

MEC Kubernetes Platform & Storage

Byungsu Park

Edge Service Tech. Dev. Team, SK Telecom, Korea

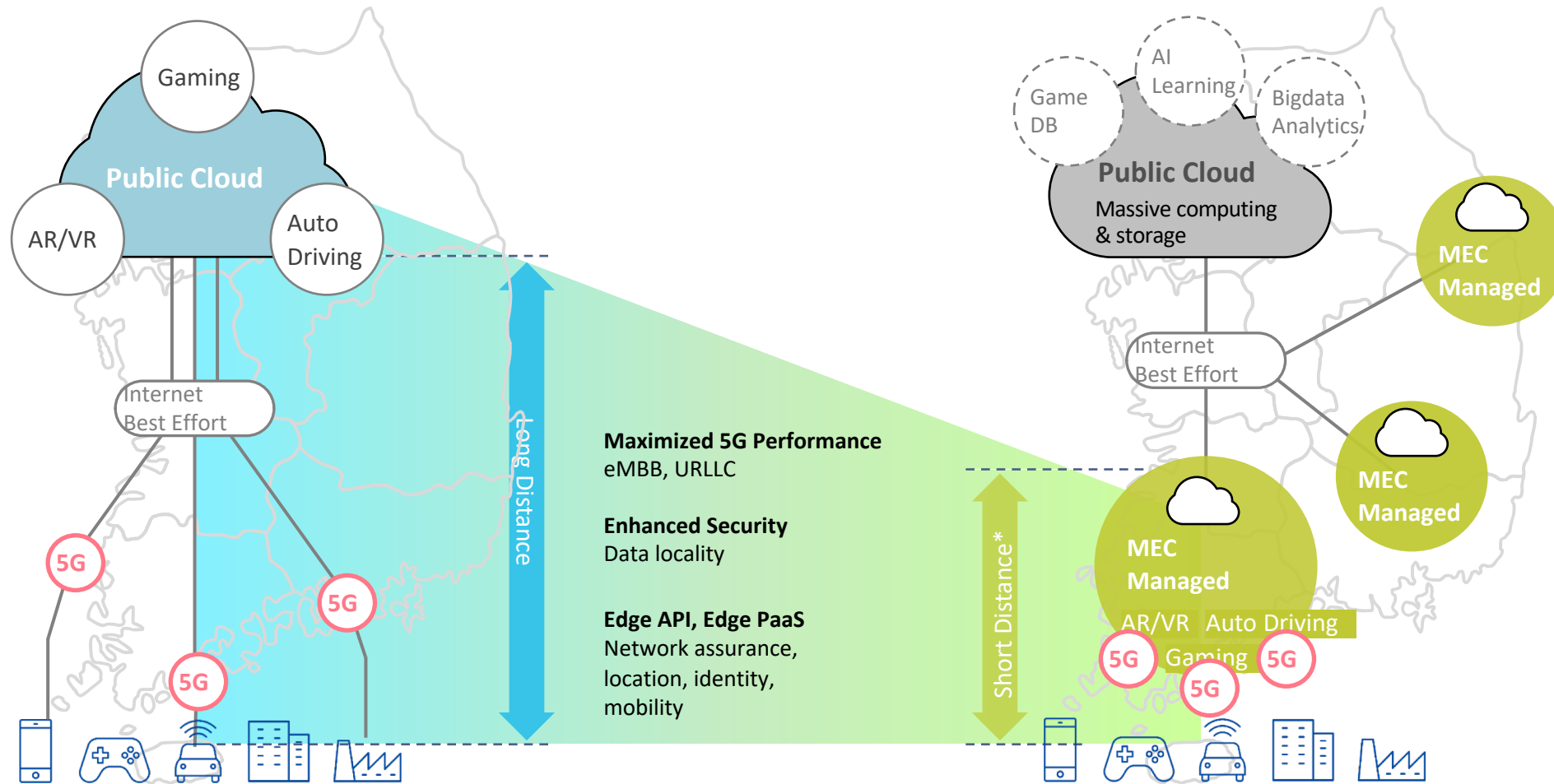
Outline

- **SKT's 5G MEC Overview**
- **Current Challenges in Telco**
- **KubeVirt Overview**
- **KubeVirt Volumes & Container-Native Storage**

SKT's 5G MEC Overview

SKT's 5G Mobile Edge Computing Concept

Cloud computing environment nearest to 5G consumers, very reliable for ultra-broadband, real-time, and privacy-sensitive services



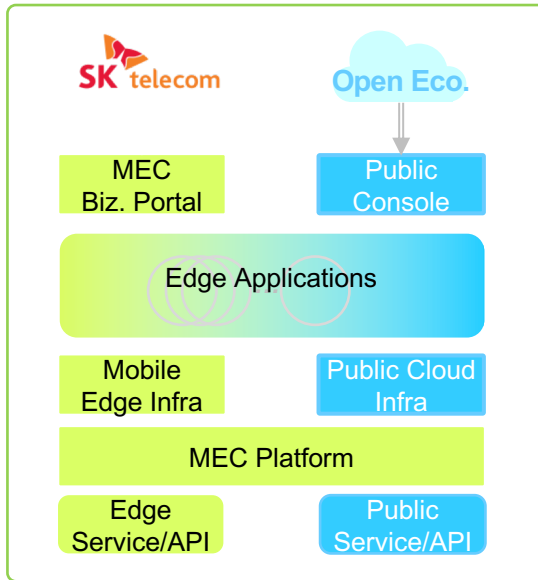
* Round trip latency <= (theoretical value) 10ms

SKT's 5G MEC Deployment Model

SKT 5G MEC supports both distributed edge and on-site edge models

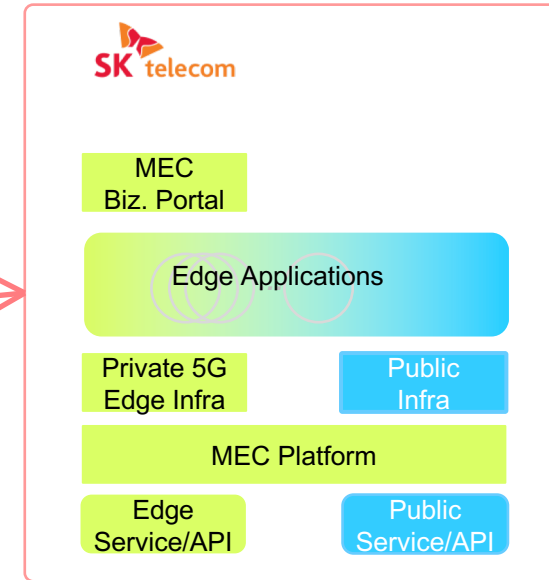
Distributed Edge

*"Edge cloud business,
Nation-wide service"*



On-Site Edge

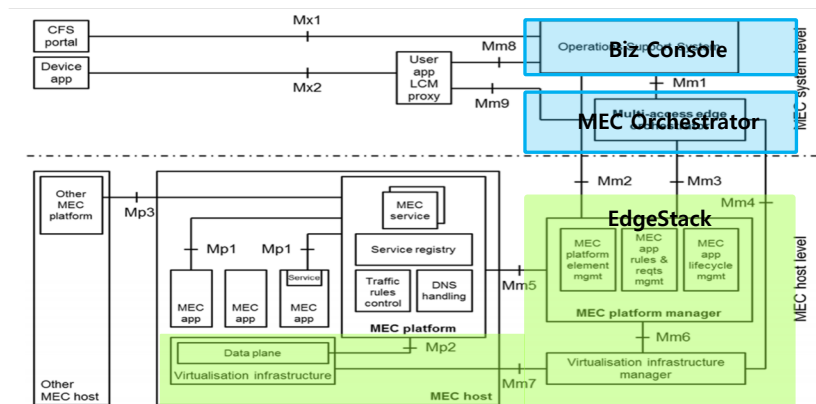
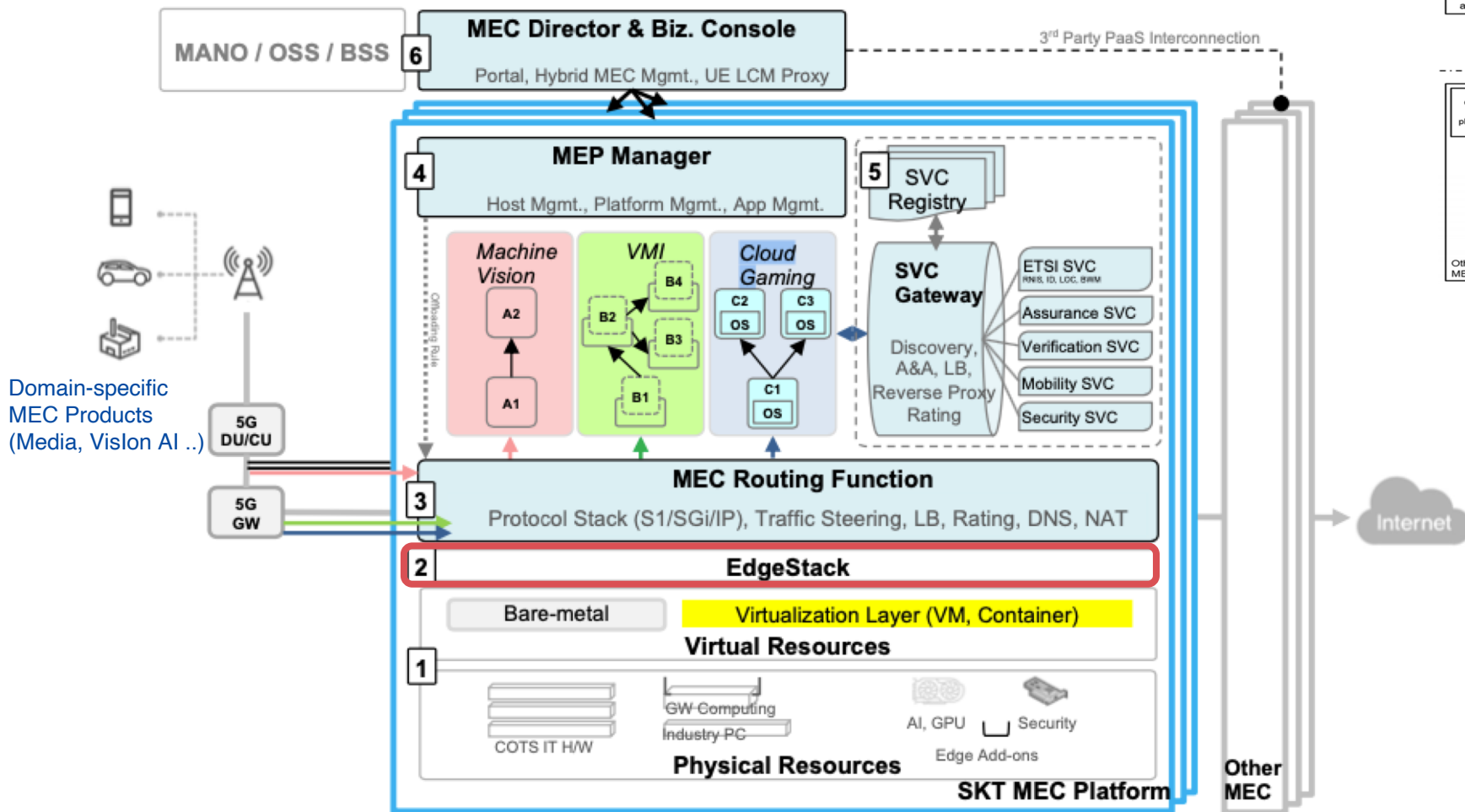
*"Low latency, security sensitive
services at customer site"*



