



KubeCon



CloudNativeCon

China 2018

# Protecting Stateful Workloads with CSI Snapshot

Xing Yang (Huawei) & Jing Xu (Google)



# Data in the Cloud



KubeCon

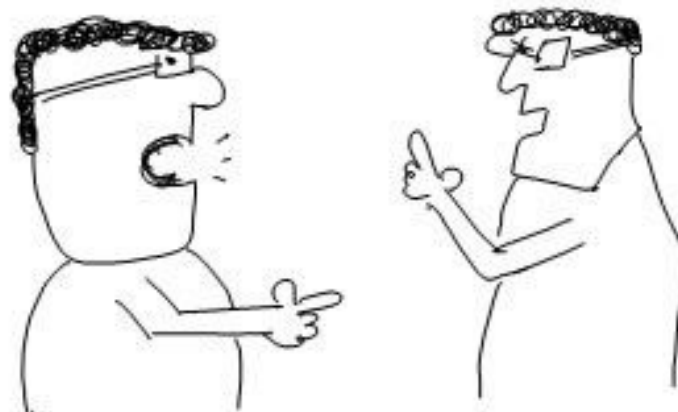


CloudNativeCon

China 2018

WHERE THE HECK  
IS MY DATA?

IT'S THERE, UP  
IN THE CLOUDS.



Brainstuck.com

Source: <https://brainstuck.com/2009/>

# Data in the Cloud



KubeCon



CloudNativeCon

China 2018

Is my data  
safe in the  
cloud?



Yeah, until  
it rains.



Brainstuck.com



Source: <https://brainstuck.com/2009/>



# It might rain anytime anywhere ...

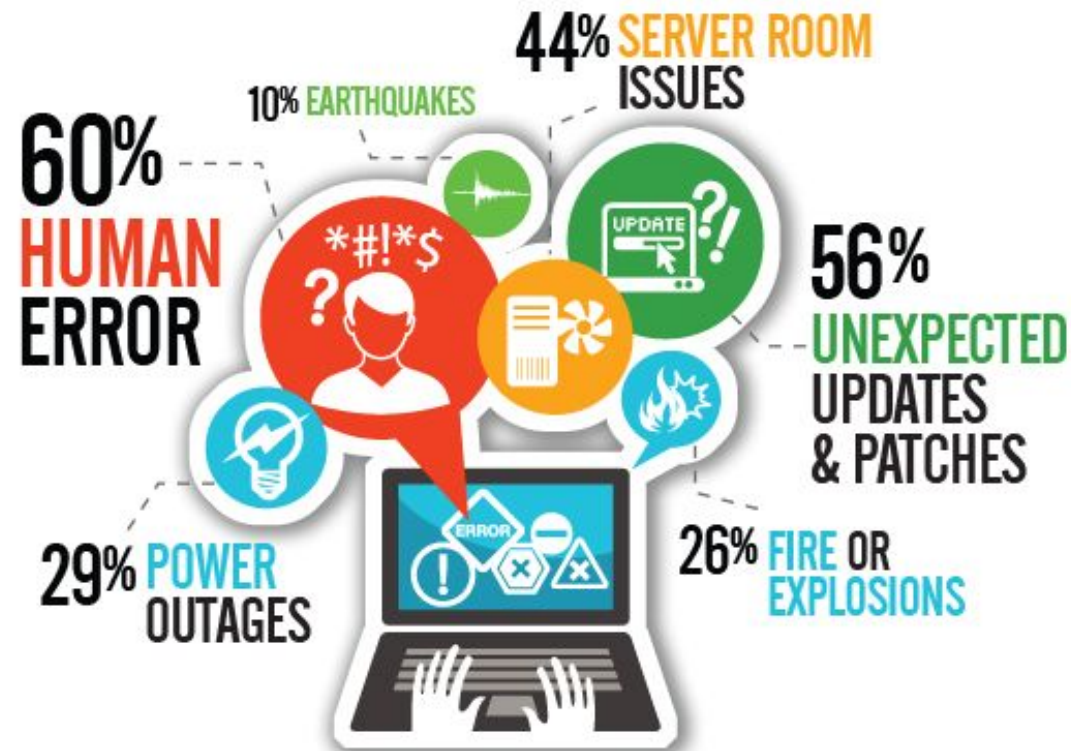


KubeCon



CloudNativeCon

China 2018



Source: [Disaster Recovery Preparedness Council](#) (2014 DR Benchmark Survey)

# Snapshot anytime anywhere

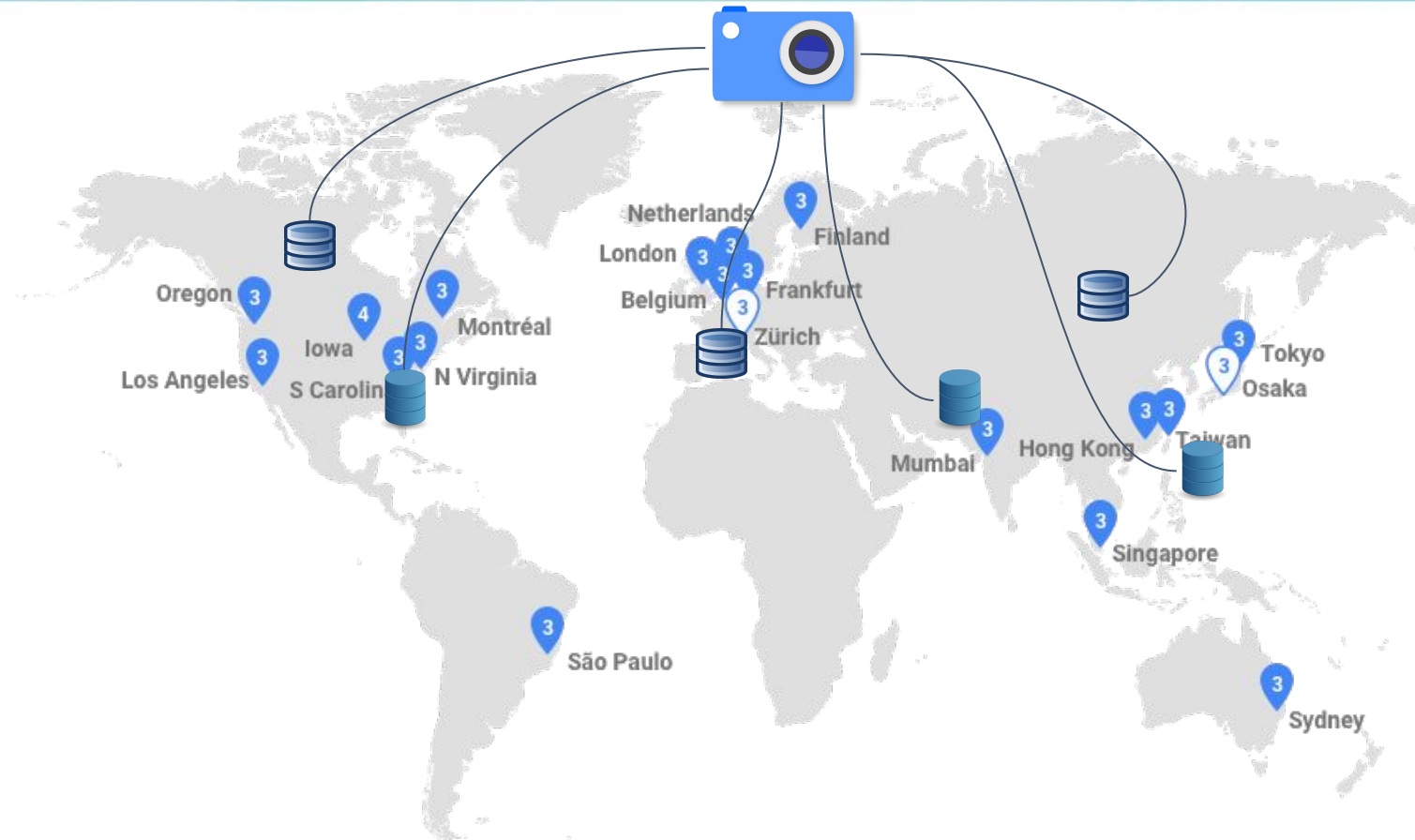


KubeCon



CloudNativeCon

China 2018



Google Cloud Platform Locations

# Agenda



KubeCon



CloudNativeCon

China 2018

- Motivation
  - Why snapshot in the cloud
- Background
  - Kubernetes Persistent Volumes and storage plugins
  - CSI Introduction
- CSI Snapshot Design
  - CSI snapshot support
  - Kubernetes Snapshot Concepts
  - CSI Snapshot Management and Deployment
- Demo

# Kubernetes Persistent Volumes



KubeCon



CloudNativeCon

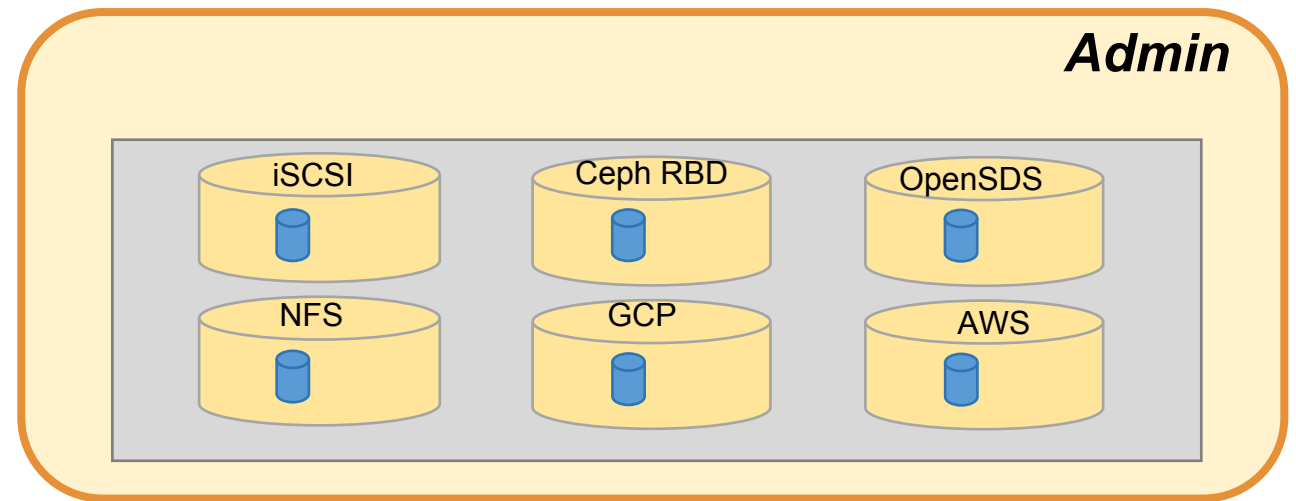
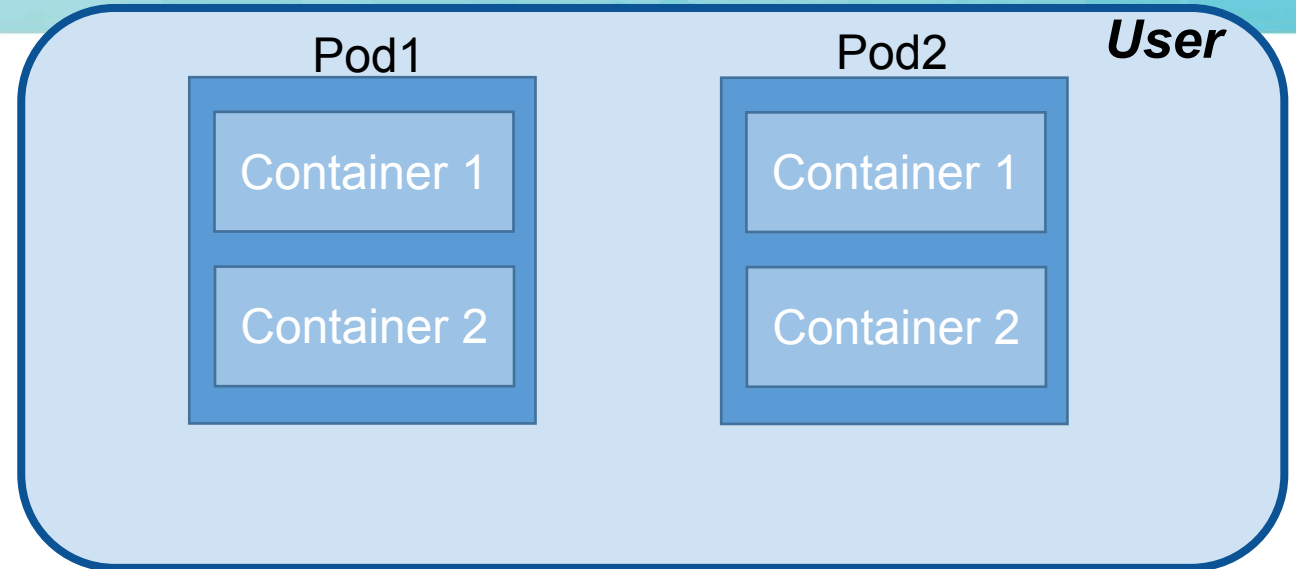
China 2018

