KubeCon | CloudNativeCon

North America 2024

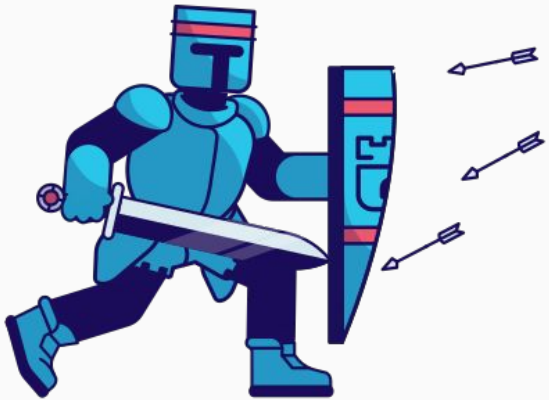
Rook: Storage for Kubernetes

Travis Nielsen, Blaine Gardner, Annette Clewett
IBM Storage

November 2024

Agenda

- Introduction to Rook and Ceph
  - Storage Platform features
  - Topology Considerations
  - Maintenance best practices
  - Project and Community
- New features in v1.15
- Roadmap for v1.16+
- Application Disaster Recovery



Raise your hand if…

- Are you here to learn about Rook for the first time?
- Have you experimented with Rook?
- Have you deployed Rook in production?
- Who is going to Cephalocon next month?

Introduction to Rook

Questions that led to Rook

- Storage is commonly provided by cloud providers
- What about storage in your datacenter?
- Storage is traditionally not part of the cluster
  - Why should storage be external to K8s?
- Why not manage storage as any other K8s application?

Storage Platform

- Which storage platform to trust?
- Enterprises don’t trust a new data platform
- We didn’t want to build a new storage platform
- Decision to build on stable, enterprise-ready **Ceph**

What is Rook?

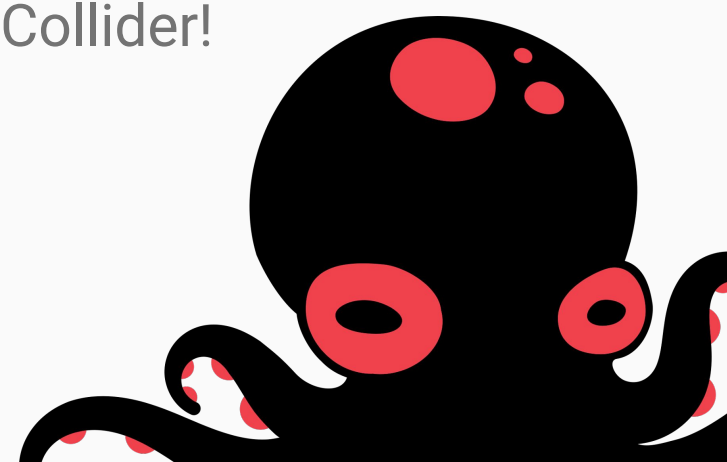
- Brings Ceph storage into your Kubernetes cluster
- Manages Ceph storage with an operator and Custom Resource Definitions (CRDs)
- Automates deployment, configuration, upgrades
- Allows apps to consume storage like any other K8s storage
  - Storage Classes, Persistent Volume Claims
- Open Source (Apache 2.0)

What is Ceph?

- Distributed Software-Defined Storage solution: https://ceph.io/
- All-in-one open-source storage platform
  - `RBD  Block storage  RWO`
  - `CephFS  Shared File System  RWX`
  - `RGW  Object storage  S3 Buckets`

Why Ceph?

- Fully Open Source, great community
- Proven history of enterprise adoption and support
  - First release in July 2012
  - Deployed in CERN’s Large Hadron Collider!
- Designed for scalability



Ceph data durability

- Ceph designed to be consistent, not eventually consistent
- Data is sharded across AZs, racks, nodes, disks
- Replication is configurable to failure domains
- Even in extreme disasters, data can be recovered manually
- Erasure coding support
  - Most commonly used for object storage, and also works with block and shared filesystem

Architectural Layers

- Rook
  - Operator owns the **deployment** and **management** of Ceph
- CSI
  - Ceph CSI driver dynamically **provisions** and **mounts** storage to user application Pods
- Ceph
  - **Data** layer

How do you install Rook?