Legend:
● 5G Offices ◆ On-Site

SKT's 5G MEC Technology

5G MEC Software Architecture

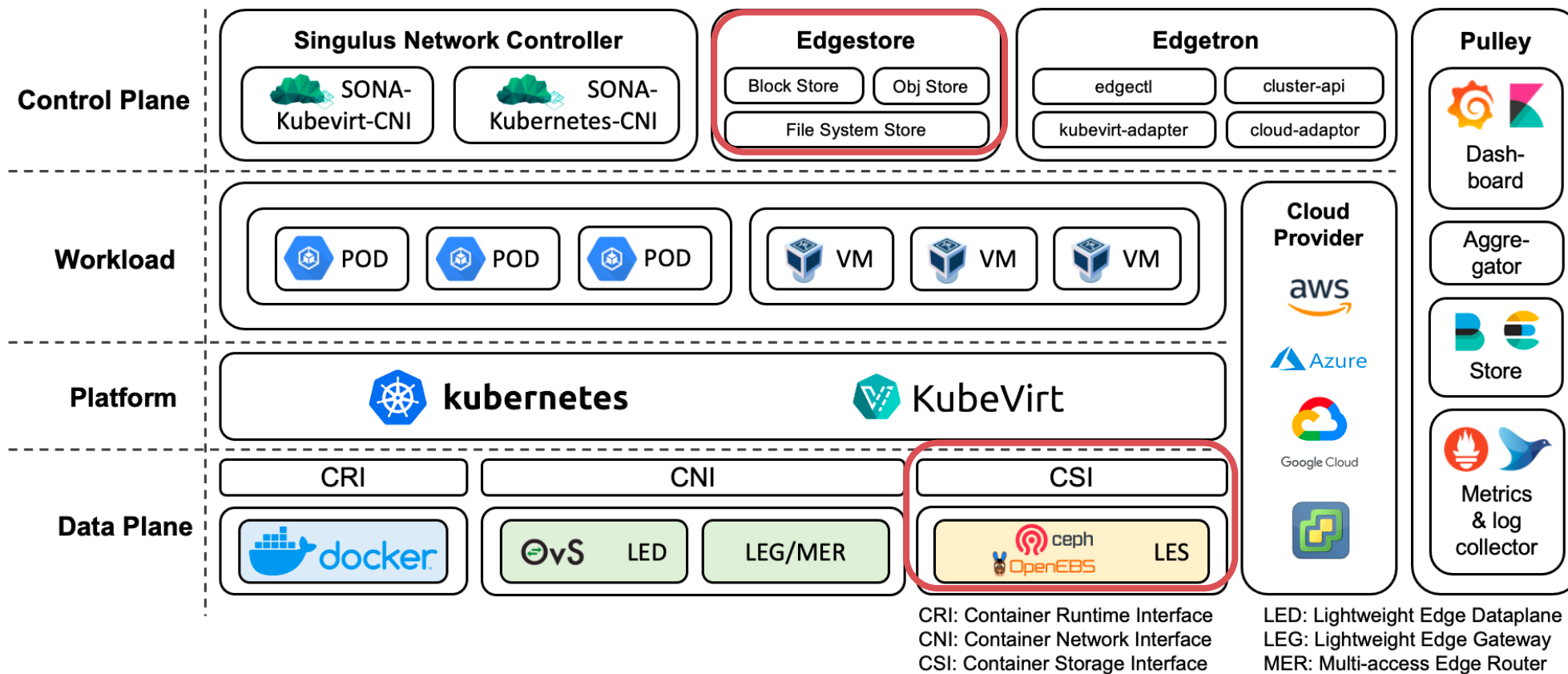


ETSI MEC Reference Architecture

High Performance Edge Cloud Solution

EdgeStack provides lightweight but high performance edge specific cloud solution

EdgeStack Architecture



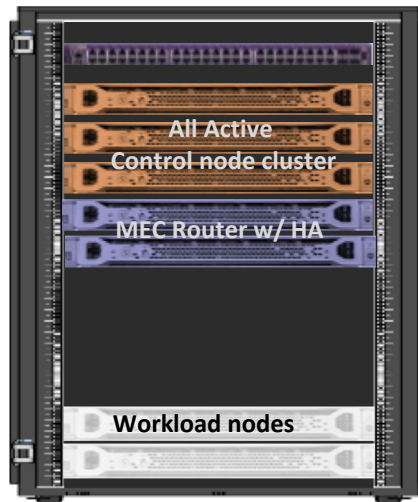
Deployment Option and Form Factors

Single node



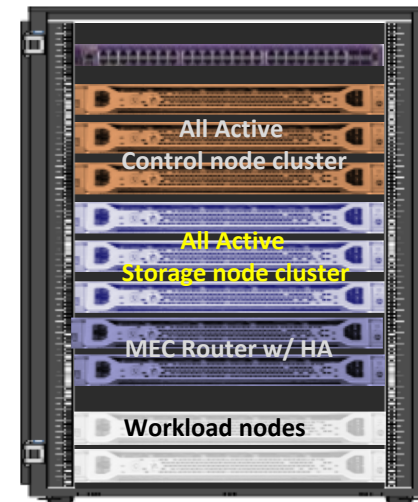
[Starter]

Three nodes



[Standard]

Six nodes



[Enterprise]

Scale-out Storage

High Availability

Storage

Usage

VM level

Small

PoC, Test-bed

Node level

Large

SMB Commercial

Node level

Customizable

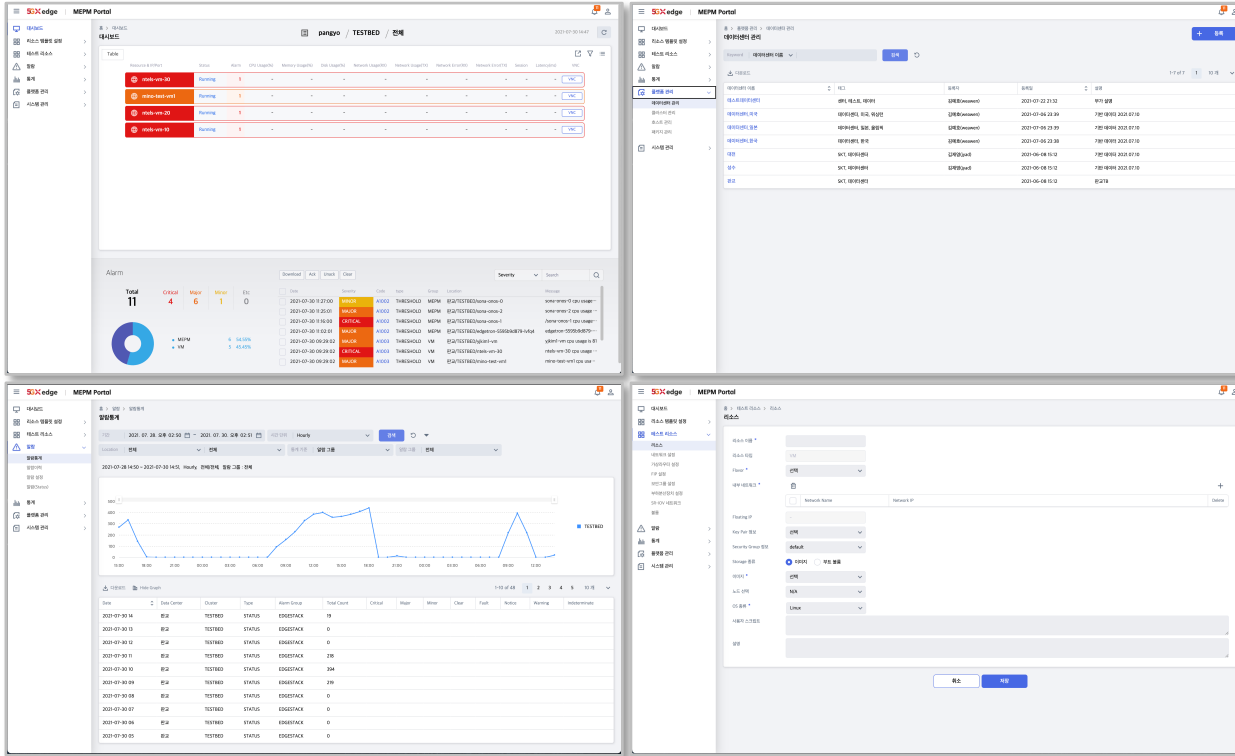
Large Enterprise



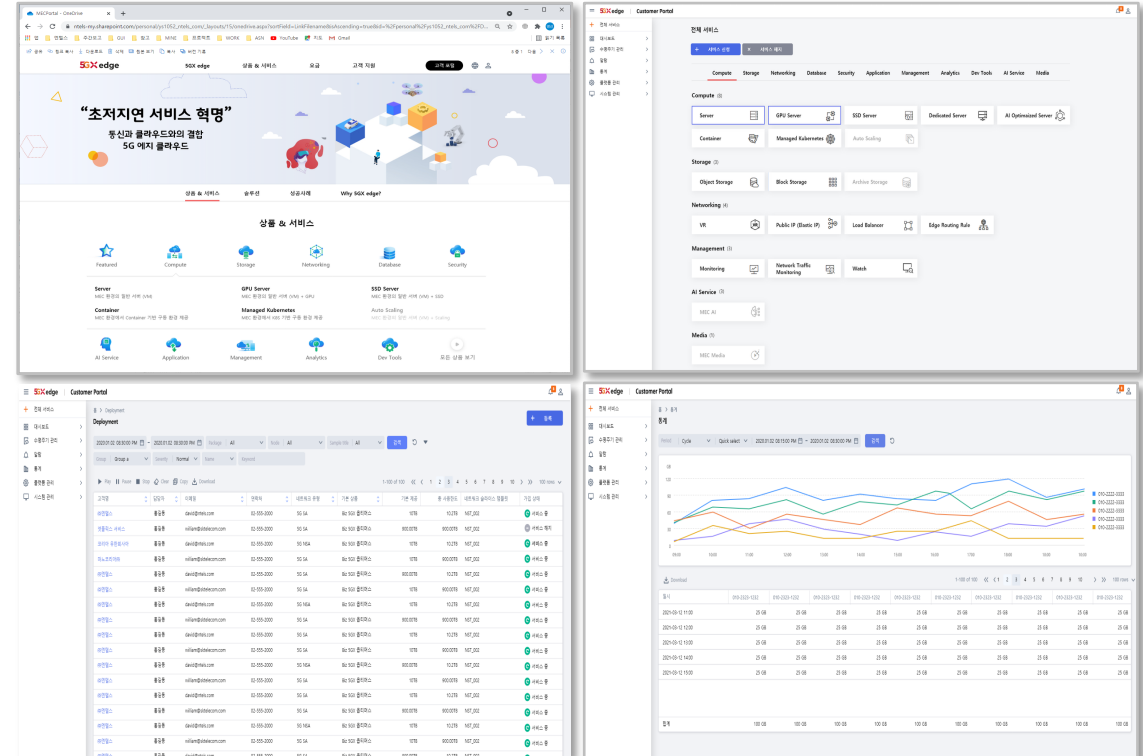
is optional

SKT's 5G MEC Platform Solutions

MEC Orchestrator



MEC Biz Console



Current Challenges in Telco

Challenges (1/3)

- **Legacy Applications**

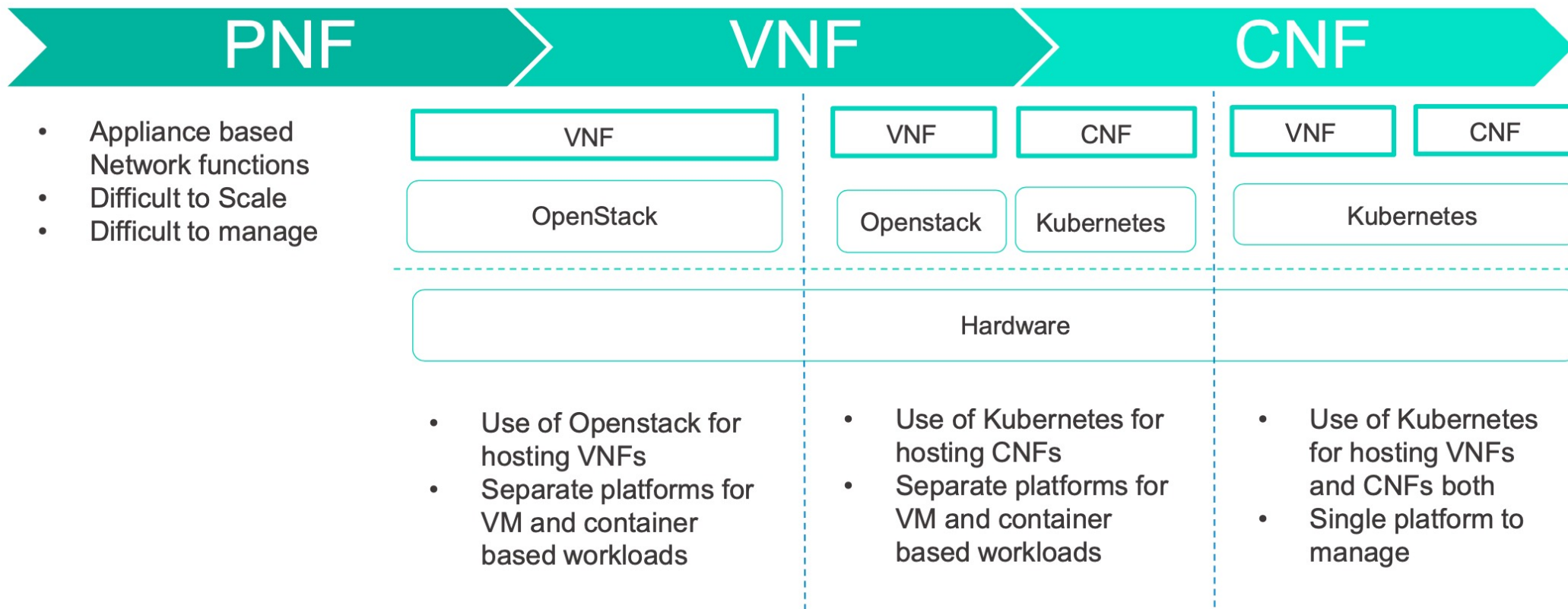
- Applications cannot or extremely difficult to be containerized
 - No source code
 - Requires entirely re-architected
 - Relies on vendor proprietary operating system (e.g., solaris, UNIX, etc.)
- Non-replaceable legacy applications