## Users

- Request storage

## Admins

- Allocate storage



# Kubernetes Persistent Volumes



KubeCon



CloudNativeCon

China 2018

## Users

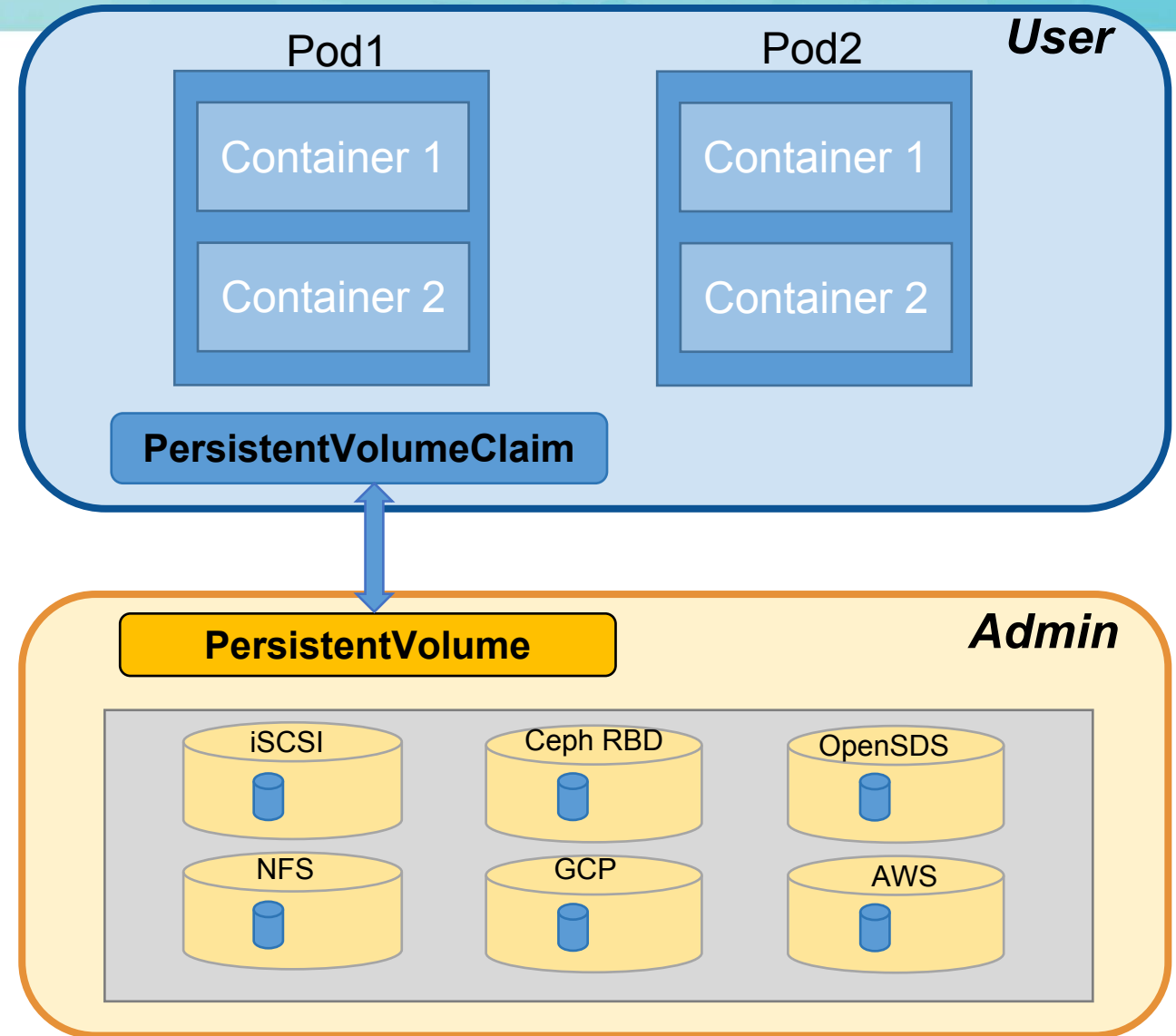
- Request storage
- **PersistentVolumeClaim (PVC)**

## Admins

- Allocate storage
- **PersistentVolume (PV)**

## Binding

- Match PVC and PV





# Kubernetes Persistent Volumes



KubeCon



CloudNativeCon

China 2018

## Users

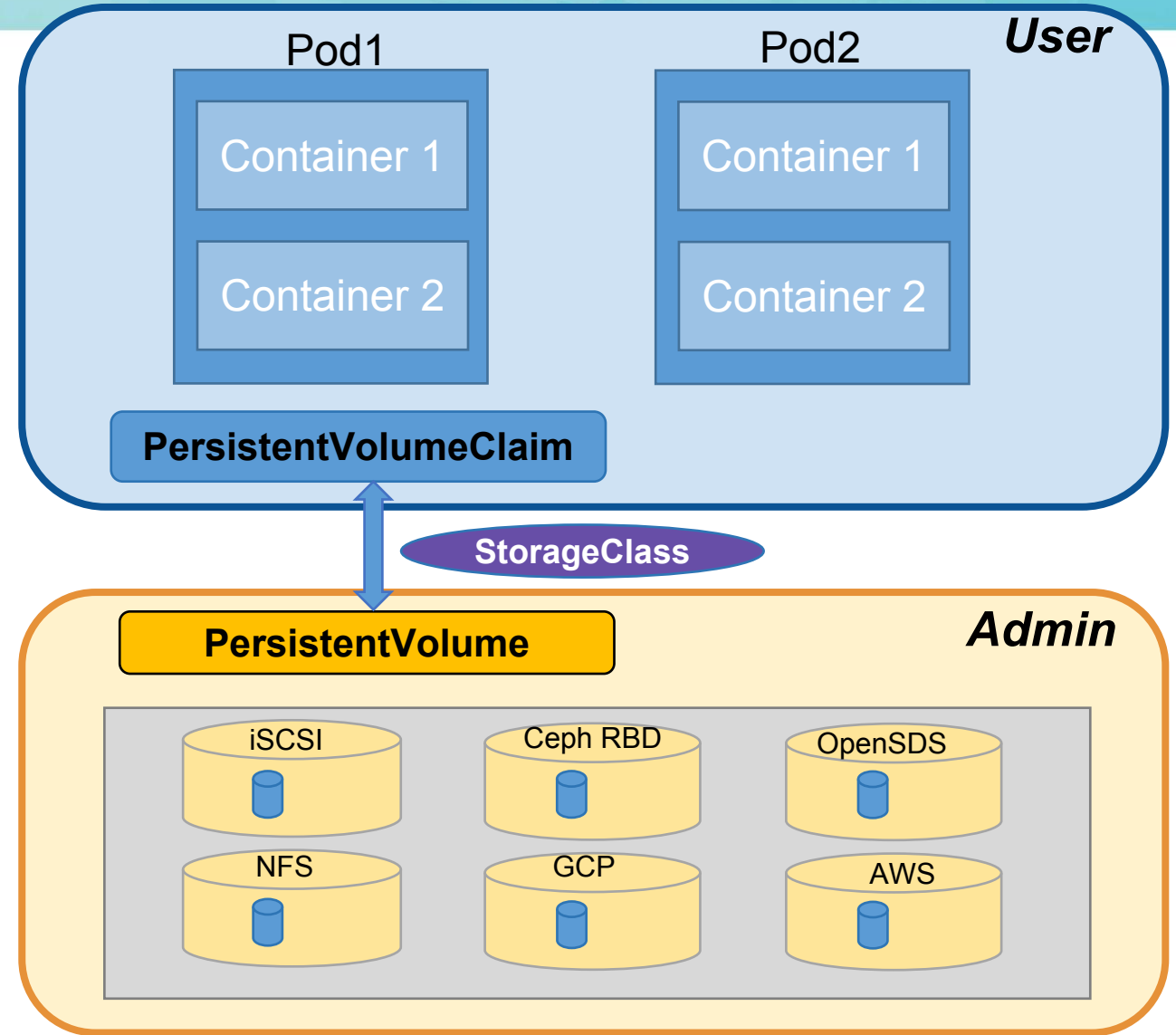
- Requires storage
- PersistentVolumeClaim (PVC)

## Admins

- Allocates storage
- PersistentVolume (PV)

