
  Id           46m        4       (kubelet 192.168.1.189) Normal      NodeHasSufficientDisk  Node 192.168.1.189 status is now: NodeHasSufficient
Disk
  Id           46m        1       (kubelet 192.168.1.189) Normal      NodeReady         Node 192.168.1.189 status is now: NodeReady
  4m           1m         2       (kernel-monitor 192.168.1.189) Warning     TaskKilling       task docker:20744 blocked for more than 120 seconds
```

Operation

KUBECTL

- Debug
 - `kubectl create/get/describe/delete (type name) --namespace=<ns_name>`
- Node maintenance
 - `kubectl drain/uncordon <node>`
- `kubectl patch`
 - `kubectl patch (type name) -p PATCH`
- Resource annotation
 - `kubectl annotate (type name) key=value`
- `kubectl label`
 - `kubectl annotate (type name) key=value`

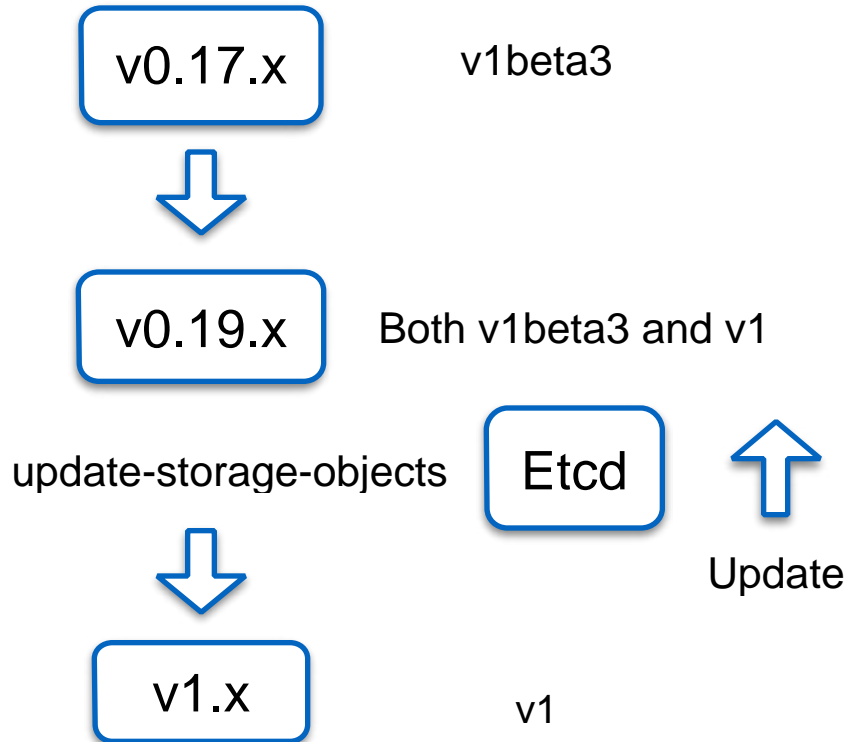
KOPS

Clusters operation tool

- `kops create cluster`
- `kops update cluster`
- `kops get cluster`
- `kops delete cluster`



Upgrade



- ✓ No hard breaking changes over version boundaries

• Kubernetes Version Definition

- X.Y.Z: X=>major, Y=>minor, Z=>patch
- API vX[betaY]

• Upgrade

- ✓ Upgrade from 1.x to any other 1.x release as rolling upgrade
 - Master first and then each node
- ✓ Run latest patch releases of a given minor release
- ✗ Upgrade more than two minor releases at a time

Thanks!



官网: <https://www.tenxcloud.com>