- **Virtual Networking Functions (VNF) are still there**

- Longer cycle of transition from VNF to CNF
- Container Networking Functions (CNF) itself is still evolving to meet Telco's requirements
- The majority of network functions in LTE are still based on VNF
- LTE and NSA-5G will co-exist for long time

Challenges (2/3)

- No Single Hosting Platform for VNF and CNF**



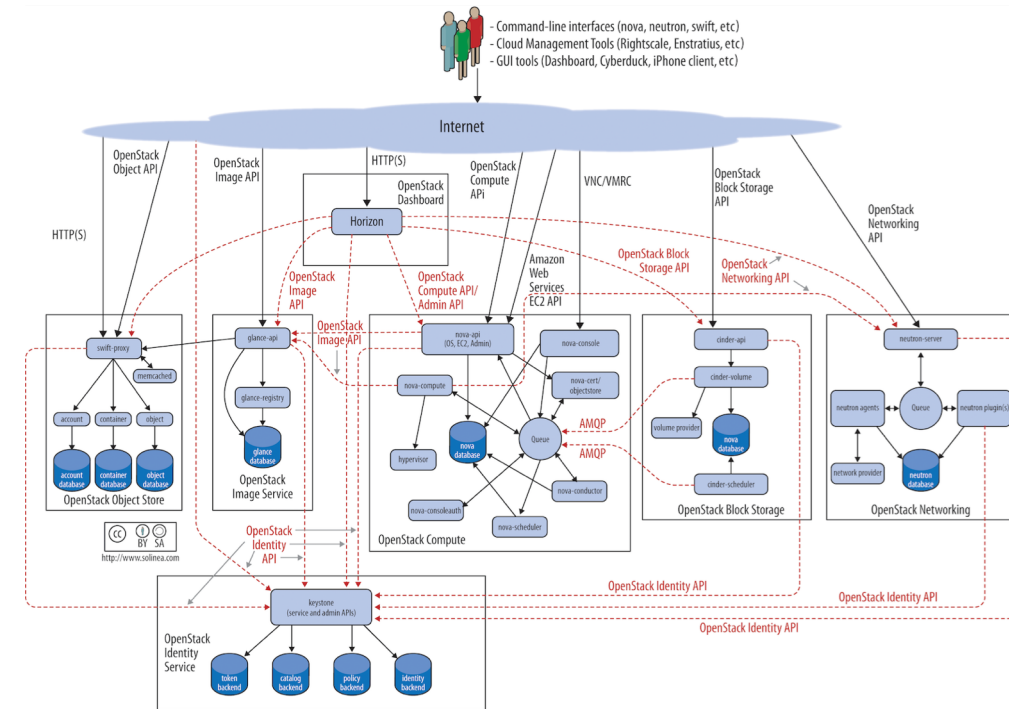
Challenges (3/3)

- **Requirements in MEC Platform**

- Requires both VNFs and CNFs hosting capabilities
- Requires to host third party applications (VM/container)

- **Limitations**

- No native support for VM hosting with Kubernetes
- To meet above two requirements need two different platforms
 - OpenStack based
 - Kubernetes based
- Adoption of two different platforms increases CAPEX and OPEX
- OpenStack platform
 - Too complex to manage
 - Deploying the entire stack is a nightmare



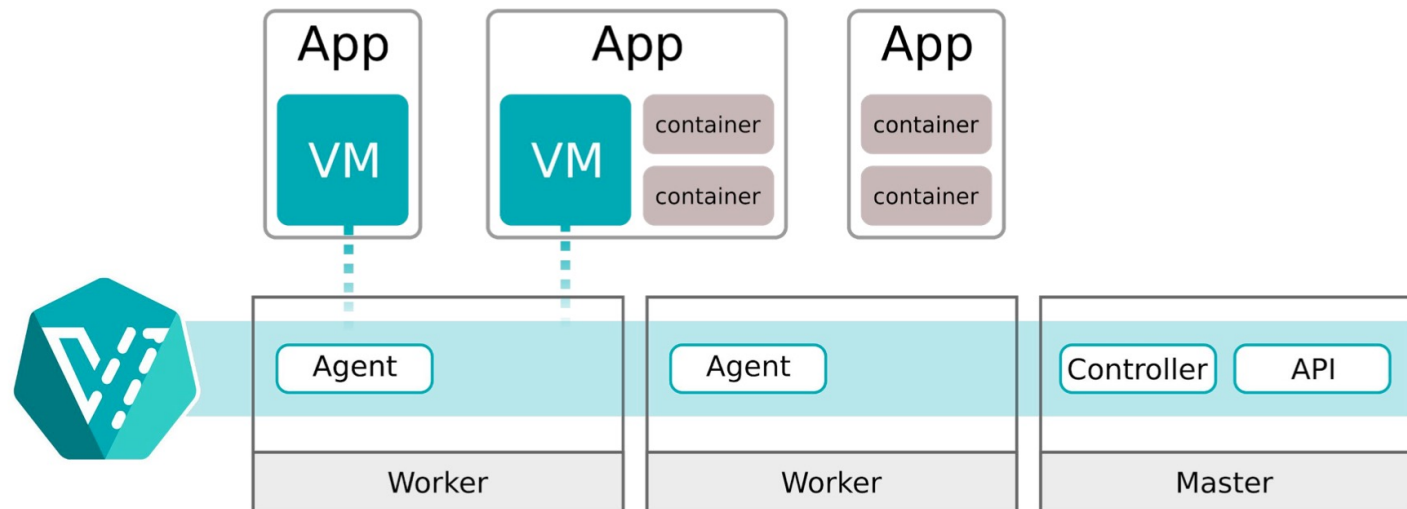
OpenStack Architecture

KubeVirt Overview

KubeVirt (1/2)

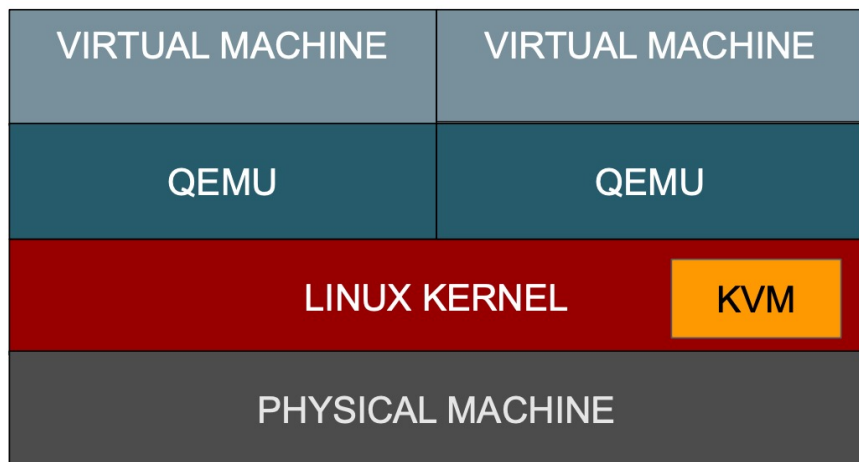
- **Brief Summary**

- CNCF Sandbox project (<https://kubevirt.io/>)
- Aims to run both VM and POD (container) on a single Kubernetes platform
- Helps in marrying VM and container world
- Use similar mechanism to manage VM like POD

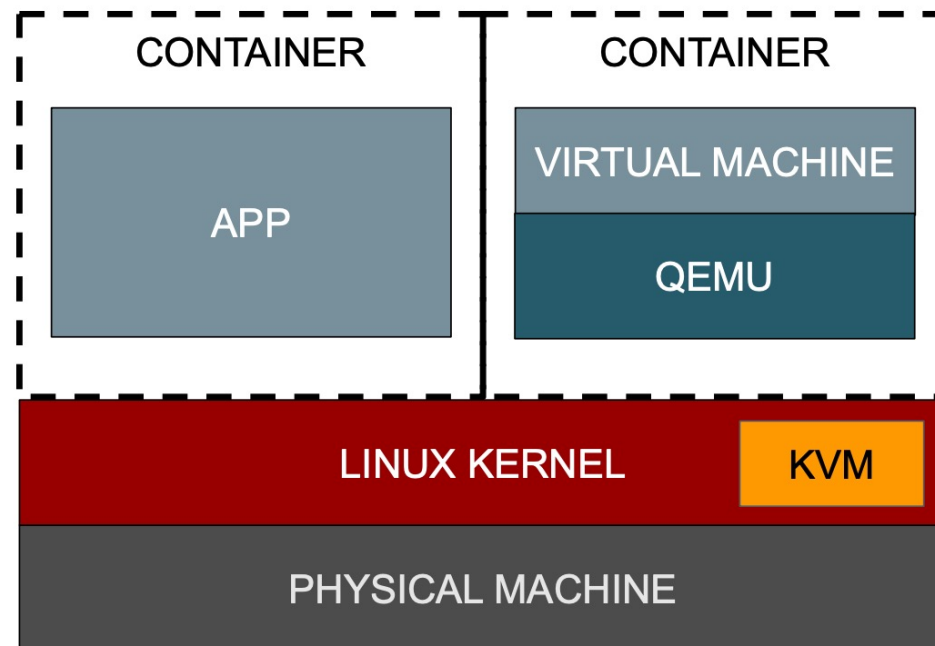


KubeVirt (2/2)