## Dynamic Provisioning

- StorageClass



# Kubernetes Storage Management

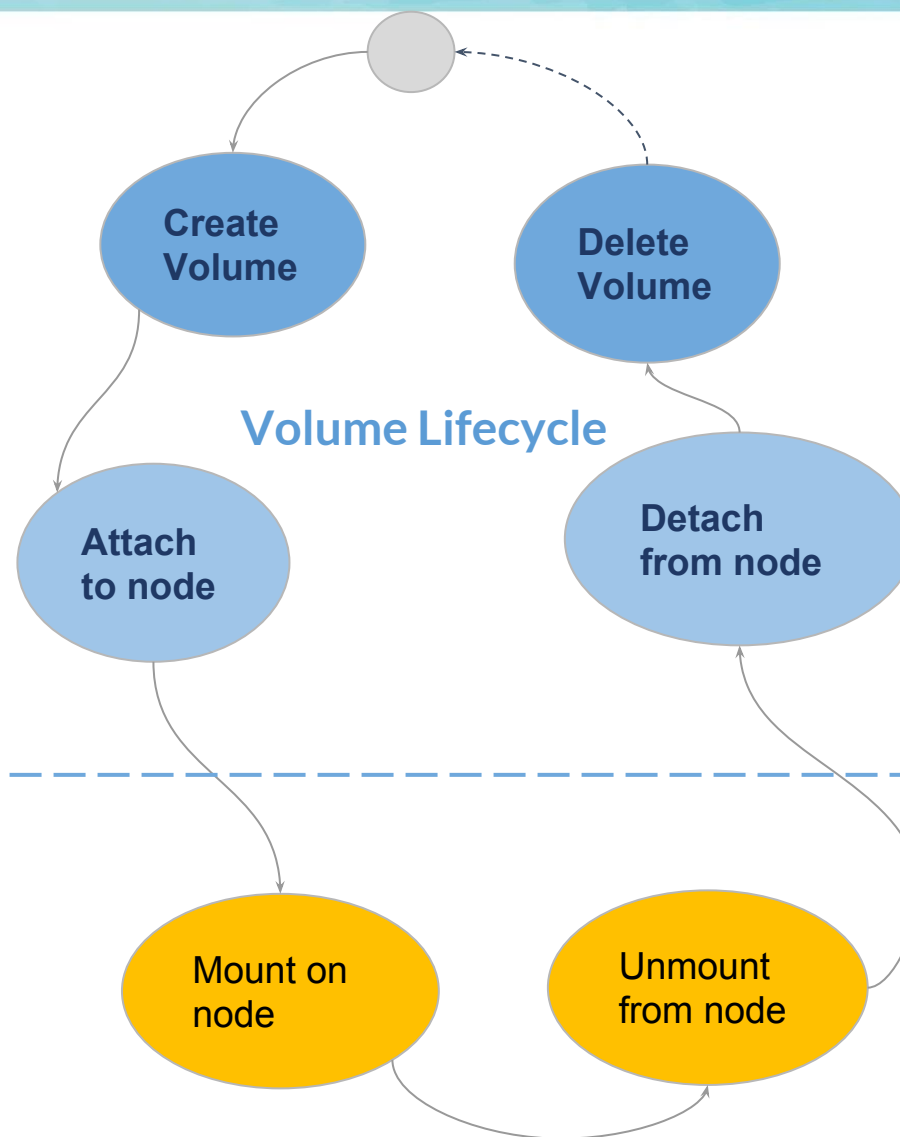


KubeCon



CloudNativeCon

China 2018



# Kubernetes Storage Management



KubeCon

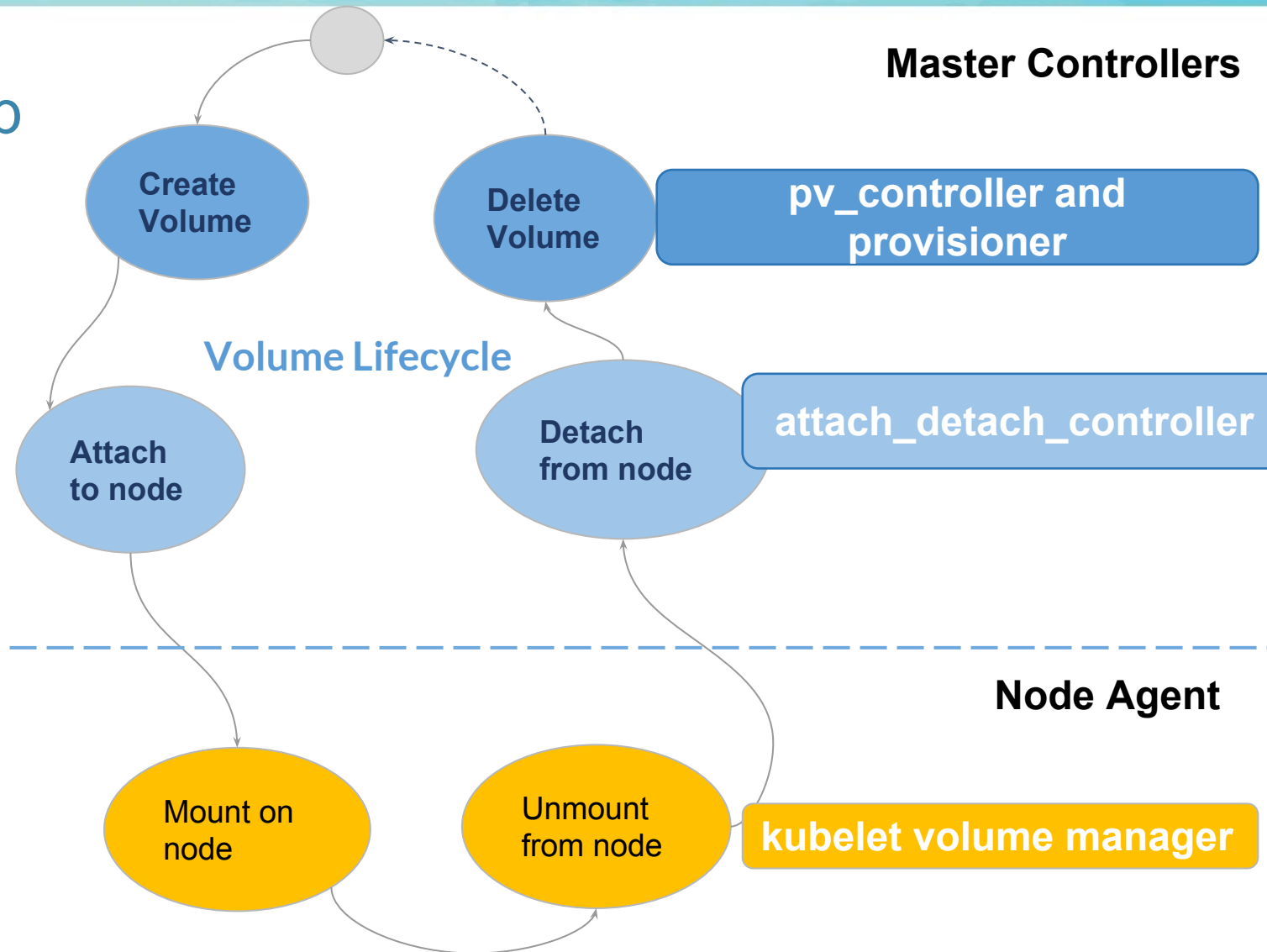


CloudNativeCon

China 2018

## Controller reconciliation loop

```
for {  
    desired := getDesiredState()  
    current := getCurrentState()  
    makeChanges(desired, current)  
}
```



# Storage Plugins



KubeCon

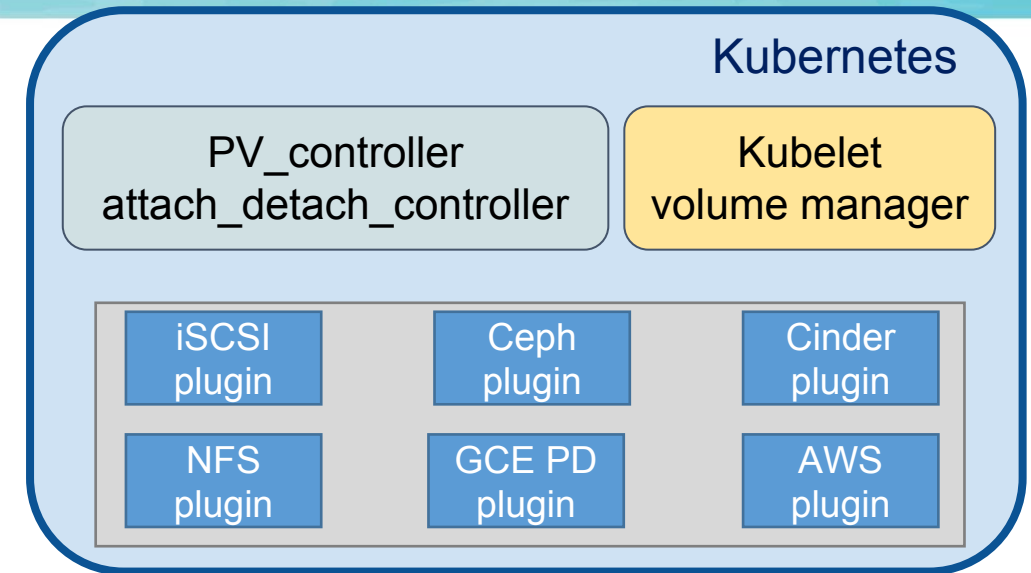


CloudNativeCon

China 2018

## Previously, In-tree plugins

- Coupled release cycles
- Force plugins to be open source
- Testing burden
- Root access to install binaries



Branch: master ▾ [kubernetes](#) / [pkg](#) / [volume](#) /

**k8s-ci-robot** Merge pull request #60195 from humblec/glusterfs-pvspec-3 ...

..

<a href="#">awsebs</a>	Merge pull request #67851 from aniket-s-k
<a href="#">azure_dd</a>	Merge pull request #67851 from aniket-s-k
<a href="#">azure_file</a>	Merge pull request #69718 from andyzhan
<a href="#">cephfs</a>	fix golint errors in pkg/volume/*
<a href="#">cinder</a>	adding support for expanding in use persis
<a href="#">configmap</a>	fix golint errors in pkg/volume/*
<a href="#">csi</a>	Merge pull request #70540 from gnufied/fi:



# Container Storage Interface



KubeCon



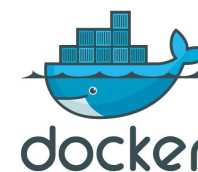
CloudNativeCon

China 2018

## Container Orchestration Systems



MESOS



CLOUD  
FOUNDRY

## Storage Providers

GCE PD

AWS EBS

Ceph RBD

Cinder

3rd Party  
Storage

# Container Storage Interface



KubeCon



CloudNativeCon

China 2018

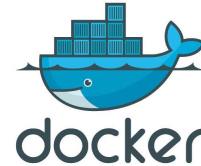
## What's new?

- Out-of-tree
- Services based
- Idempotency

## Container Orchestration Systems



MESOS



CLOUD  
FOUNDRY



## Container Storage Interface

iSCSI  
plugin

NFS plugin

GCE PD  
plugin

Ceph  
plugin

Cinder  
plugin

AWS  
plugin

## Storage Providers

GCE PD

AWS EBS

Ceph RBD

Cinder

3rd Party  
Storage

# Container Storage Interface

## What's new?

- Out-of-tree
- Services based
- Idempotency

## Container Orchestration Systems



## Container Storage Interface



iSCSI  
plugin

NFS plugin

GCE PD  
plugin

Ceph  
plugin

Cinder  
plugin

AWS  
plugin

## Storage Providers

GCE PD

AWS EBS

Ceph RBD

Cinder

3rd Party  
Storage

**CSI 1.0 will be available very soon!**  
**New features will be only added to CSI**

# CSI API Overview

## 3 core gRPC services

- Controller service
  - handles create/delete volume and runs on a node
- Node service
  - supports mount/unmount volume and runs on every worker node
- Identity service
  - provides info and capabilities of a plugin

```
service Identity {  
    rpc GetPluginInfo(GetPluginInfoRequest)  
        returns (GetPluginInfoResponse) {}  
  
    rpc GetPluginCapabilities(GetPluginCapabilitiesRequest)  
        returns (GetPluginCapabilitiesResponse) {}  
  
    rpc Probe (ProbeRequest)  
        returns (ProbeResponse) {}  
}  
  
service Controller {  
    rpc CreateVolume (CreateVolumeRequest)  
        returns (CreateVolumeResponse) {}  
  
    rpc DeleteVolume (DeleteVolumeRequest)  
        returns (DeleteVolumeResponse) {}  
  
    rpc ControllerPublishVolume (ControllerPublishVolumeRequest)  
        returns (ControllerPublishVolumeResponse) {}  
  
    rpc ControllerUnpublishVolume (ControllerUnpublishVolumeRequest)  
        returns (ControllerUnpublishVolumeResponse) {}  
  
    rpc ValidateVolumeCapabilities (ValidateVolumeCapabilitiesRequest)  
        returns (ValidateVolumeCapabilitiesResponse) {}  
  
    rpc ListVolumes (ListVolumesRequest)  
        returns (ListVolumesResponse) {}  
  
    rpc GetCapacity (GetCapacityRequest)  
        returns (GetCapacityResponse) {}  
  
    rpc ControllerGetCapabilities (ControllerGetCapabilitiesRequest)  
        returns (ControllerGetCapabilitiesResponse) {}  
  
    rpc CreateSnapshot (CreateSnapshotRequest)  
        returns (CreateSnapshotResponse) {}  
  
    rpc DeleteSnapshot (DeleteSnapshotRequest)  
        returns (DeleteSnapshotResponse) {}  
  
    rpc ListSnapshots (ListSnapshotsRequest)  
        returns (ListSnapshotsResponse) {}  
}  
  
service Node {  
    rpc NodeStageVolume (NodeStageVolumeRequest)  
        returns (NodeStageVolumeResponse) {}  
  
    rpc NodeUnstageVolume (NodeUnstageVolumeRequest)  
        returns (NodeUnstageVolumeResponse) {}  
  
    rpc NodePublishVolume (NodePublishVolumeRequest)  
        returns (NodePublishVolumeResponse) {}  
}
```



# CSI Architecture - Plugin runs on all nodes

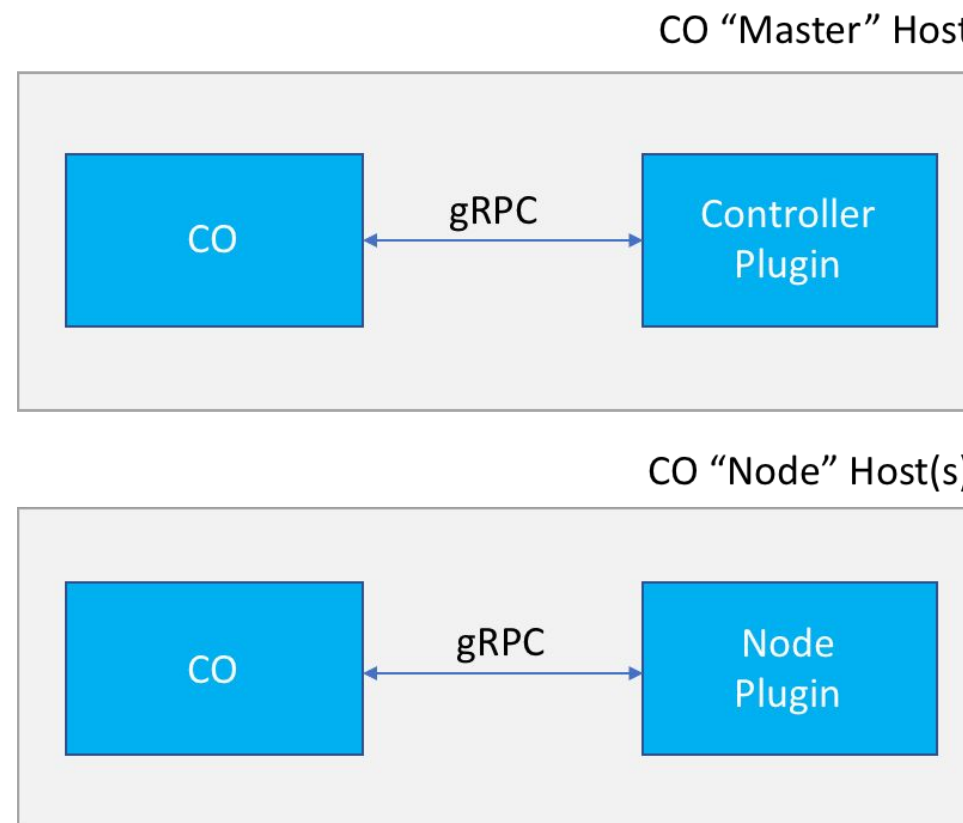


KubeCon



CloudNativeCon

China 2018



Source: <https://github.com/container-storage-interface/spec/blob/master/spec.md>