- Helm charts
- Example manifests for many configurations
- Quickstart guide
  - https://rook.io and click Get Started

Rook installation environments

- Anywhere Kubernetes runs
  - Cloud or on-premises
  - Virtual or bare metal hardware
  - Underlying storage can be node-attached devices, cloud volumes, or loopback devices for testing

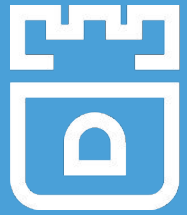
Why Deploy Rook in the Cloud?

- Overcomes limitations of cloud providers
  - Number of PVs per node
  - Poor performance of small PVs
  - Storage not available across AZs
- Enables cross-cloud support with a consistent storage platform

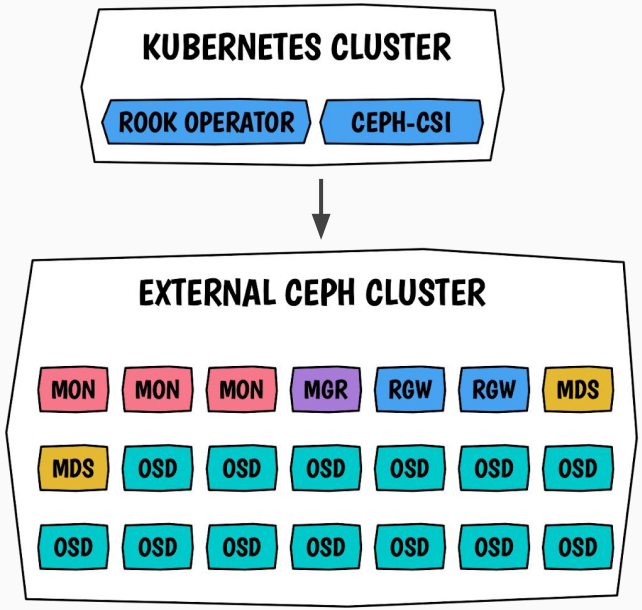
Dedicated Storage Configurations

- **Hyperconverged**
  - Default, simplest to configure on same nodes as applications
  - Set resource requests/limits to reduce cross-app impact
- **Dedicated storage nodes**
  - Node affinity/taints specify storage nodes vs app worker nodes
  - Ensures storage performance/reliability is not impacted by applications
- **External Ceph**
  - Storage resources are fully outside K8s, or in another K8s cluster
  - Centralized storage management for multiple K8s clusters

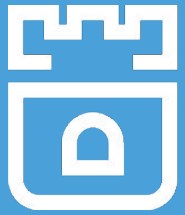
External Cluster Connection



- Connect to a Ceph cluster outside of the current K8s cluster
- Multiple K8s clusters can access the same storage platform
- Storage between client clusters can be isolated

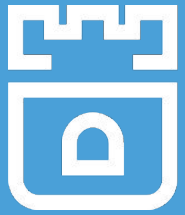


Ceph CSI Driver



- Thinly provisions and mounts storage
  - Ceph RBD, CephFS, NFS
  - Volume expansion
- Topology aware
  - Read from the OSD nearest the client
- Snapshots, clones
- Ephemeral volumes
- Group snapshots (coming soon)

CSI Addons



- RBD Mirroring
- Encryption Key rotation
- Reclaim space
- Network fencing: Handle offline nodes
- Volume replication

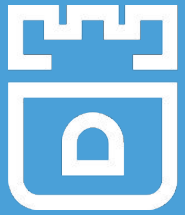
Object Storage Provisioning

- Container Object Storage Interface (COSI)
  - Experimental
- Object Bucket Claim (OBC)
  - Similar pattern to a Persistent Volume Claim (PVC)
  - The operator creates a bucket
  - Give access via a K8s Secret

Cluster Maintenance

- All nodes can be taken down in one entire failure domain (AZ) with no downtime and all data remains available for both reads/writes
- Pod Disruption Budgets (PDBs) ensure storage remains available during node maintenance
- Rook upgrades ensure only one failure domain is affected (no downtime)
- If entire cluster goes down, data is safe as long as at least one failure domain comes back online

Kubectl plugin



- CLI tool for troubleshooting
  - Cluster status
  - DR status
- Management operations that do not fit typical desired-state patterns
  - Ceph commands (toolbox replacement)
  - Restore mon quorum
  - Remove OSDs
  - Mon and OSD maintenance operations
  - Restore CRs accidentally in deleted state
- What would you like to see automated?