Traditional Linux Virtualization Stack



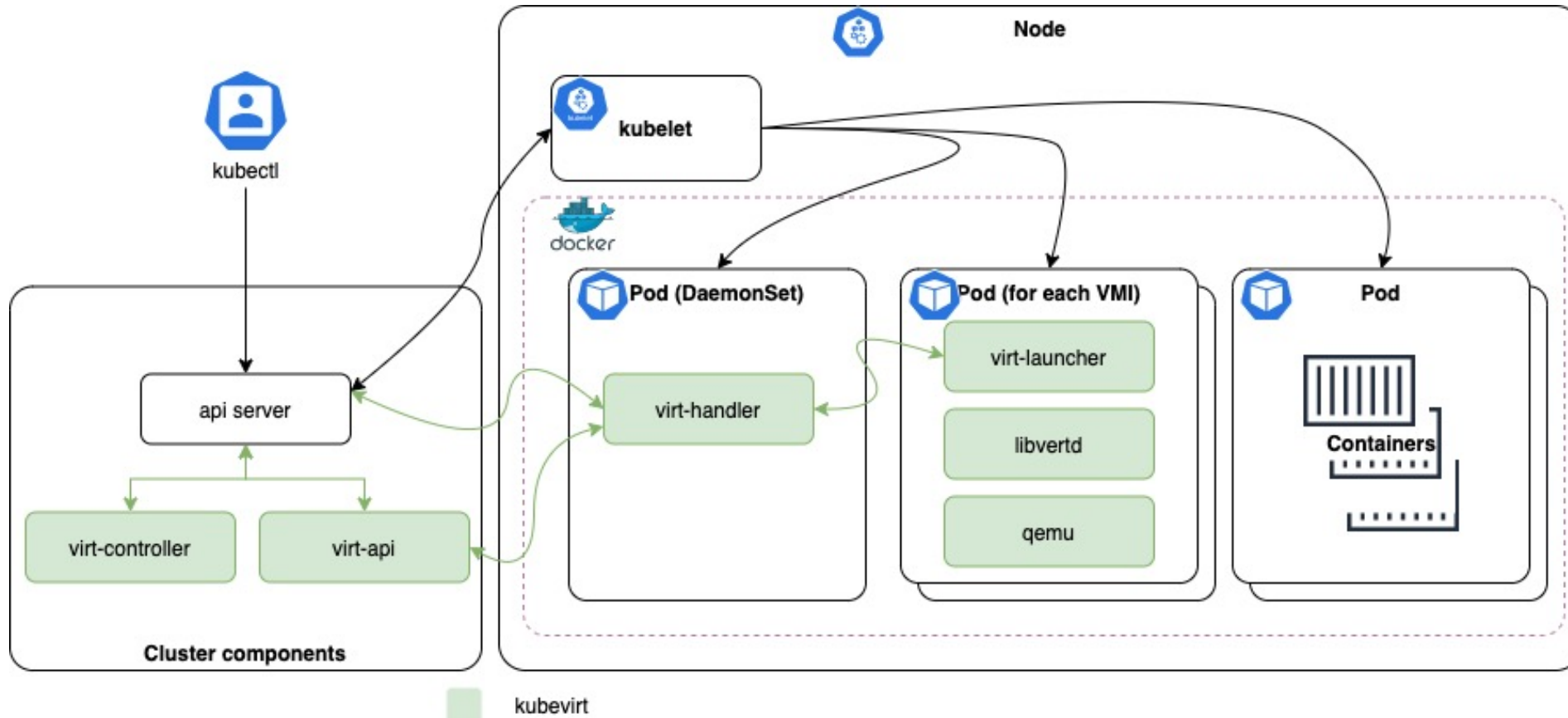
Run Containers and VMs side by side



KubeVirt Architecture

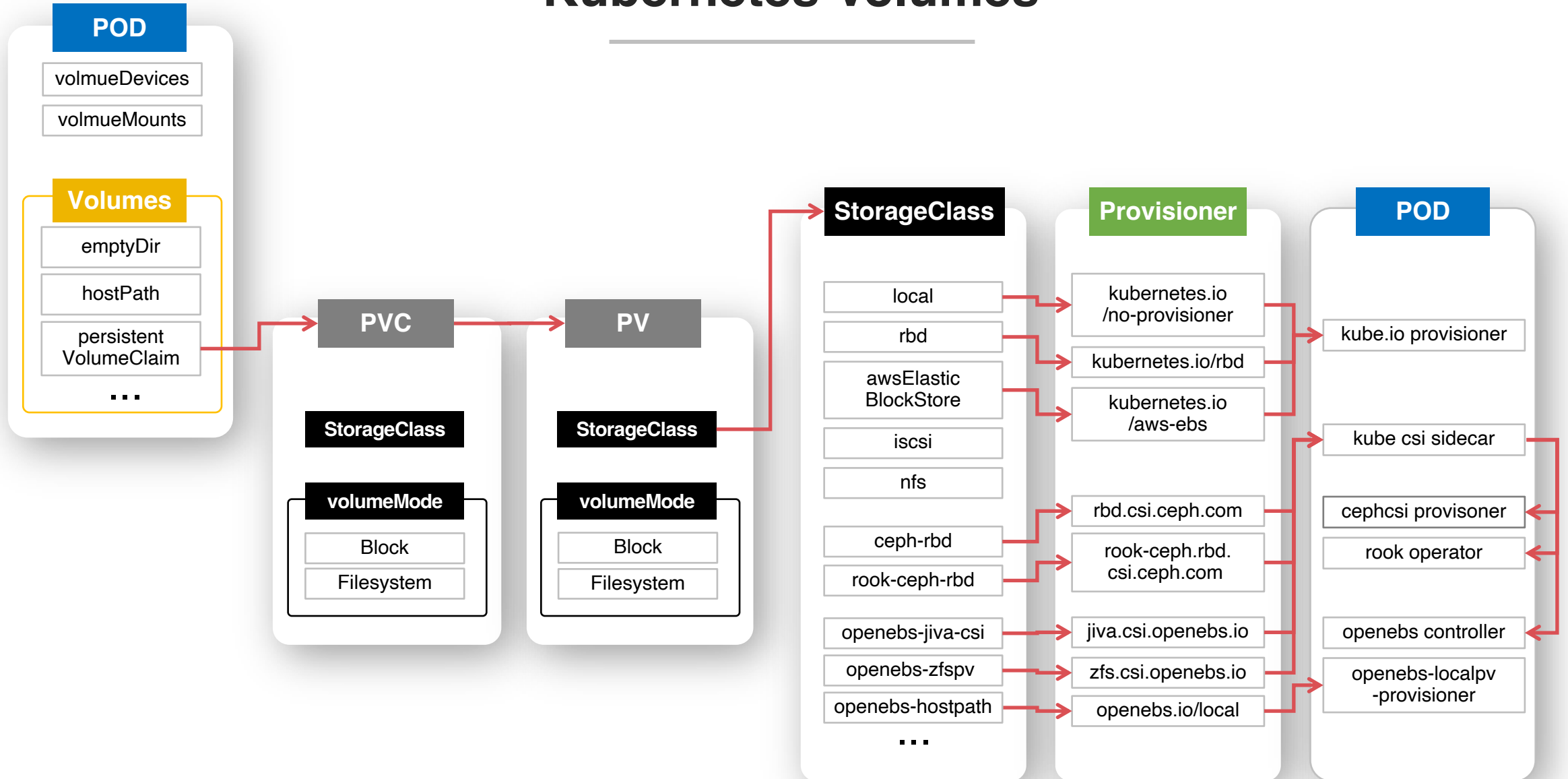
- **KubeVirt Components**

- virt-api-server, virt-controller, virt-handler, virt-launcher, libvirt

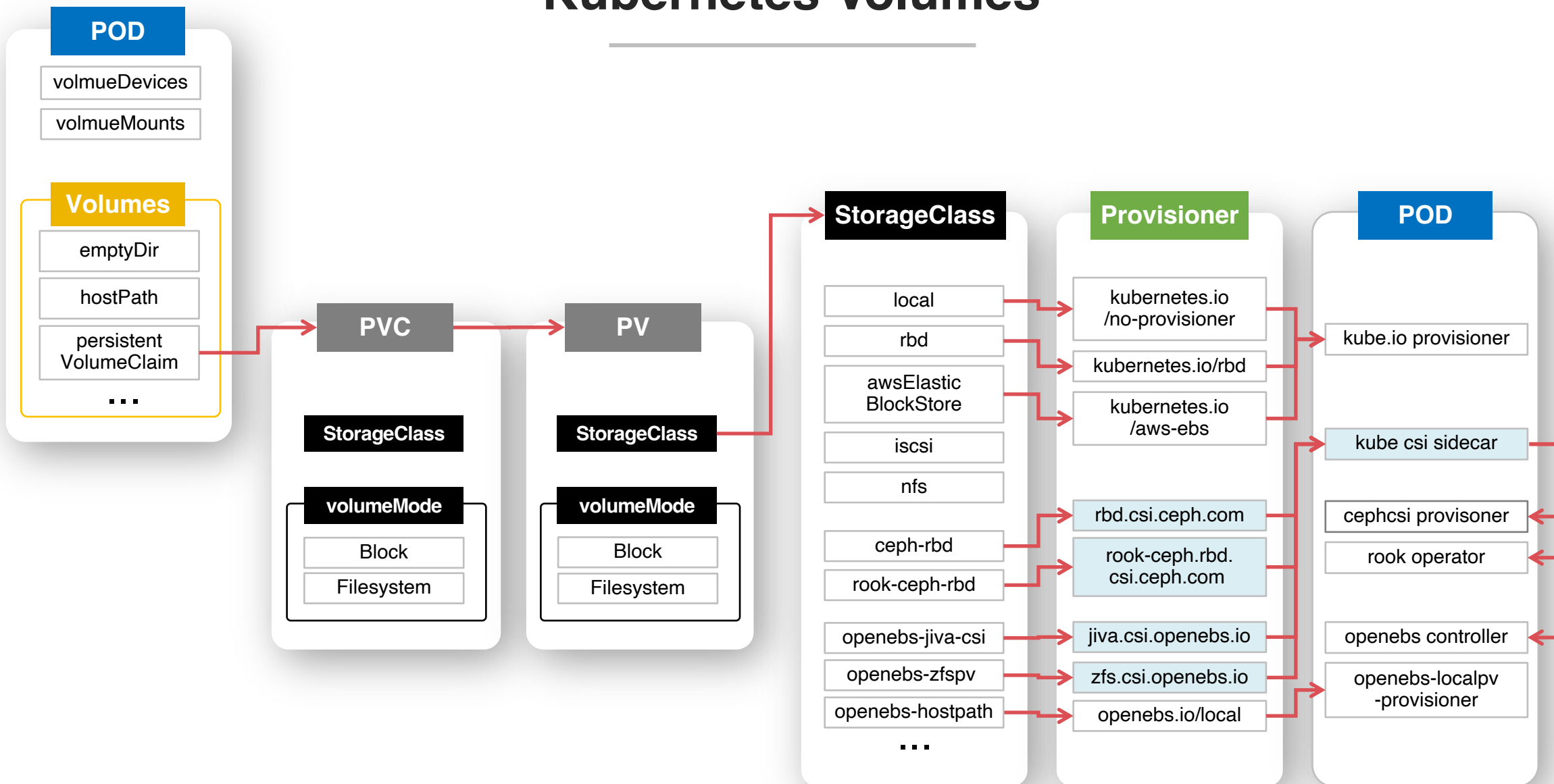


KubeVirt Volumes

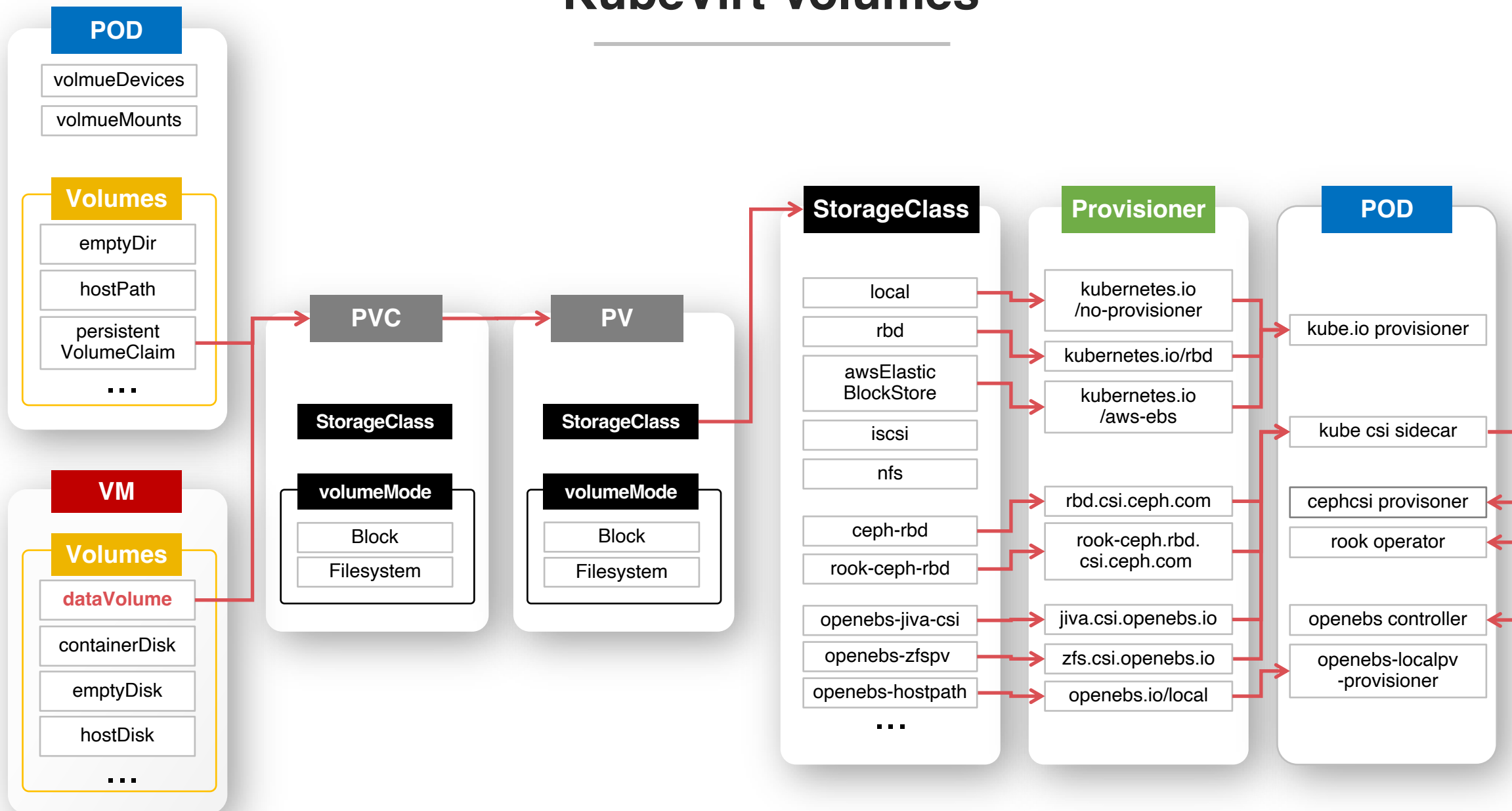
Kubernetes Volumes



Kubernetes Volumes



KubeVirt Volumes



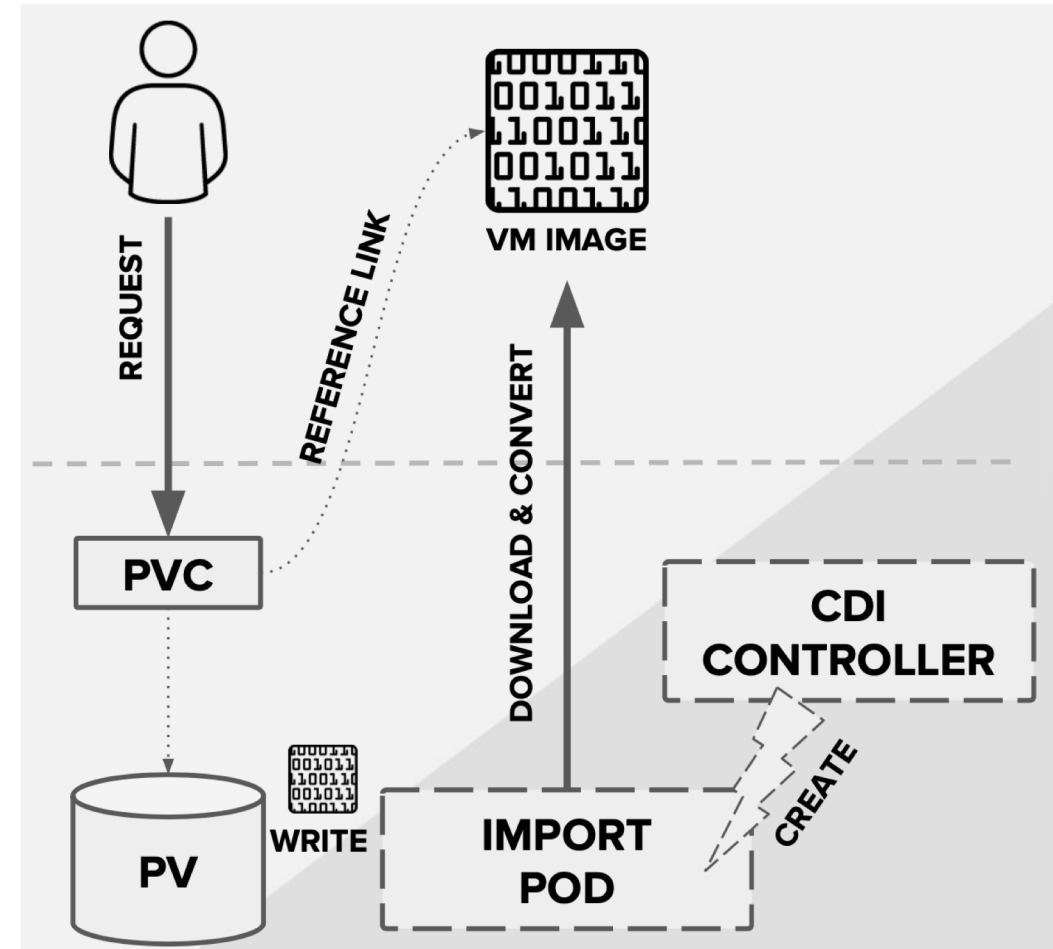
CDI Data Volumes

- **Containerized Data Importer (CDI)**

- Provides DataVolume resource that encapsulates both PVC and VM image imports in a single object

```

apiVersion: cdi.kubevirt.io/v1alpha1
kind: DataVolume
metadata:
  name: my-cloned-dv
spec:
  source:
    http:
      url: "https://cloud.centos.org/centos/7/images/CentOS-7-x86_64-GenericCloud.qcow2"
  pvc:
    accessModes:
      - ReadWriteOnce
    resources:
      requests:
        storage: 10Gi
      storageClassName: storage-provisioner
  
```



Direct-attached Storage & Replication supported CAS

MEC Storage Challenges

- **MEC Infrastructure Environment Features**
 - Limited resources & costs
 - Needs support for ultra-low latency & high-performance service
- **MEC Requirements**
 - High availability & stable shared storage pool in resource-constrained MEC
 - Storage function suitable for MEC with limited storage capacity
 - High-performance storage service to meet ultra-low latency MEC App.

MEC Storage Challenges

- **MEC Infrastructure Environment Features**
 - Limited resources & costs
 - Needs support for ultra-low latency & high-performance service
- **MEC Requirements**
 - High availability & stable shared storage pool in resource-constrained MEC
→ **Hyper-converged Distributed Storage**
 - Storage function suitable for MEC with limited storage capacity
→ **Tiering based Resource Optimization**
 - High-performance storage service to meet ultra-low latency MEC App.