# CSI Architecture - Headless plugins



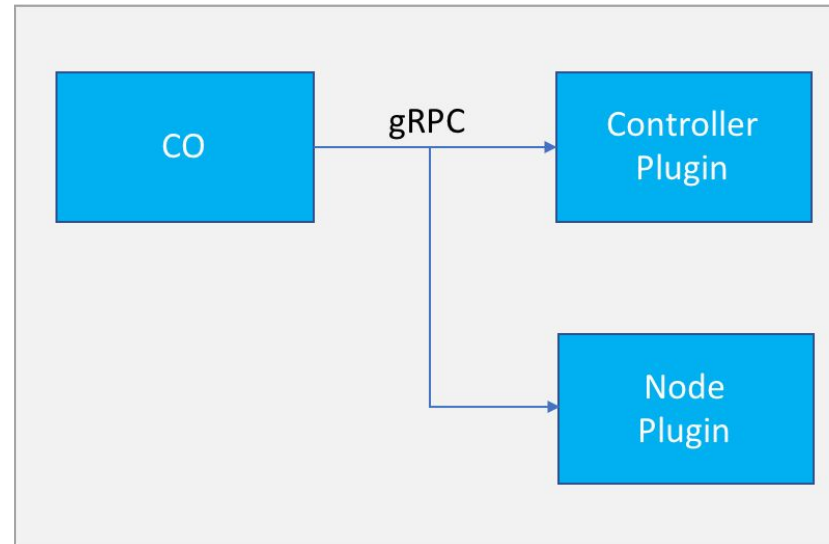
KubeCon



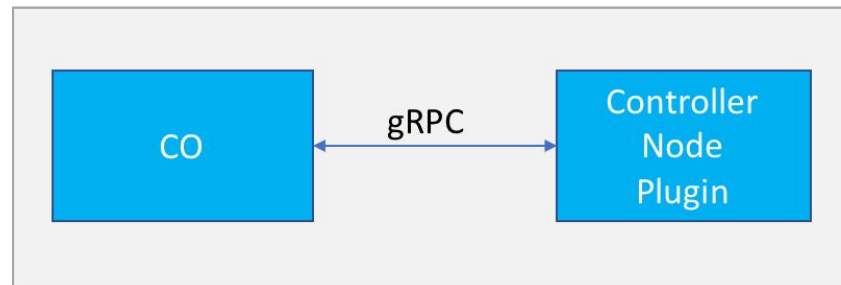
CloudNativeCon

China 2018

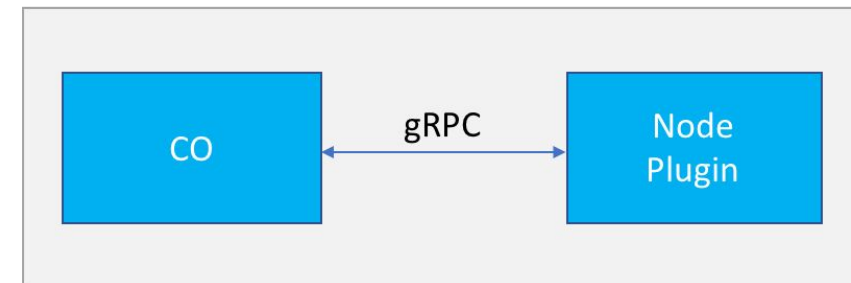
CO "Node" Hosts



CO "Node" Host(s)



CO "Node" Host(s)



# Agenda



KubeCon



CloudNativeCon

China 2018

- Motivation
  - Why snapshot in the cloud
- Background
  - Kubernetes Persistent Volumes
  - CSI Introduction
- CSI Snapshot Design
  - CSI snapshot support
  - Kubernetes Snapshot Concepts and Examples
  - CSI Snapshot Management and Deployment
- Demo

# CSI Spec - Snapshot RPCs



KubeCon



CloudNativeCon

China 2018

- ControllerGetCapabilities
- **CreateVolume**
- DeleteVolume
- ControllerPublishVolume
- ControllerUnpublishVolume
- ListVolumes
- GetCapacity
- ValidateVolumeCapabilities
- **CreateSnapshot**
- **DeleteSnapshot**
- **ListSnapshots**

## *Controller Capabilities:*

- CREATE\_DELETE\_VOLUME
- PUBLISH\_UNPUBLISH\_VOLUME
- LIST\_VOLUMES
- GET\_CAPACITY
- **CREATE\_DELETE\_SNAPSHOT**
- **LIST\_SNAPSHOTS**
- **CLONE\_VOLUME**
- PUBLISH\_READONLY



# CSI CreateSnapshot



KubeCon



CloudNativeCon

China 2018

```
message CreateSnapshotRequest {  
  string source_volume_id = 1;  
  string name = 2;  
  map<string, string> secrets = 3;  
  map<string, string> parameters = 4;  
}
```

```
message CreateSnapshotResponse {  
  Snapshot snapshot = 1;  
}  
message Snapshot {  
  int64 size_bytes = 1;  
  string snapshot_id = 2;  
  string source_volume_id = 3;  
  int64 created_at = 4;  
  boolean ready_to_use = 5;  
}
```

# CSI CreateSnapshot



KubeCon



CloudNativeCon

China 2018

```
message CreateSnapshotRequest {  
  string source_volume_id = 1;  
  string name = 2;  
  map<string, string> secrets = 3;  
  map<string, string> parameters = 4;  
}
```

## Create Snapshot

Cut Snapshot

Upload Snapshot

```
message CreateSnapshotResponse {  
  Snapshot snapshot = 1;  
}  
  
message Snapshot {  
  int64 size_bytes = 1;  
  string snapshot_id = 2;  
  string source_volume_id = 3;  
  int64 created_at = 4;  
  bool ready_to_use = 5;  
}
```

# CSI CreateSnapshot



KubeCon



CloudNativeCon

China 2018

```
message CreateSnapshotRequest {  
  string source_volume_id = 1;  
  string name = 2;  
  map<string, string> secrets = 3;  
  map<string, string> parameters = 4;  
}
```

## Create Snapshot

Cut Snapshot

Upload Snapshot

Synchronized call:  
return after snapshot is cut

```
message CreateSnapshotResponse {  
  Snapshot snapshot = 1;  
}  
  
message Snapshot {  
  int64 size_bytes = 1;  
  string snapshot_id = 2;  
  string source_volume_id = 3;  
  int64 created_at = 4;  
  bool ready_to_use = 5;  
}
```

# CSI CreateSnapshot



KubeCon



CloudNativeCon

China 2018

```
message CreateSnapshotRequest {  
  string source_volume_id = 1;  
  string name = 2;  
  map<string, string> secrets = 3;  
  map<string, string> parameters = 4;  
}
```

## Create Snapshot

Cut Snapshot

Upload Snapshot

Synchronized call:  
return after snapshot is cut

Idempotent call:  
ready\_to\_use = false

```
message CreateSnapshotResponse {  
  Snapshot snapshot = 1;  
}  
  
message Snapshot {  
  int64 size_bytes = 1;  
  string snapshot_id = 2;  
  string source_volume_id = 3;  
  int64 created_at = 4;  
  bool ready_to_use = 5;  
}
```



# CSI CreateSnapshot



KubeCon



CloudNativeCon

China 2018

```
message CreateSnapshotRequest {  
  string source_volume_id = 1;  
  string name = 2;  
  map<string, string> secrets = 3;  
  map<string, string> parameters = 4;  
}
```

## Create Snapshot

Cut Snapshot

Upload Snapshot

Synchronized call:  
return after snapshot is cut

Idempotent call:  
ready\_to\_use = false

ready\_to\_use = true

```
message CreateSnapshotResponse {  
  Snapshot snapshot = 1;  
}  
  
message Snapshot {  
  int64 size_bytes = 1;  
  string snapshot_id = 2;  
  string source_volume_id = 3;  
  int64 created_at = 4;  
  bool ready_to_use = 5;  
}
```

# CSI CreateVolumeRequest



KubeCon



CloudNativeCon

China 2018

```
message CreateVolumeRequest {  
  string name = 1;  
  CapacityRange capacity_range = 2;  
  repeated VolumeCapability volume_capabilities = 3;  
  map<string, string> parameters = 4;  
  map<string, string> secrets = 5;  
  
  VolumeContentSource volume_content_source = 6;  
  
  TopologyRequirement accessibility_requirements = 7;  
}
```

```
message VolumeContentSource {  
  message SnapshotSource {  
    string id = 1;  
  }  
  oneof type {  
    SnapshotSource snapshot = 1;  
    VolumeSource volume = 2;  
  }  
}
```

**VolumeContentSource:** *to create a volume from a snapshot source or volume source.*

# Kubernetes Persistent Volumes



KubeCon



CloudNativeCon

China 2018

## Users

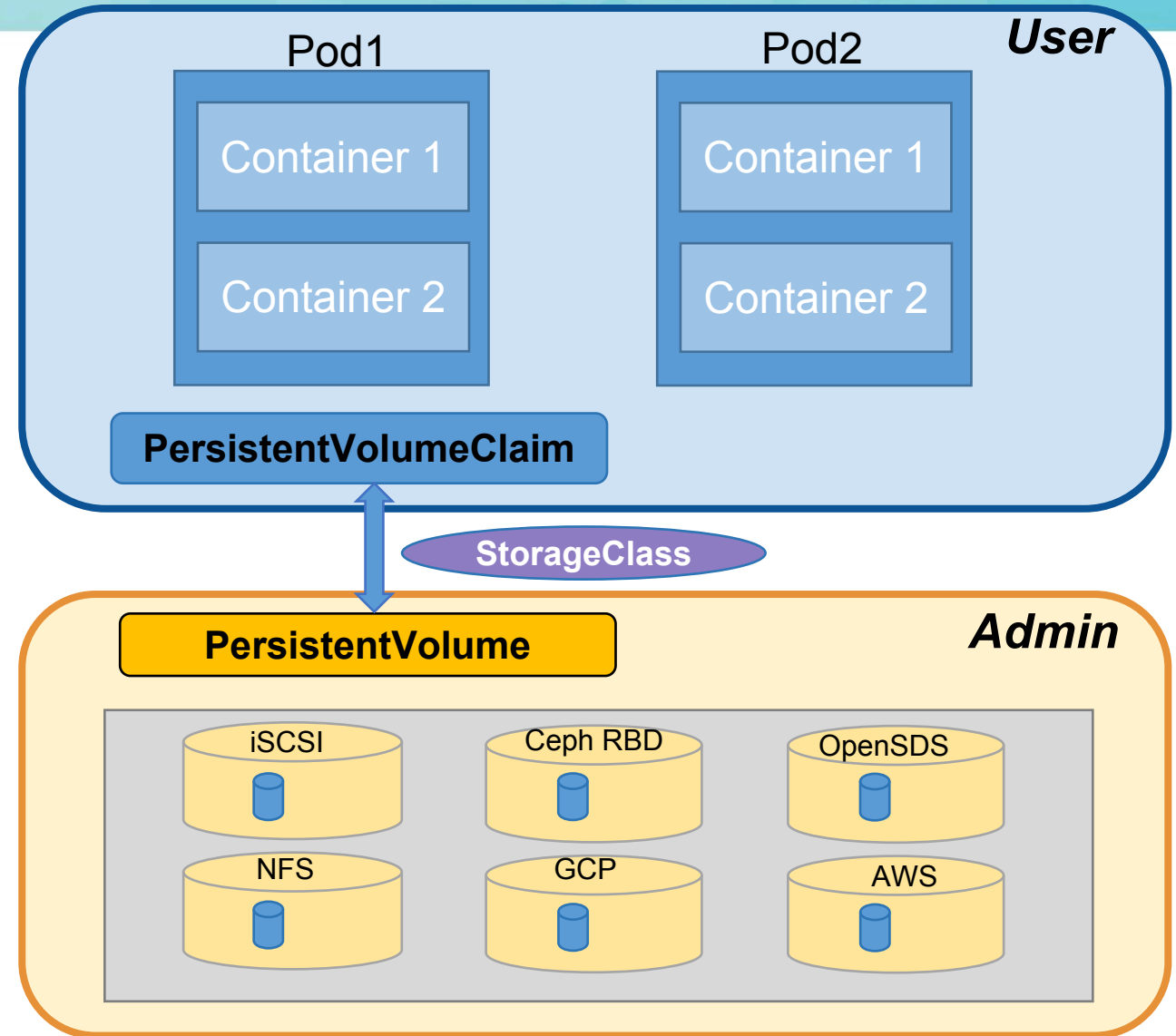
- Requires storage
- PersistentVolumeClaim (PVC)