→ **DPU / NVMe(of) based HW accelerated Edge Storage**

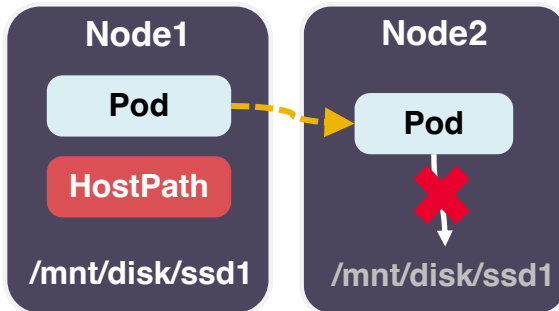
Direct-attached Storage



- **Portworx(pure storage) / storageOS with good performance ! → but, vendor lock-in**
- **Limited resources & costs → DAS required → OpenEBS LocalPV !**
 - To support HA, App must support replication such as 3 Copy (most DBs such as redis)
 - Need backup

OpenEBS LocalPV

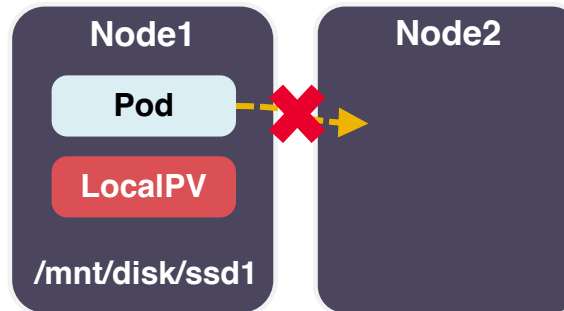
- **Local Persistent Volume**
 - Volume topology aware pod scheduling (vs. HostPath)
 - Cons: Dynamic provisioning not supported
- **OpenEBS LocalPV**
 - Local Persistent Volume + **Dynamic provisioning**



HostPath

Not working!

Node2 doesn't have /mnt/disk/ssd1



LocalPV

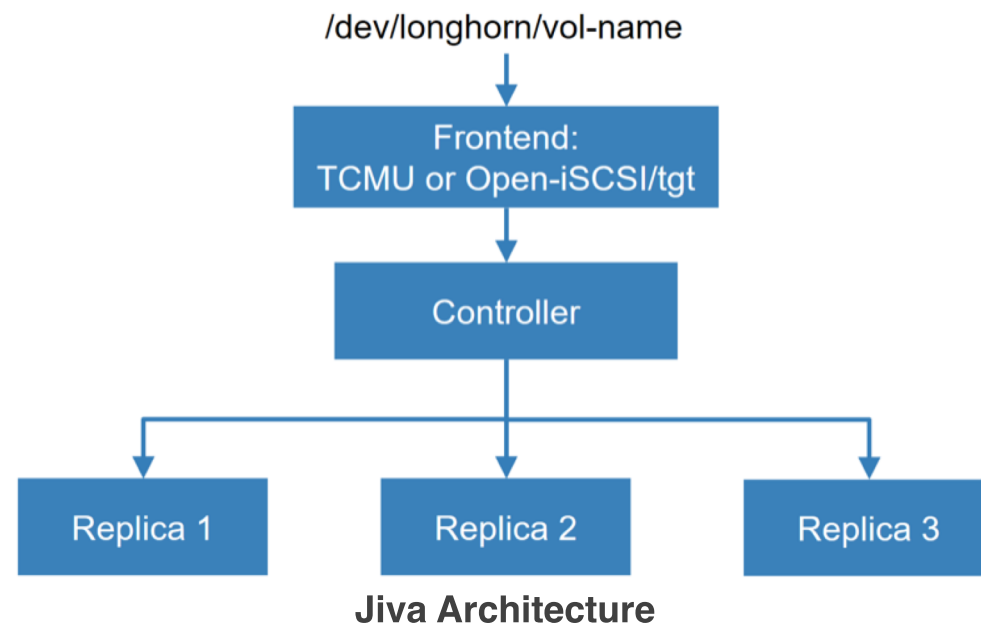
**POD is scheduled
based on "Node Affinity"**

```
apiVersion: v1
kind: PersistentVolume
metadata:
  annotations:
    pv.kubernetes.io/provisioned-by: openebs.io/local
  creationTimestamp: "2021-11-08T16:26:20Z"
  finalizers:
    - kubernetes.io/pv-protection
  labels:
    openebs.io/cas-type: local-hostpath
  name: pvc-d8e1713d-4cea-4fc3-a28b-d0f54e61754a
  resourceVersion: "11809"
  uid: ed746b74-3cff-48ed-9008-10733d895b90
spec:
  accessModes:
    - ReadWriteOnce
  capacity:
    storage: 50Gi
  claimRef:
    apiVersion: v1
    kind: PersistentVolumeClaim
    name: openebs-pvc-dcc04a57-f714-425d-a808-adeeb43eeac5-jiva-rep-0
    namespace: openebs
    resourceVersion: "11604"
    uid: d8e1713d-4cea-4fc3-a28b-d0f54e61754a
  local:
    fsType: ""
    path: /var/lib/openebs/local/pvc-d8e1713d-4cea-4fc3-a28b-d0f54e61754a
  nodeAffinity:
    required:
      nodeSelectorTerms:
        - matchExpressions:
            - key: kubernetes.io/hostname
              operator: In
              values:
                - master-node-1
  persistentVolumeReclaimPolicy: Delete
  storageClassName: openebs-hostpath
  volumeMode: Filesystem
status:
  phase: Bound
```

OpenEBS LocalPV

Replication supported CAS (OpenEBS)

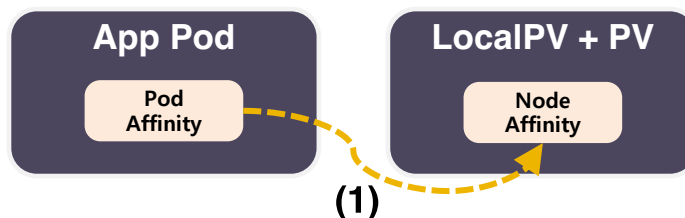
APPLICATION REQUIREMENTS	STORAGE TYPE	OPENEBS VOLUMES
Low Latency, High Availability, Synchronous replication, Snapshots, Clones, Thin provisioning	SSDs/Cloud Volumes	OpenEBS Mayastor
High Availability, Synchronous replication, Snapshots, Clones, Thin provisioning	Disks/SSDs/Cloud Volumes	OpenEBS cStor
High Availability, Synchronous replication, Thin provisioning	hostpath or external mounted storage	OpenEBS Jiva
Low latency, Local PV	hostpath or external mounted storage	Dynamic Local PV - Hostpath
Low latency, Local PV	Disks/SSDs/Cloud Volumes	Dynamic Local PV - Device
Low latency, Local PV, Snapshots, Clones	Disks/SSDs/Cloud Volumes	OpenEBS Dynamic Local PV - ZFS



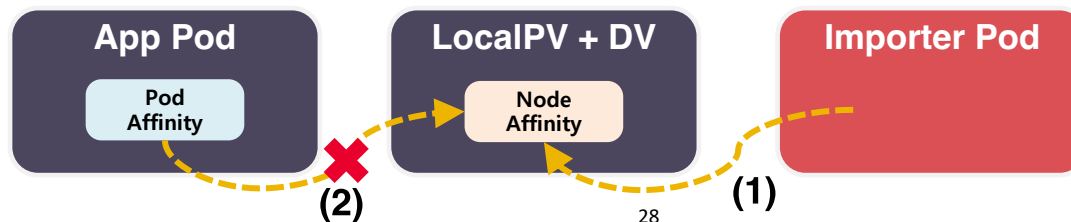
- **cStor:** iSCSI & ZFS based replication supported CAS
- **Jiva:** iSCSI & Longhorn (SUSE) based replication supported CAS
- **Mayastor:** Low latency workloads for converged and segregated storage by leveraging NVMe / NVMe over Fabrics (NVMe-oF / SPDK)

POD & Node Affinity with DV (1/2)

- If using DV, POD doesn't properly support "POD Affinity"
- LocalPV + PV
 - Volume binding mode: "WaitForFirstConsumer"
 - Once App POD affinity is determined, OpenEBS provisioner receives the affinity and sets PV node affinity with it (1)

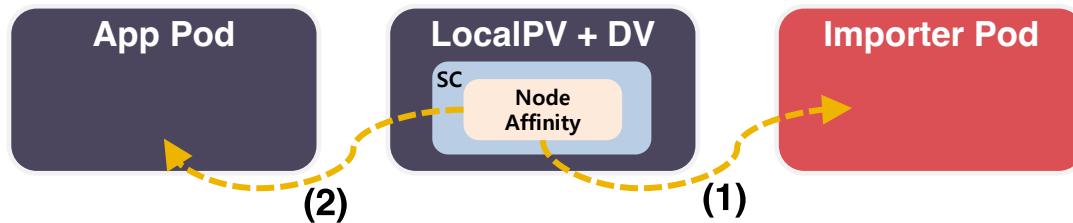


- LocalPV + DV
 - Volume binding mode: like "Immediate"
 - DV is created, mounted on the CDI Importer and VM Image is downloaded
 - **DV node affinity is set as CDI Importer's POD affinity (1)**
 - If App POD affinity and DV affinity are different, DV will not mount on App POD (2)



POD & Node Affinity with DV (2/2)

- If using DV, POD doesn't properly support "POD Affinity"
- Solution
 - To select specific node(s), create SC by specifying target node in "allowedTopologies"
 - DV selects targeted SC
 - Do not select POD affinity for App Pod, select the prepared dv

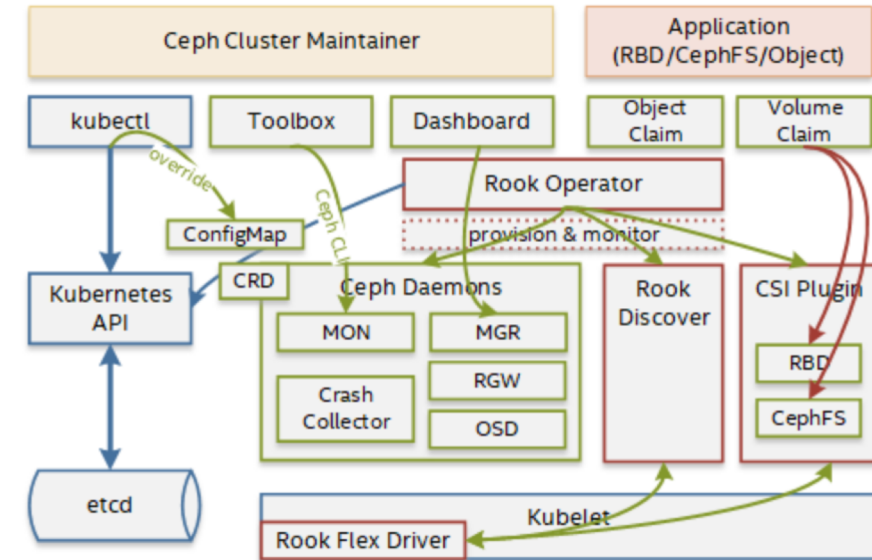
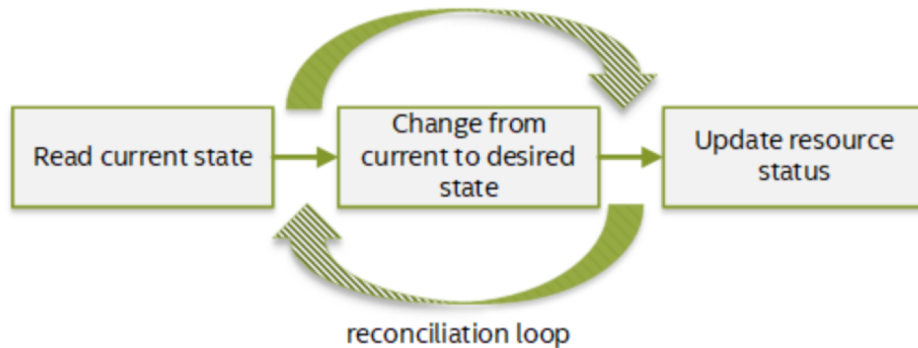


```
allowedTopologies:
- matchLabelExpressions:
  - key: kubernetes.io/hostname
    values:
      - worker-node-1
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  annotations:
    cas.openebs.io/config: |
      - name: StorageType
        value: hostpath
      - name: BasePath
        value: /var/lib/openlibs/local
    kubectl.kubernetes.io/last-applied-configuration: |
      {"allowedTopologies":[{"matchLabelExpressions":[{"key": "kubernetes.io/hostname", "values": ["worker-node-1"]}]}]}
StorageType\n value: hostpath\n- name: BasePath\n val
ForFirstConsumer"}
openebs.io/cas-type: local
creationTimestamp: "2021-11-08T16:21:06Z"
name: worker-node-1-static
resourceVersion: "9794"
uid: c1e864f6-8be3-400a-af7a-06cabd9f56e7
provisioner: openebs.io/local
reclaimPolicy: Delete
volumeBindingMode: WaitForFirstConsumer
```

StorageClass Example

Distributed Scale-out Storage & HA

Rook



- Open Source cloud-native storage orchestrator
- Run Ceph on top of kubernetes and manage via kubernetes
 - A simple container that has all that is needed to bootstrap and monitor the storage cluster
 - The operator starts and monitors Ceph daemon pods, such as MONs, OSDs, MGR, and others.
 - Monitors the daemon to ensure the cluster is healthy
 - Ceph MONs are started or failed over when necessary
 - Other adjustments are made as the cluster grows or shrinks

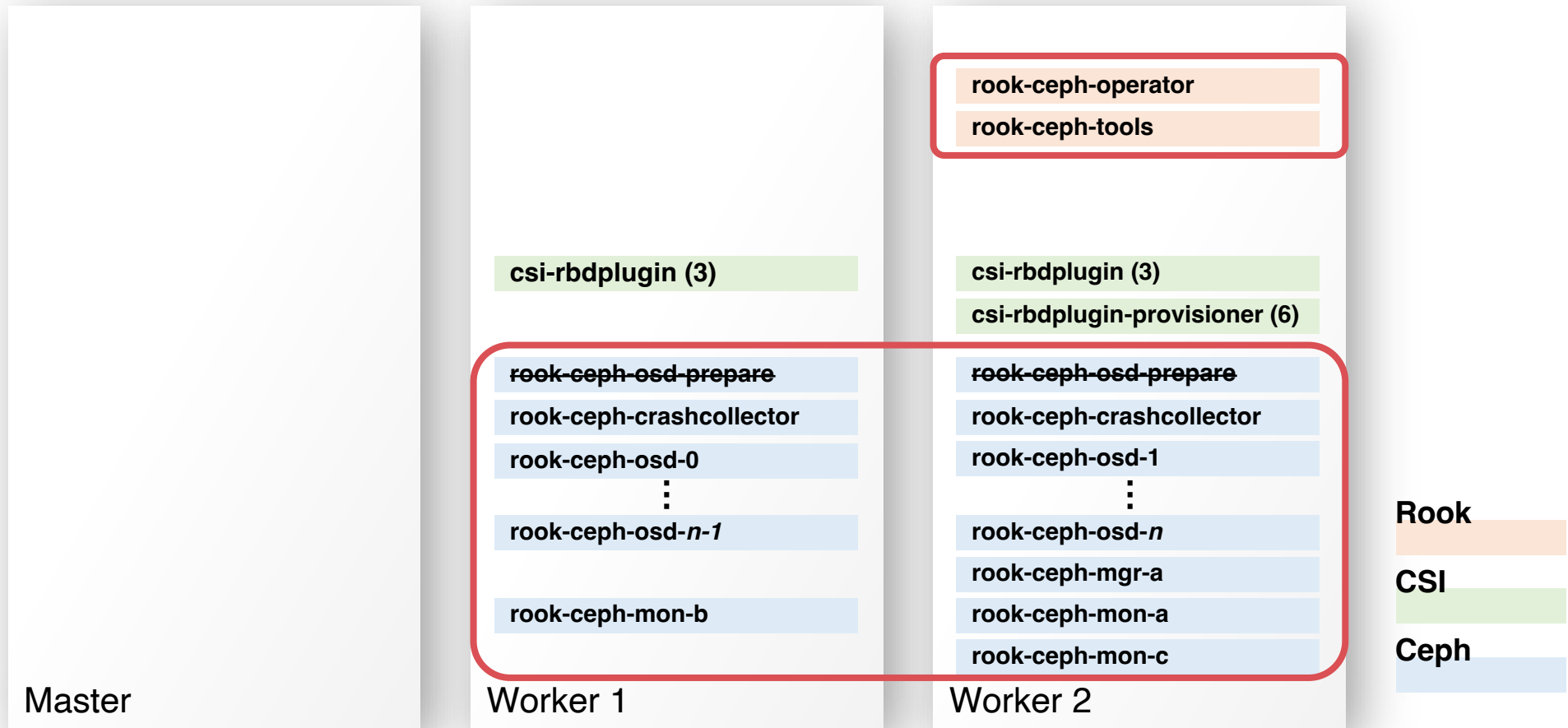
Rook vs. Ceph-CSI (1/3)