## Admins

- Allocates storage
- PersistentVolume (PV)

## Dynamic Provisioning

- StorageClass



# Kubernetes Volume Snapshots



KubeCon



CloudNativeCon

China 2018

## Users

- VolumeSnapshot

## Admins

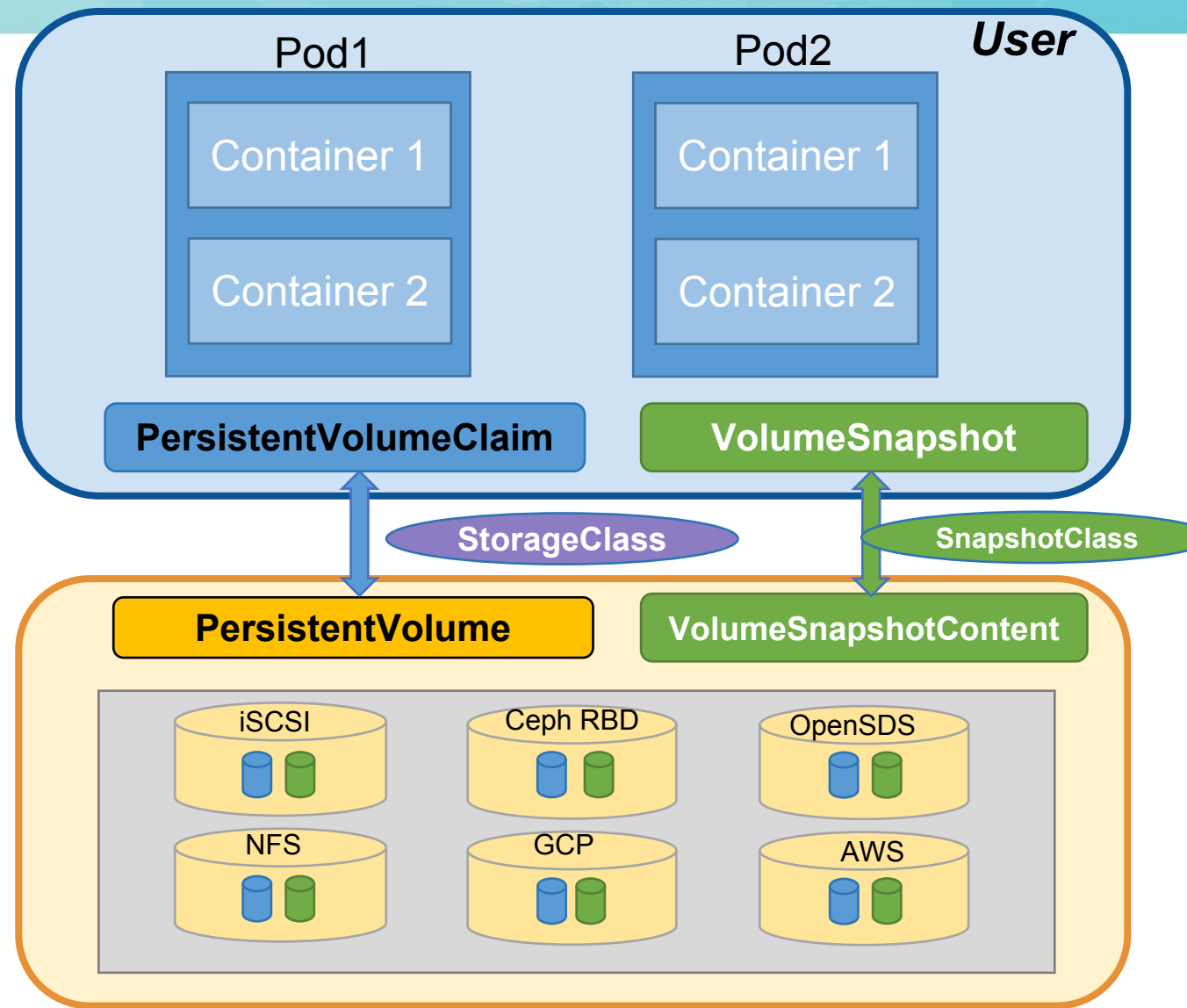
- VolumeSnapshotContent

## Provisioning

- VolumeSnapshotClass

All are CRD objects

-- make kubernetes plugabal and extensible



# VolumeSnapshot API

```
type VolumeSnapshot struct {  
    metav1.TypeMeta `json:",inline"`  
    metav1.ObjectMeta `json:"metadata"`  
    Spec VolumeSnapshotSpec  
    Status VolumeSnapshotStatus  
}
```

# VolumeSnapshot API

```
type VolumeSnapshot struct {  
    metav1.TypeMeta `json:",inline"`  
    metav1.ObjectMeta `json:"metadata"`  
    Spec VolumeSnapshotSpec  
    Status VolumeSnapshotStatus  
}
```

```
type VolumeSnapshotSpec struct {  
    Source *core_v1.TypedLocalObjectReference  
    SnapshotContentName string  
    VolumeSnapshotClassName *string  
}
```

```
type VolumeSnapshotStatus struct {  
    CreationTime *metav1.Time  
    RestoreSize *resource.Quantity  
    Ready bool  
    Error *storage.VolumeError  
}
```



## VolumeSnapshot API

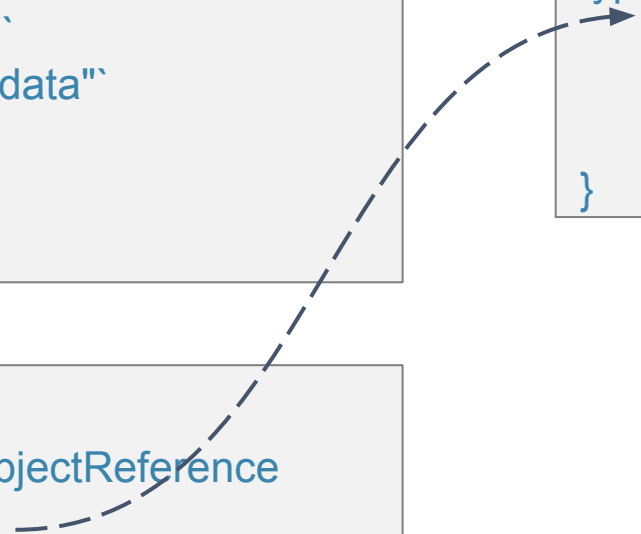
```
type VolumeSnapshot struct {  
    metav1.TypeMeta `json:",inline"`  
    metav1.ObjectMeta `json:"metadata"`  
    Spec VolumeSnapshotSpec  
    Status VolumeSnapshotStatus  
}
```

```
type VolumeSnapshotSpec struct {  
    Source *core_v1.TypedLocalObjectReference  
    SnapshotContentName string  
    VolumeSnapshotClassName *string  
}
```

```
type VolumeSnapshotStatus struct {  
    CreationTime *metav1.Time  
    RestoreSize *resource.Quantity  
    Ready bool  
    Error *storage.VolumeError  
}
```

## VolumeSnapshotContent API

```
type VolumeSnapshotContent struct {  
    metav1.TypeMeta `json:",inline"`  
    metav1.ObjectMeta `json:"metadata"`  
    Spec VolumeSnapshotContentSpec  
}
```



## VolumeSnapshot API

```
type VolumeSnapshot struct {  
    metav1.TypeMeta `json:",inline"`  
    metav1.ObjectMeta `json:"metadata"`  
    Spec VolumeSnapshotSpec  
    Status VolumeSnapshotStatus  
}
```

```
type VolumeSnapshotSpec struct {  
    Source *core_v1.TypedLocalObjectReference  
    SnapshotContentName string  
    VolumeSnapshotClassName *string  
}
```

```
type VolumeSnapshotStatus struct {  
    CreationTime *metav1.Time  
    RestoreSize *resource.Quantity  
    Ready bool  
    Error *storage.VolumeError  
}
```

## VolumeSnapshotContent API

```
type VolumeSnapshotContent struct {  
    metav1.TypeMeta `json:",inline"`  
    metav1.ObjectMeta `json:"metadata"`  
    Spec VolumeSnapshotContentSpec  
}
```

```
type VolumeSnapshotContentSpec struct {  
    VolumeSnapshotSource  
    VolumeSnapshotRef *core_v1.ObjectReference  
    PersistentVolumeRef *core_v1.ObjectReference  
    VolumeSnapshotClassName *string  
}
```

```
type VolumeSnapshotSource struct {  
    CSI *CSIVolumeSnapshotSource  
}
```

```
type CSIVolumeSnapshotSource struct {  
    Driver string `json:"driver"`  
    SnapshotHandle string  
    CreationTime *int64  
    RestoreSize *int64  
}
```

# Restore Volume: Data Source



KubeCon



CloudNativeCon

China 2018

- Used for volume snapshotting and restoring volume from snapshot.
- Can be used to support other types of data sources such as PVC in the future.

```
type PersistentVolumeClaimSpec struct {  
    AccessModes []PersistentVolumeAccessMode  
    Selector *metav1.LabelSelector  
    Resources ResourceRequirements  
    VolumeName string  
    StorageClassName *string  
    VolumeMode *PersistentVolumeMode  
    DataSource *TypedLocalObjectReference  
}
```

# Volume Snapshot Example



KubeCon



CloudNativeCon

China 2018

## VolumeSnapshotClass Example 1

```
apiVersion: snapshot.storage.k8s.io/v1alpha1
kind: VolumeSnapshotClass
metadata:
  name: default-snapclass
  annotations:
    snapshot.storage.kubernetes.io/is-default-class: "true"
snapshotter: com.example.csi-driver
```

## VolumeSnapshot Example

```
apiVersion: snapshot.storage.k8s.io/v1alpha1
kind: VolumeSnapshot
metadata:
  name: new-snapshot-demo
  namespace: demo-namespace
spec:
  snapshotClassName: csi-snapclass
  source:
    name: mypvc
    kind: PersistentVolumeClaim
```

## VolumeSnapshotClass Example 2

```
apiVersion: snapshot.storage.k8s.io/v1alpha1
kind: VolumeSnapshotClass
metadata:
  name: csi-snapclass
snapshotter: com.example.csi-driver
parameters:
  fakeSnapshotOption: foo
  csiSnapshotterSecretName: csi-secret
  csiSnapshotterSecretNamespace: csi-namespace
```

# Volume Snapshot Example



KubeCon



CloudNativeCon

China 2018

## VolumeSnapshotClass Example

```
apiVersion: snapshot.storage.k8s.io/v1alpha1
kind: VolumeSnapshotClass
metadata:
  name: default-snapclass
  annotations:
    snapshot.storage.kubernetes.io/is-default-class: "true"
snapshotter: com.example.csi-driver
```

\$kubect! get pvc

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS
mypvc	Bound	pvc-8dbe4163	6Gi	RWO	csi-example

## VolumeSnapshot Example

```
apiVersion: snapshot.storage.k8s.io/v1alpha1
kind: VolumeSnapshot
metadata:
  name: new-snapshot-demo
  namespace: demo-namespace
spec:
  snapshotClassName: csi-snapclass
  source:
    name: mypvc
    kind: PersistentVolumeClaim
```

*Default class  
is used if not  
specified*

# Volume Snapshot Example



KubeCon



CloudNativeCon

China 2018

## VolumeSnapshotClass Example

```
apiVersion: snapshot.storage.k8s.io/v1alpha1
kind: VolumeSnapshotClass
metadata:
  name: default-snapclass
  annotations:
    snapshot.storage.kubernetes.io/is-default-class: "true"
snapshotter: com.example.csi-driver
```

## VolumeSnapshot Example

```
apiVersion: snapshot.storage.k8s.io/v1alpha1
kind: VolumeSnapshot
metadata:
  name: new-snapshot-demo
  namespace: demo-namespace
spec:
  snapshotClassName: csi-snapclass
  source:
    name: mypvc
    kind: PersistentVolumeClaim
```