- Ceph-CSI

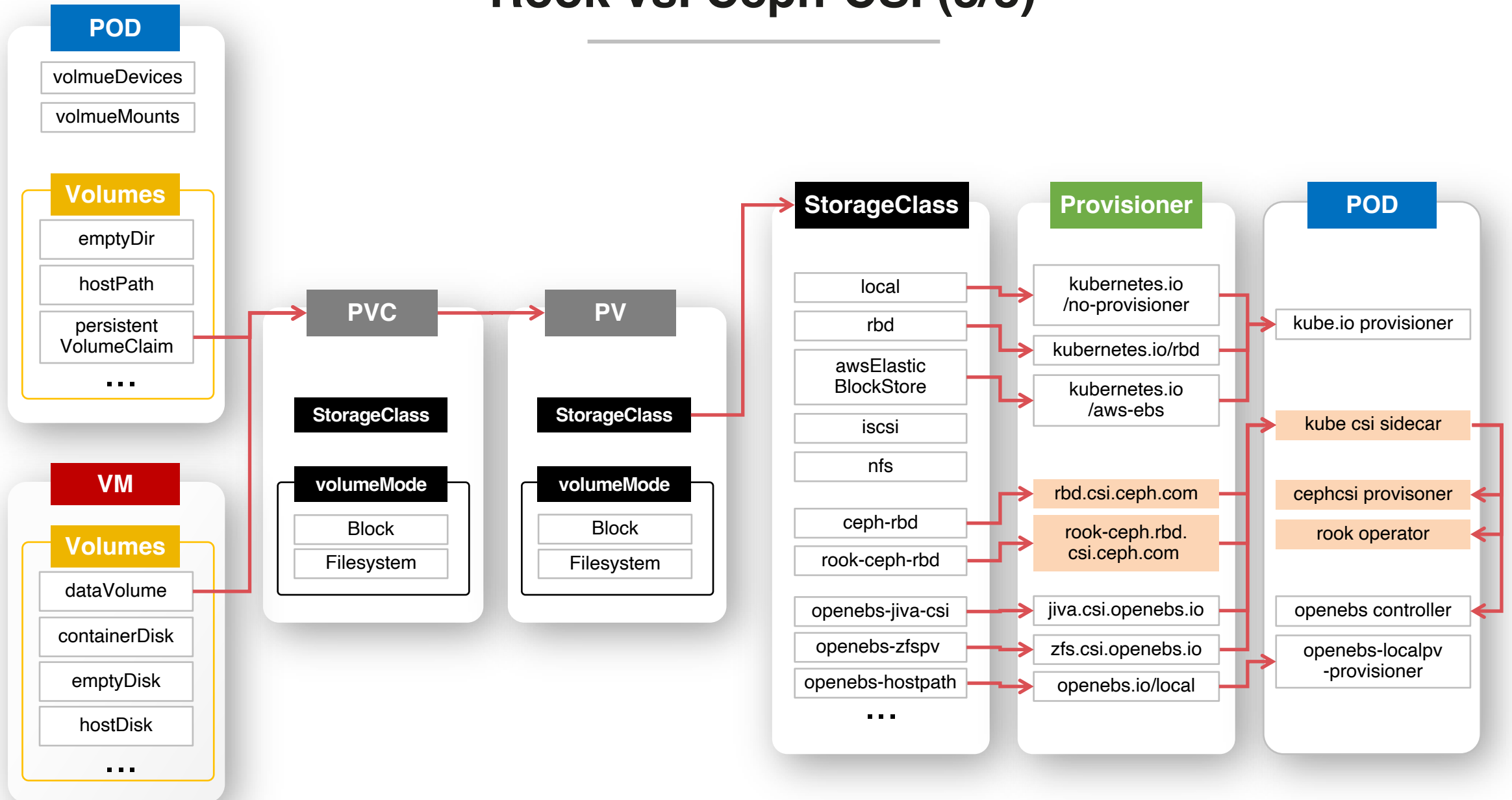


Rook vs. Ceph-CSI (2/3)

- Rook



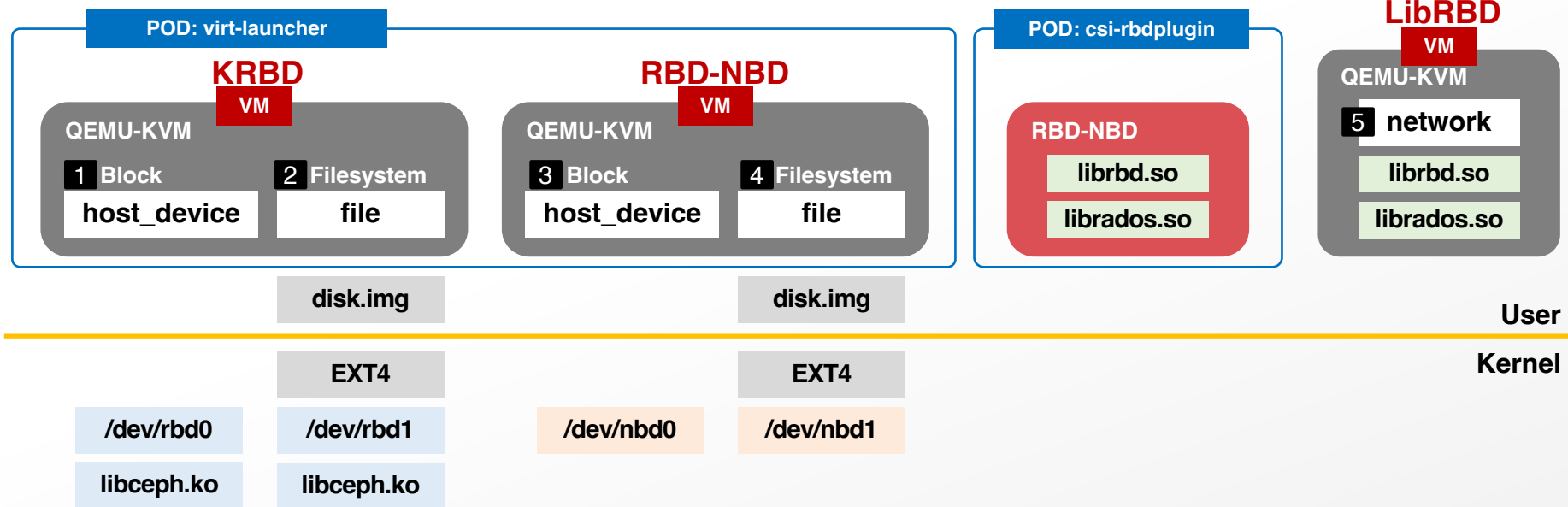
Rook vs. Ceph-CSI (3/3)



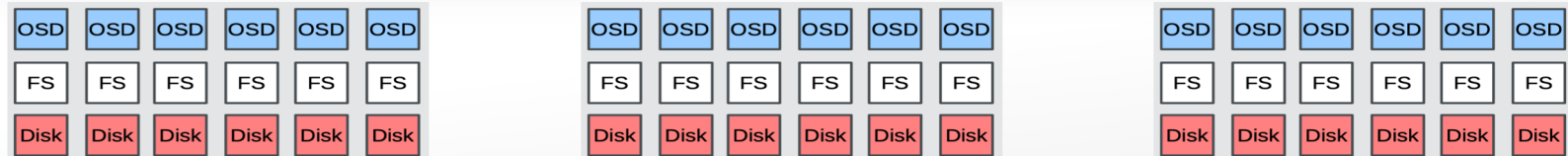
Network Block Device with KubeVirt (1/4)

- Ceph client: KRBD vs. RBD-NBD vs. LibRBD
- Qemu disk type: “host_device” vs. “file” vs. “network”
- Rbd-nbd support status: Alpha