*Default class  
is used if not  
specified*

## \$kubect! get pvc

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS
mypvc	Bound	pvc-8dbe4163	6Gi	RWO	csi-example

## \$kubect! create -f snapshot.yaml

```
volumesnapshot.snapshot.storage.k8s.io/new-snapshot-demo
created
```

```
Name:      demo-snapshot-podpvc
Namespace:  default
API Version: snapshot.storage.k8s.io/v1alpha1
Kind:      VolumeSnapshot
...
Spec:
  Snapshot Class Name:  csi-snapclass
  Snapshot Content Name: snapcontent-a915b87c
  Source:
    Kind: PersistentVolumeClaim
    Name: podpvc
Status:
  Creation Time: 2018-11-08T21:53:01Z
  Ready:      true
  Restore Size: 6Gi
```



# Restore Volume from Snapshot



KubeCon



CloudNativeCon

China 2018

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: pvc-restore
  Namespace: demo-namespace
spec:
  storageClassName: csi-storageclass
  dataSource:
    name: new-snapshot-demo
    kind: VolumeSnapshot
    apiGroup: snapshot.storage.k8s.io
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
```

# Agenda



KubeCon



CloudNativeCon

China 2018

- Motivation
  - Why snapshot in the cloud
- Background
  - Kubernetes Persistent Volumes and storage plugins
  - CSI Introduction
- CSI Snapshot Design
  - CSI snapshot support
  - Kubernetes Snapshot Concepts and Examples
  - CSI Snapshot Management and Deployment
- Demo

# CSI Volume Management



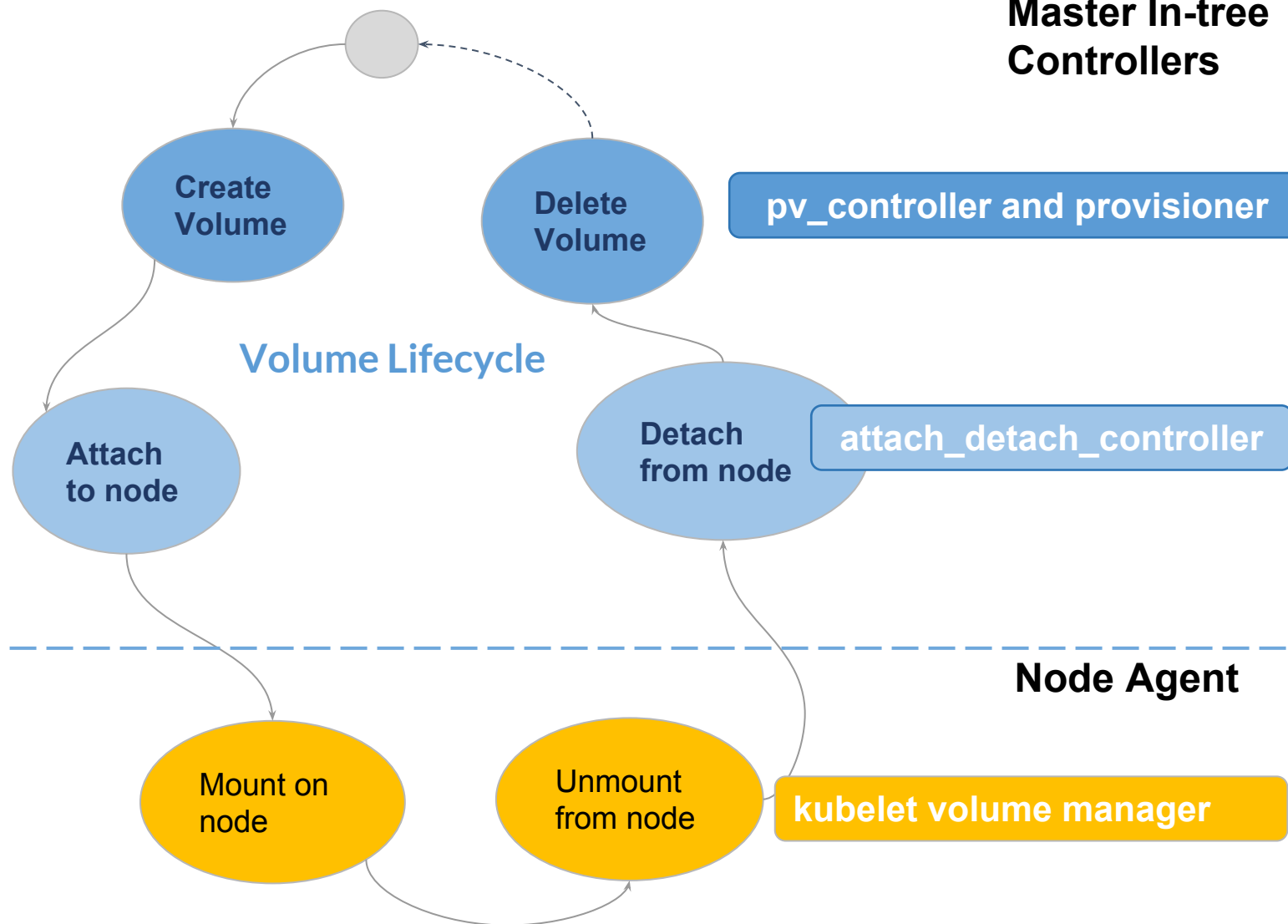
KubeCon



CloudNativeCon

China 2018

**Master In-tree  
Controllers**



# CSI Volume Management



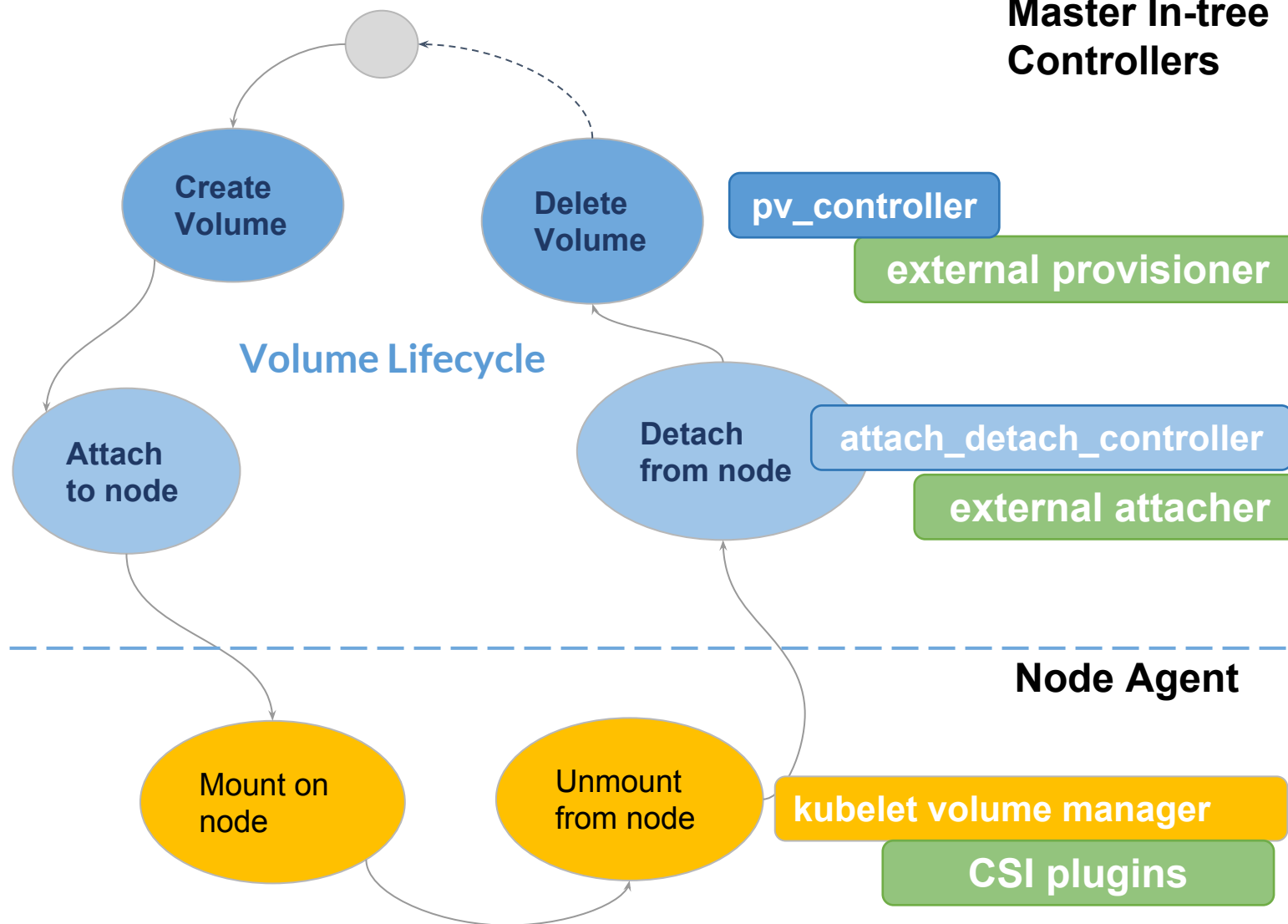
KubeCon



CloudNativeCon

China 2018

**Master In-tree  
Controllers**



# CSI Snapshot Management



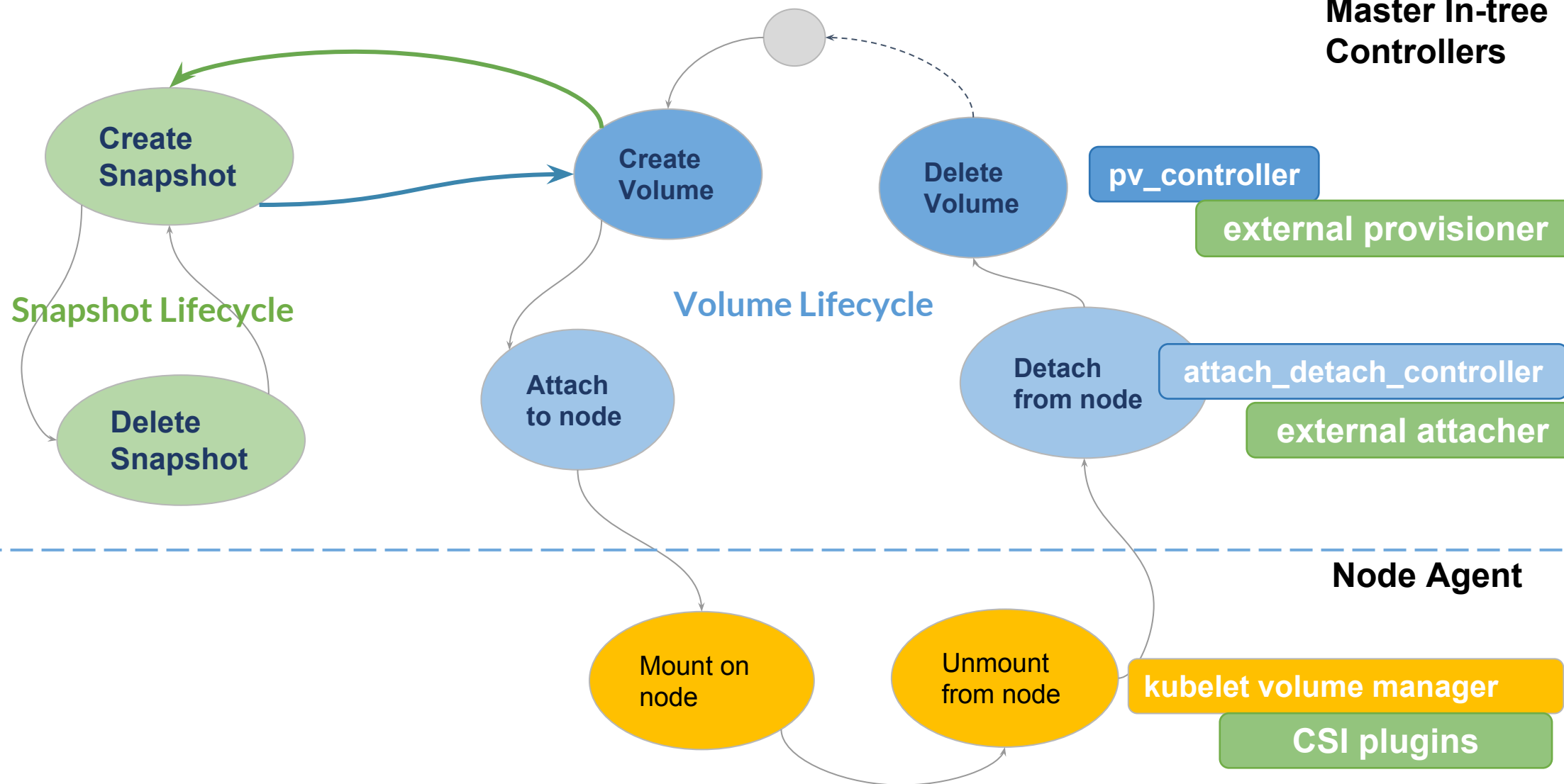
KubeCon



CloudNativeCon

China 2018

**Master In-tree  
Controllers**



# CSI Snapshot Management



KubeCon

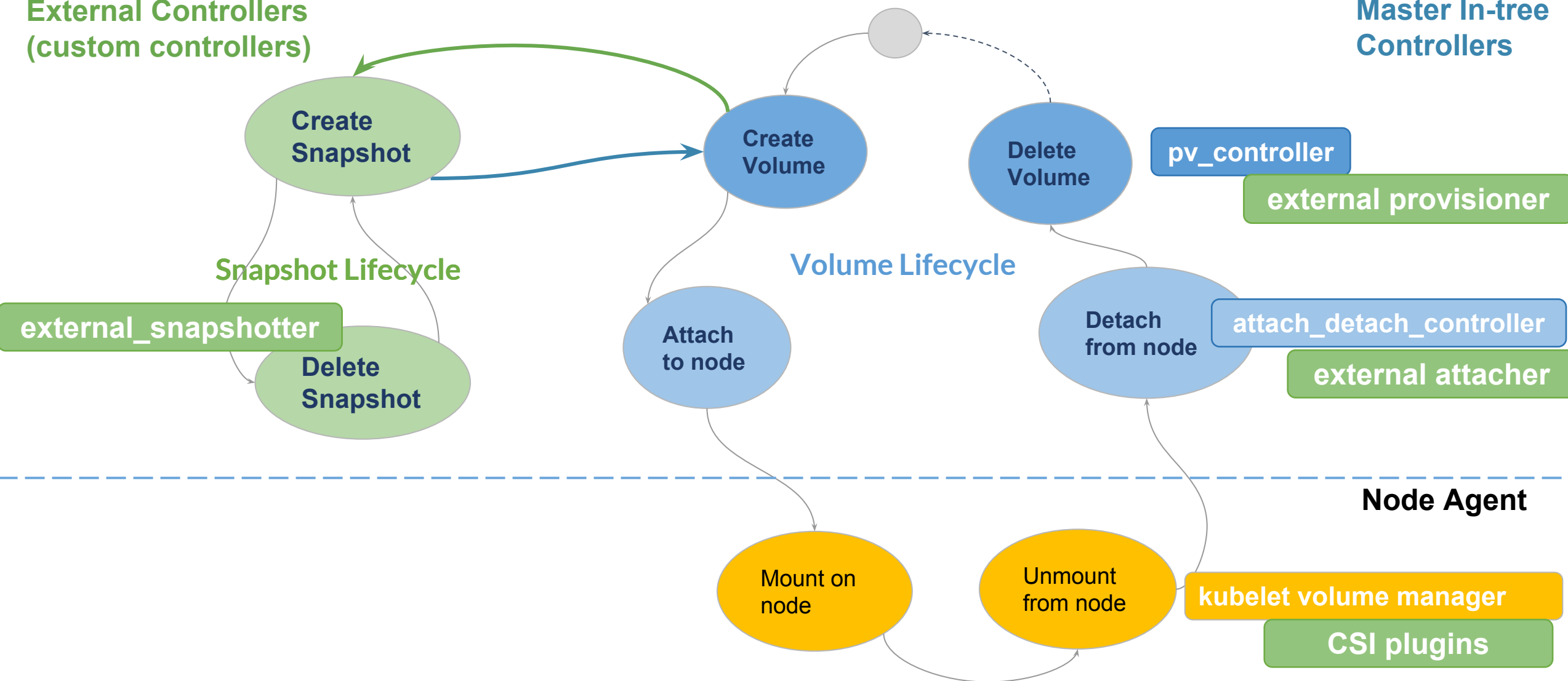


CloudNativeCon

China 2018

**External Controllers  
(custom controllers)**

**Master In-tree  
Controllers**





# CSI Snapshot Management



KubeCon

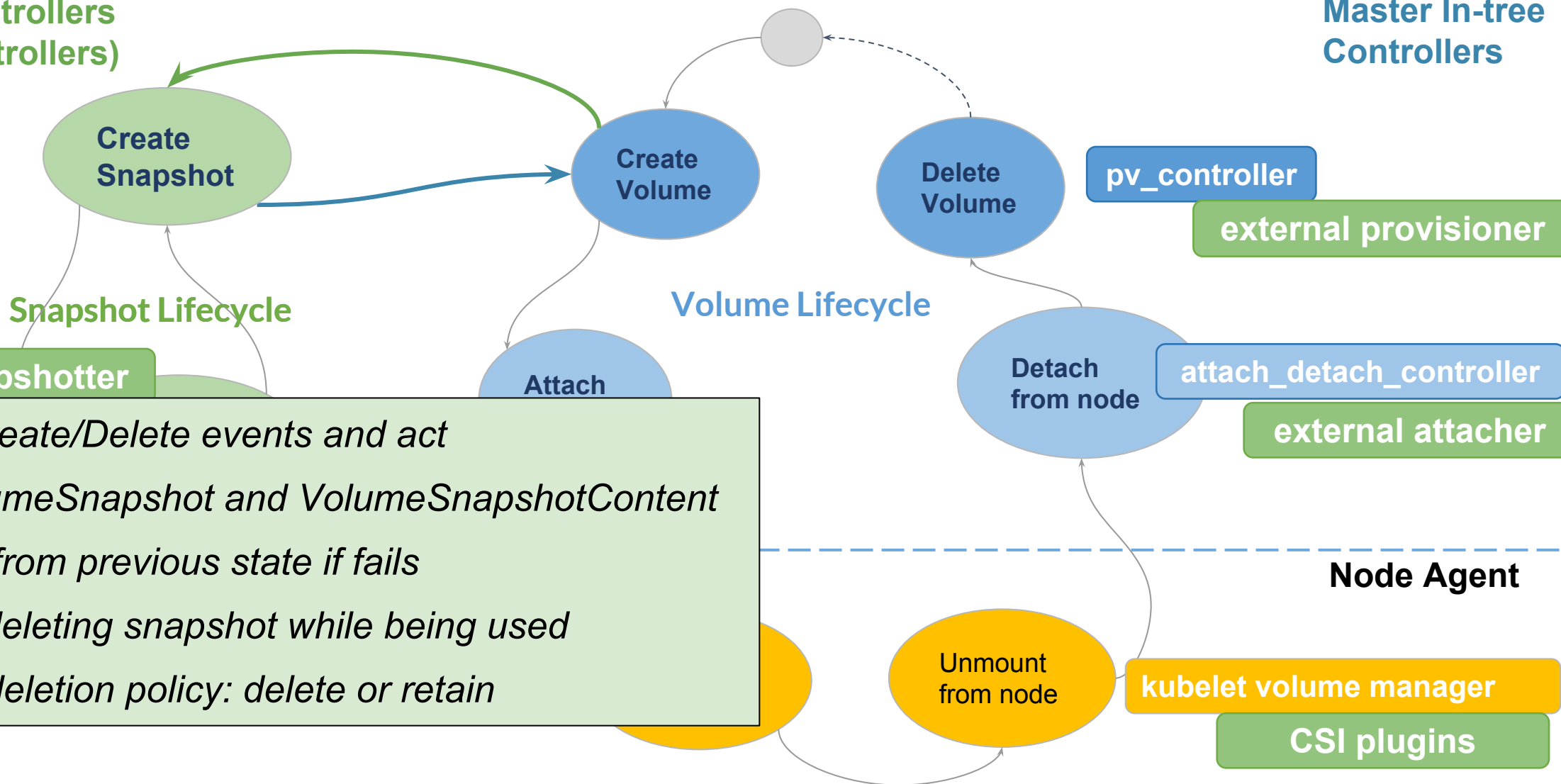