CLIENT

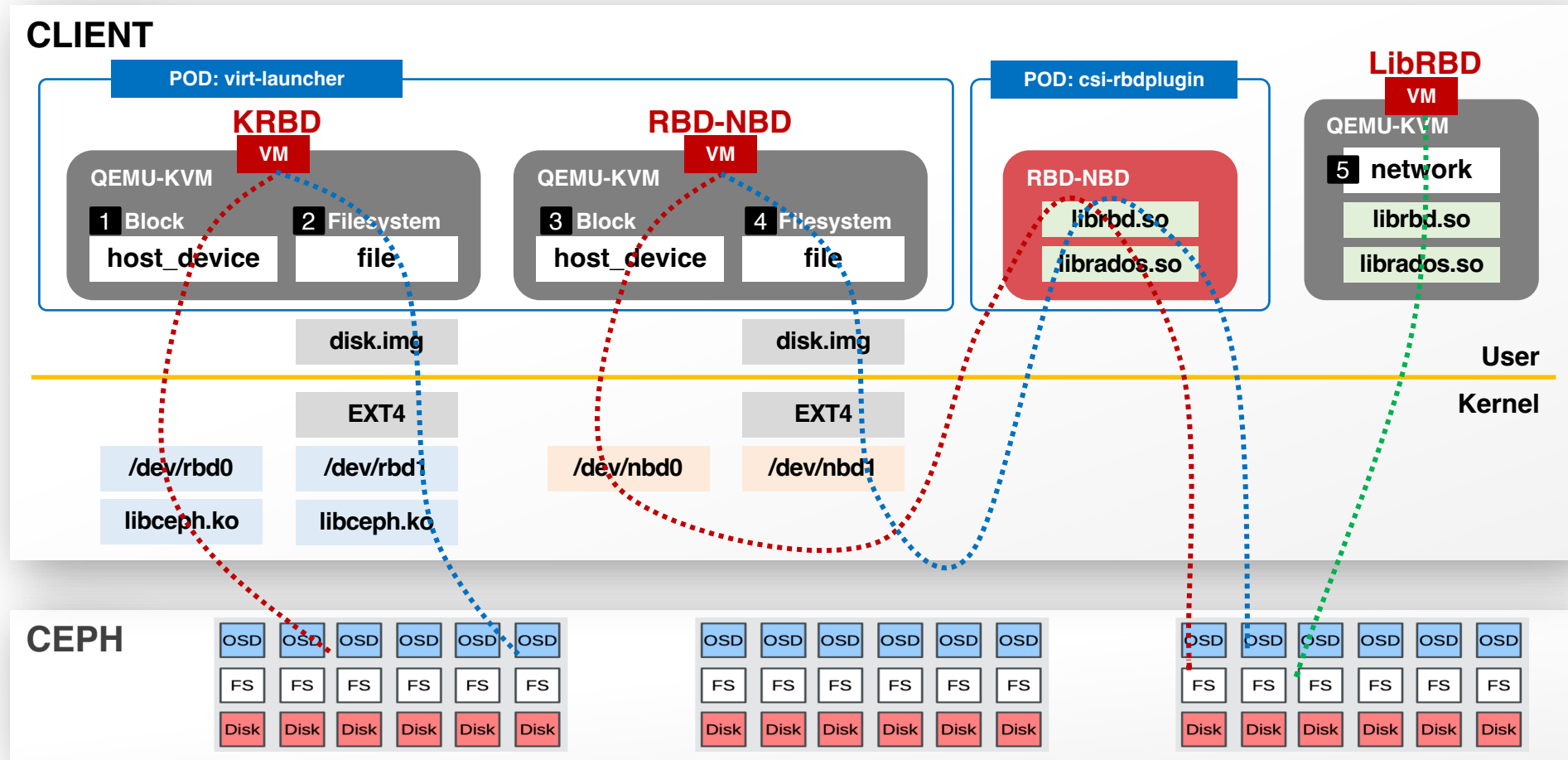


CEPH



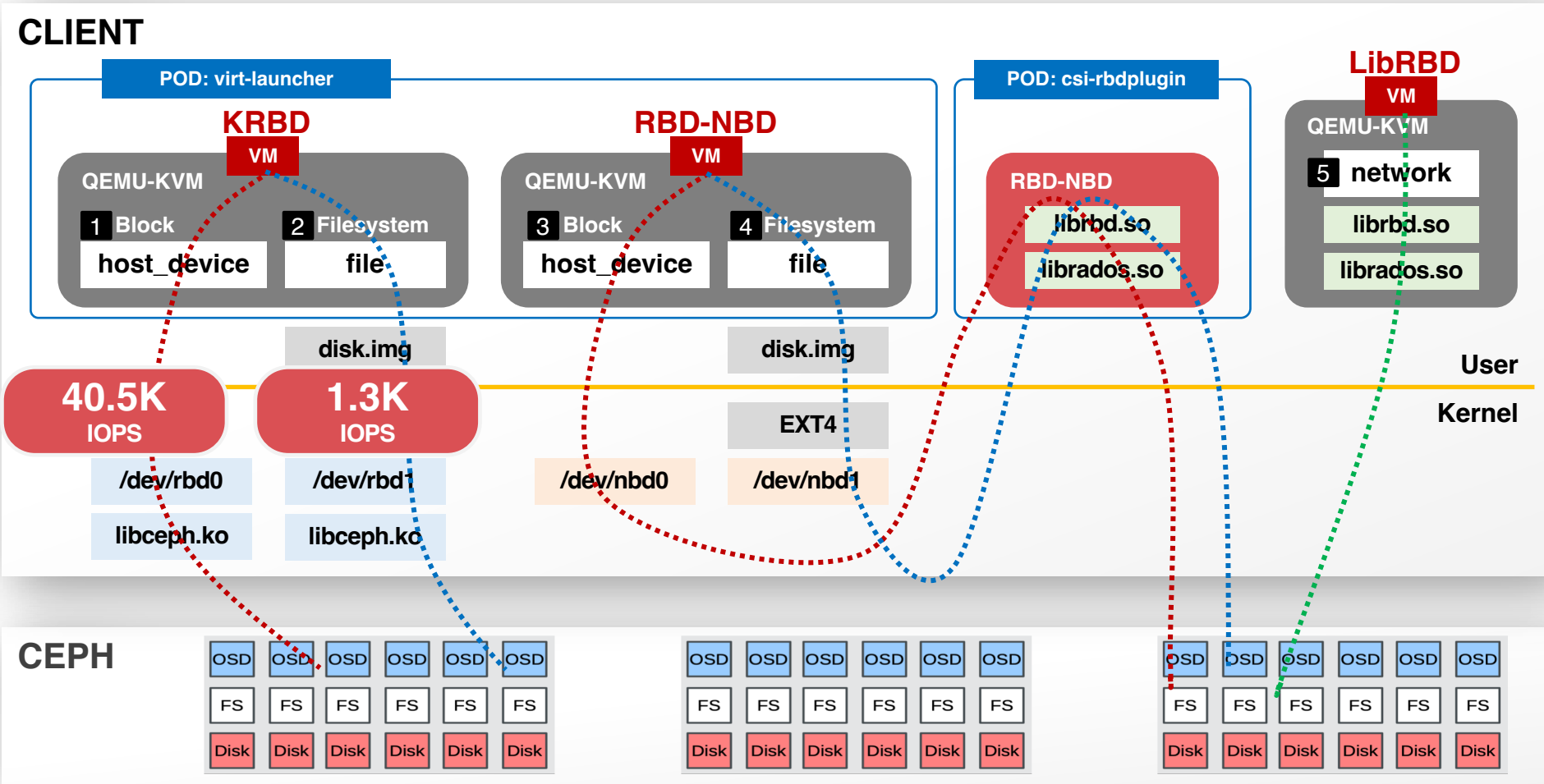
Network Block Device with KubeVirt (2/4)

- Ceph client: KRBD vs. RBD-NBD vs. LibRBD
- Qemu disk type: “host_device” vs. “file” vs. “network”
- Rbd-nbd support status: Alpha



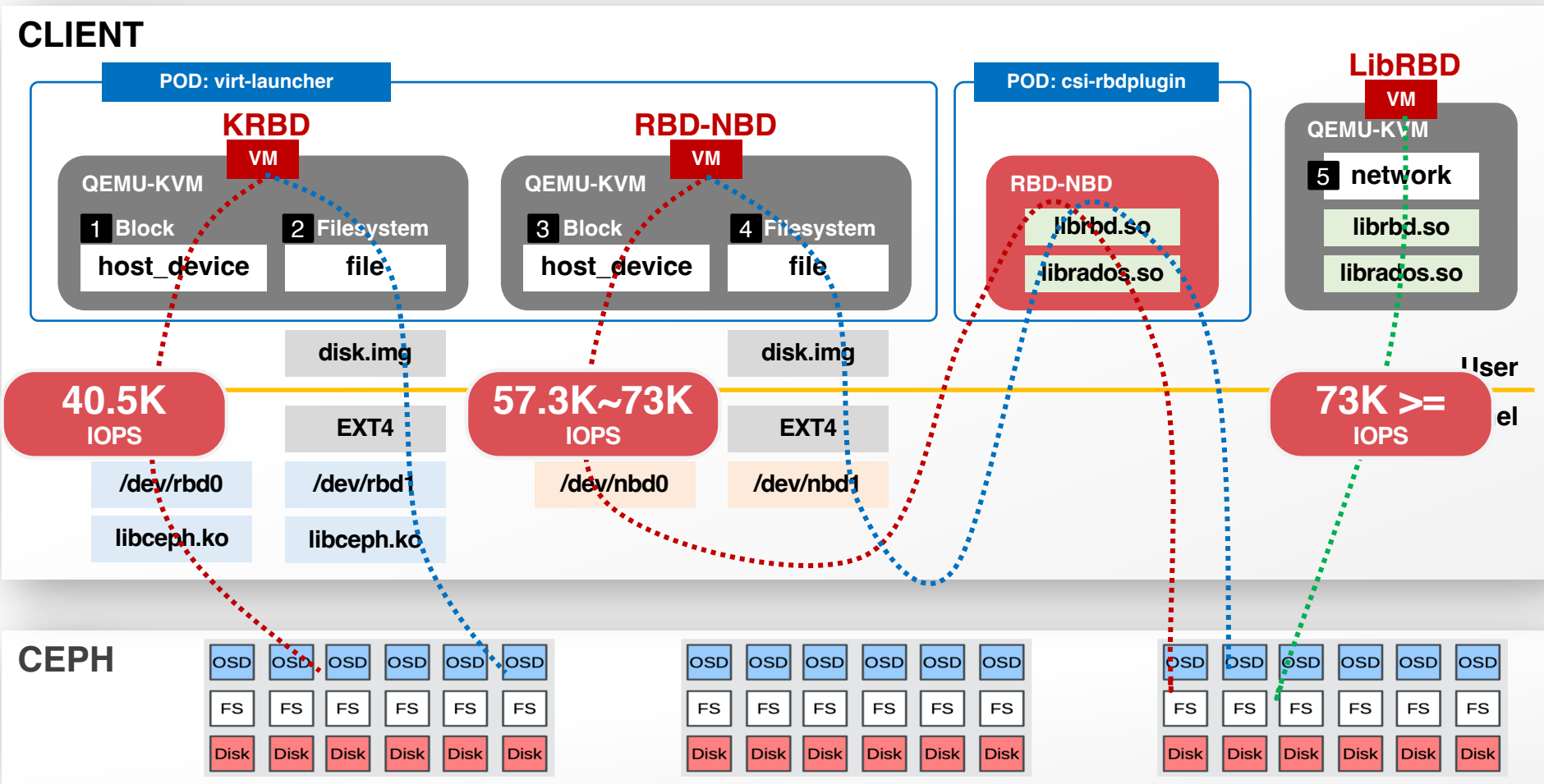
Network Block Device with KubeVirt (3/4)

- `fio --norandommap --group_reporting --rw=randwrite --direct=1 --bs=4K --iodepth=8 --numjobs=8 --name file1 --randrepeat=0 --ramp_time=10 --ioengine=libaio --filename=/dev/vdc --runtime=120 --allow_mounted_write=1`
- x 3 Clients



Network Block Device with KubeVirt (4/4)

- `fio --norandommap --group_reporting --rw=randwrite --direct=1 --bs=4K --iodepth=8 --numjobs=8 --name file1 --randrepeat=0 --ramp_time=10 --ioengine=libaio --filename=/dev/vdc --runtime=120 --allow_mounted_write=1`
- x 3 Clients



HA Policy with POD Owning Non-shareable PVC (1/3)

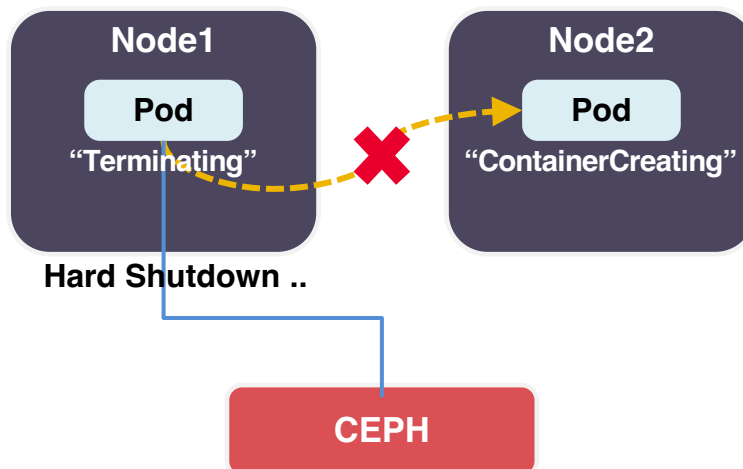
- **Hard Shutdown case**

- Network outage or more than 300 sec network delay

- **Some PODs cannot move to another node and are stuck still in “Terminating” even after 300**

edgestack	edgetron-547d496848-mnbxj	1/1	Running	0	6m10s	10.233.105.22	kubevirt-master-2
edgestack	edgetron-547d496848-zk6k9	1/1	Terminating	0	32m	10.233.68.8	kubevirt-master-1
edgestack	registry-docker-registry-545f9fb45-9l85h	1/1	Terminating	0	17m	10.233.68.21	kubevirt-master-1
edgestack	registry-docker-registry-545f9fb45-sclsn	0/1	ContainerCreating	0	6m10s	<none>	kubevirt-master-3
edgestack	sona-onos-0	2/3	Running	0	32m	10.233.105.8	kubevirt-master-2
edgestack	sona-onos-1	3/3	Running	0	32m	10.233.78.8	kubevirt-master-3

kubectl get po -A -o wide

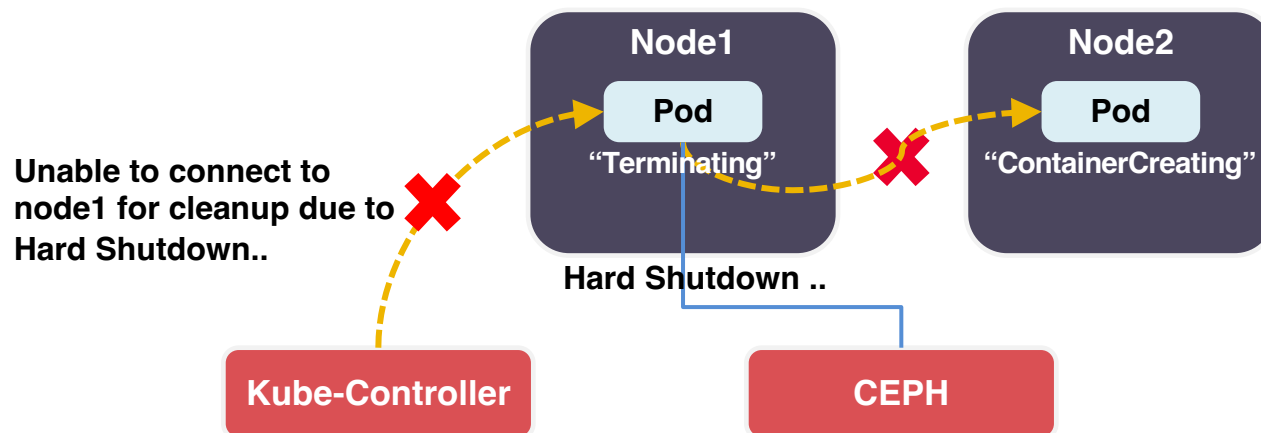


HA Policy with POD Owing Non-shareable PVC (2/3)

Reason

- Kubernetes can complete terminating of POD only when eventually connected to shutdown node
- POD with non-shareable PVC (**ReadWriteOnce**)
 - If Terminating is not completed, impossible to create POD at other nodes
 - Due to PVC multi-attachment attempts

Type	Reason	Age	From	Message
Normal	Scheduled	23m	default-scheduler	Successfully assigned edgestack/registry-docker-registry-856bcd4d4-jh5s7 to kubevirt-master-2
Warning	FailedAttachVolume	23m	attachdetach-controller	Multi-Attach error for volume "pvc-4cdfedbb-3ba1-4fac-bf6a-9d9e098eb05f" Volume is already used by pod(s) re
Warning	FailedMount	16m (x2 over 21m)	kubelet	Unable to attach or mount volumes: unmounted volumes=[data], unattached volumes=[registry-docker-registry-co



HA Policy with POD Owning Non-shareable PVC (3/3)

- **Solutions**

- Force remove “Terminating POD”
 - through frequent monitoring or alarms by administrator
 - through Kubernetes ConJob automatically (Kubernetes v1.21 GA)

- **Soft Shutdown Case**

- Graceful Node Shutdown (Kubernetes v1.21 Beta)

Thank You!

Byungsu Park

pbs1108@gmail.com