CloudNativeCon

China 2018

**External Controllers  
(custom controllers)**

**Master In-tree  
Controllers**



# CSI Snapshot Deployment

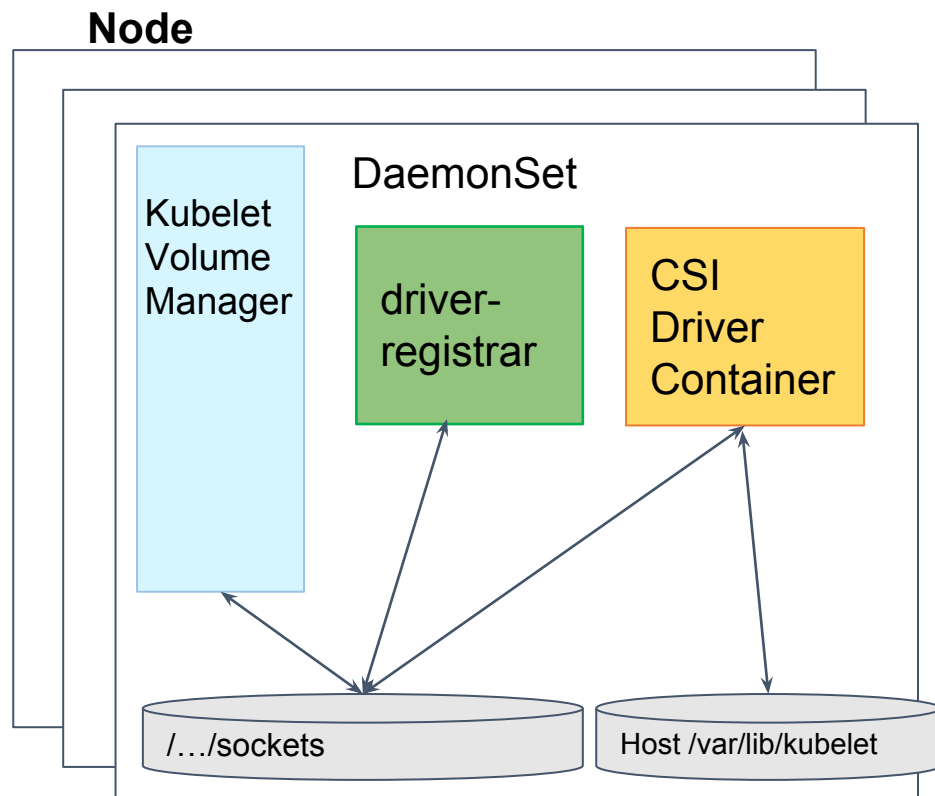


KubeCon

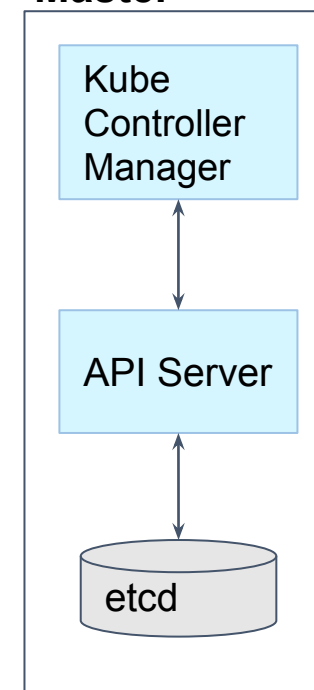


CloudNativeCon

China 2018



## Master



# CSI Snapshot Deployment

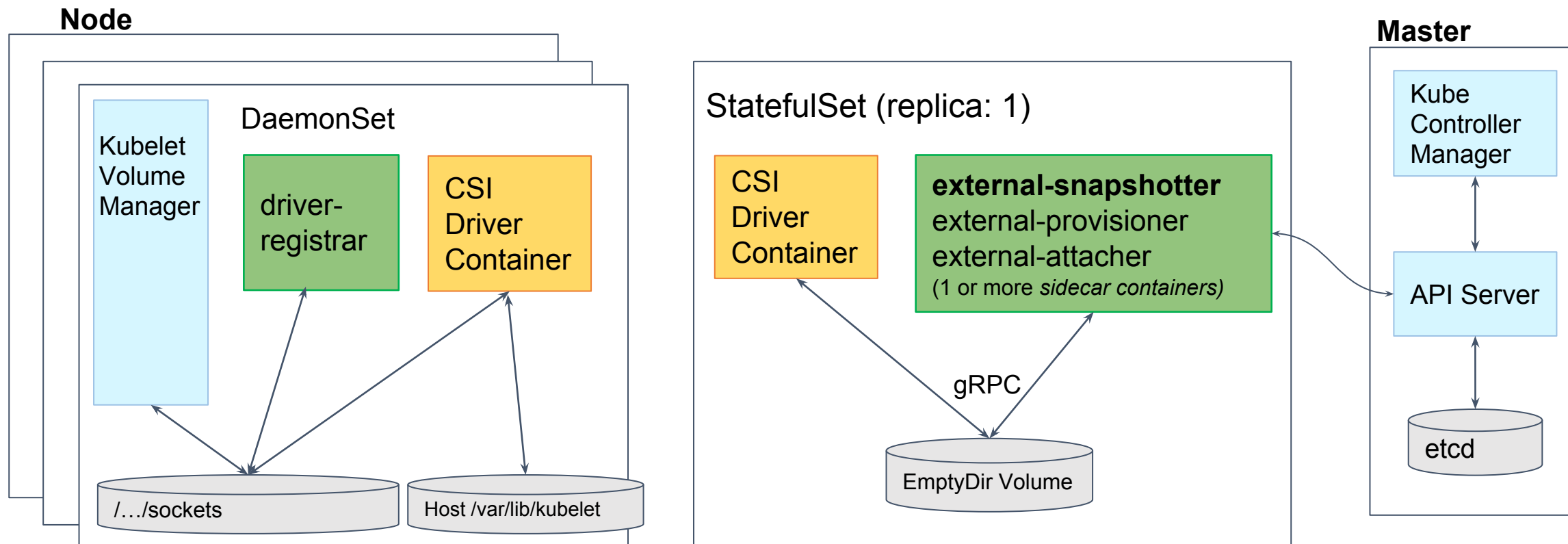


KubeCon



CloudNativeCon

China 2018



# CSI Drivers with Snapshot Support



KubeCon



CloudNativeCon

China 2018

## Alpha feature in Kubernetes v1.12

- [GCE Persistent Disk CSI Driver](#)
- [OpenSDS CSI Driver](#)
- [Ceph RBD CSI Driver](#)
- [Portworx CSI Driver](#)
- [Gluster CSI Driver](#)
- Support for other [drivers](#) is pending

### Production Drivers

Name	Status	More Information
Cinder	v0.2.0	A Container Storage Interface (CSI) Storage Plug-in for Cinder
DigitalOcean Block Storage	v0.0.1 (alpha)	A Container Storage Interface (CSI) Driver for DigitalOcean Block Storage
AWS Elastic Block Storage	v0.0.1(alpha)	A Container Storage Interface (CSI) Driver for AWS Elastic Block Storage (EBS)
GCE Persistent Disk	Alpha	A Container Storage Interface (CSI) Storage Plugin for Google Compute Engine Persistent Disk
OpenSDS	Beta	For more information, please visit <a href="#">releases</a> and <a href="https://github.com/opensds/nbp/tree/master/csi">https://github.com/opensds/nbp/tree/master/csi</a>
Portworx	0.3.0	CSI implementation is available <a href="#">here</a> which can be used as an example also.
RBD	v0.2.0	A Container Storage Interface (CSI) Storage RBD Plug-in for Ceph
CephFS	v0.2.0	A Container Storage Interface (CSI) Storage Plug-in for CephFS
ScaleIO	v0.1.0	A Container Storage Interface (CSI) Storage Plugin for DellEMC ScaleIO
vSphere	v0.1.0	A Container Storage Interface (CSI) Storage Plug-in for VMware vSphere
NetApp	v0.2.0 (alpha)	A Container Storage Interface (CSI) Storage Plug-in for NetApp's <a href="#">Trident</a> container storage orchestrator
Ember CSI	v0.2.0 (alpha)	Multi-vendor CSI plugin supporting over 80 storage drivers to provide block and mount storage to Container Orchestration systems.
Nutanix	beta	A Container Storage Interface (CSI) Storage Driver for Nutanix
Quobyte	v0.2.0	A Container Storage Interface (CSI) Plugin for Quobyte
GlusterFS	Beta	A Container Storage Interface (CSI) Plugin for GlusterFS

# Support Snapshot in CSI Plugin



KubeCon



CloudNativeCon

China 2018

- CSI driver needs to add support for controller capabilities
  - CREATE\_DELETE\_SNAPSHOT
  - LIST\_SNAPSHOTS
- Implement controller RPCs
  - CreateSnapshot
  - DeleteSnapshot
  - ListSnapshots

# What's Next



KubeCon



CloudNativeCon

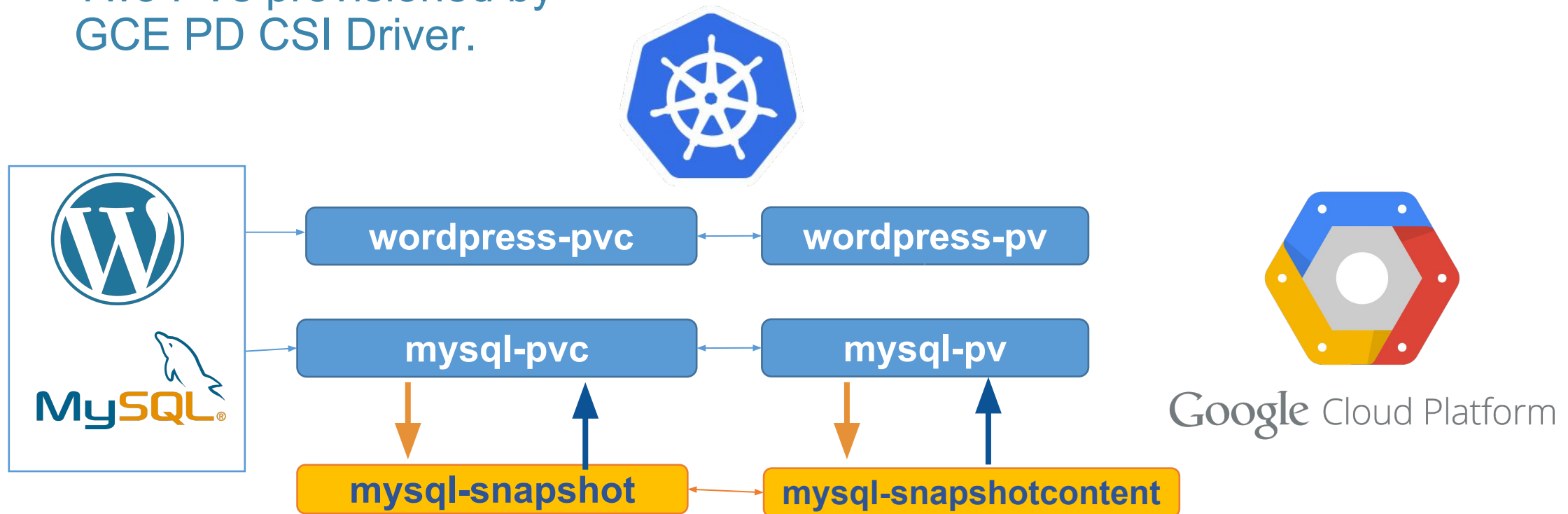
China 2018

- Topology support
- Pre-snapshot and Post-snapshot Hooks
- Application consistent group snapshot
- Policy-based management
- Revert snapshot to existing volume
- In-place restore of PVC
- Clone volume
- .....

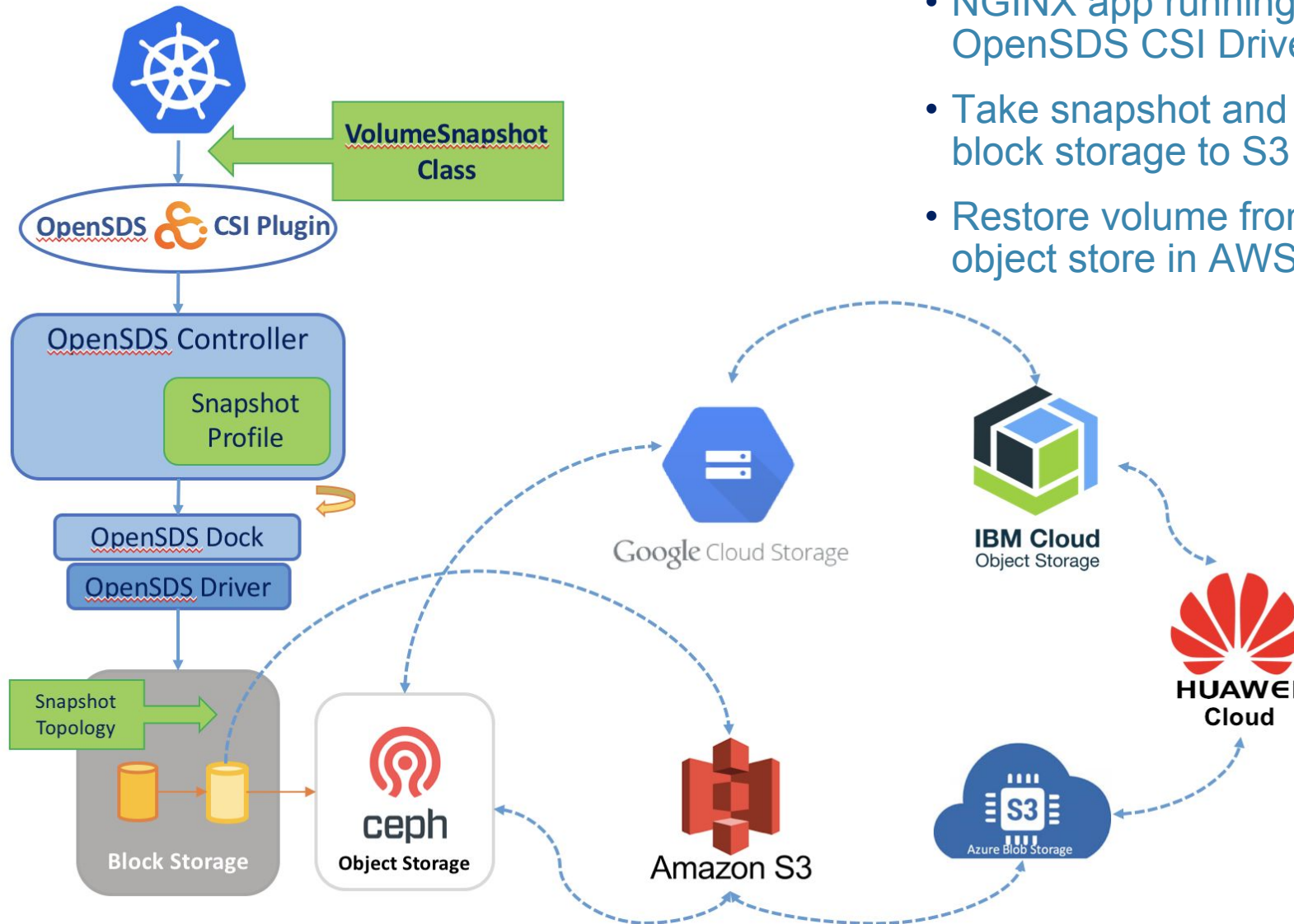


# Demo 1 – Wordpress Failure Recovery

- Wordpress and MySQL
  - Running on a multi-zone cluster
  - Two PVs provisioned by GCE PD CSI Driver.
- Take snapshot of volume
- Restore volume from snapshot stored on Google Cloud storage



# Demo 2 – Upload Snapshot



- NGINX app running with PV provisioned by OpenSDS CSI Driver with LVM backend.
- Take snapshot and upload snapshot from local LVM block storage to S3 object store in AWS.
- Restore volume from VolumeSnapshot stored in S3 object store in AWS.

# GETTING INVOLVED



KubeCon



CloudNativeCon

China 2018



<https://kubernetes.io/blog/2018/10/09/introducing-volume-snapshot-alpha-for-kubernetes/>



<https://github.com/kubernetes-csi/external-snapshotter>



<https://docs.google.com/document/d/1qdfvAj5O-tTAZzqJyz3B-yczLLxOiQd-XKpJmTEMazs/>



- <https://kubernetes.io/docs/concepts/storage/volume-snapshots/>
- <https://kubernetes-csi.github.io/docs/Implement-Snapshot-Feature.html>

Github: *jingxu97 & xing-yang*

Slack: *jinxu & xyang*

Email: [jinxu@google.com](mailto:jinxu@google.com) & [xingyang105@gmail.com](mailto:xingyang105@gmail.com)

CONTACT  
US!





**KubeCon**



**CloudNativeCon**

China 2018