Rook Project

Rook Community

- Community is important to us!
  - Monthly community meeting
- Open Source (Apache 2.0)
- Major contributors from four companies
  - Clyso, Cybozu, IBM/Red Hat, Upbound
- 400+ contributors to the Github project
- 340M container downloads

CNCF Graduated

- Rook is a CNCF graduated project for four years now!
  - Sandbox: January 2018
  - Incubation: September 2018
  - Graduation: October 2020

Stability is number 1 focus

- Six years since declared stable for production
- Many upstream users running in production
- Many downstream deployments running in production
- Runs on anything
  - NVMe disks with InfiniBand networking
  - 100 USB drives on spare desktops

Release Cycle

- Minor releases are about **every 4 months**
  - v1.15 was in August
  - v1.16 planned in December
- Regular patch releases
  - Biweekly unless there is a critical need

Project Updates

v1.15 Features

- Ceph v19 (Squid) support
- Support more OSD day-2 parameter changes
- Object store Keystone+Swift integration for OpenStack
- Ceph-CSI operator v1alpha1
- Begin deprecating “holder” pods for multus networking
- Object store pool placements

v1.16+ Roadmap

- Overall theme: supporting users with complex needs
- Mirroring for RADOS block pool namespaces (Disast. Recov.)
- More focus on object storage
  - S3 storage classes (different from K8s StorageClass)
  - Auditing/logging S3 accesses
- Holder pods deprecated for multus networking
- Removed support for Ceph Quincy (v17), at end of life

Application Disaster Recovery

Disaster Recovery goals & objectives

Disaster recovery is the ability to recover and continue business critical applications from natural or human created disasters. Recovery goals are usually expressed as Recovery Point Objective (RPO) and Recovery Time Objective (RTO).

- RPO is a measure of how frequently you take backups or snapshots of persistent data. In practice, the RPO indicates the amount of data that will be lost or need to be re-entered after an outage. For synchronous storage replication the amount of data loss will be zero. For asynchronous storage replication the amount of data loss is related to the replication interval (e.g., 5 minutes of maximum data loss).
- RTO is the amount of downtime an application or service can tolerate. The RTO answers the question, “How long can it take for our system to recover after we were notified of a business disruption?”.

Rook Application Disaster Recovery

RBD mirroring is an asynchronous replication of RBD images between multiple Ceph clusters either journal-based or snapshot-based.

- RBD Mirroring CRD
  - CephRBDMirror
    - Creates rbd-mirror daemon
- Volume Replication CRDs
  - VolumeReplicaton
  - VolumeReplicationClass
    - Provides extended APIs for storage disaster recovery

Rook Application Disaster Recovery

Ceph filesystem mirroring is a process of asynchronous replication of snapshots to a remote CephFS file system.

- Ceph Filesystem Mirroring
  - Enable mirroring in the CephFilesystem
    - Configure peers, snapshotSchedules, snapshotRetention
- Filesystem Mirroring CRD
  - CephFilesystemMirror
    - Creates cephfs-mirror daemon
- ***Filesystem Mirroring is experimental at this time.***

Asynchronous DR Failover and Failback

Rook comes with the volume replication support, which allows users to perform disaster recovery and planned migration of applications.

- Application Failover (disaster recovery)
  - Communication with Primary site cluster where application is deployed is not required.
  - The Volume Replication operator automatically sends request to forcefully mark the RBD image as primary on the Secondary site cluster.
- Application Failback (planned migration)
  - Requires both Primary and Secondary sites are online and healthy.
  - Application scaled down so RPO=0 after recovery on alternate cluster.

Open Cluster Management

Open Cluster Management is a community-driven project focused on multi-cluster and multicloud scenarios for Kubernetes apps.

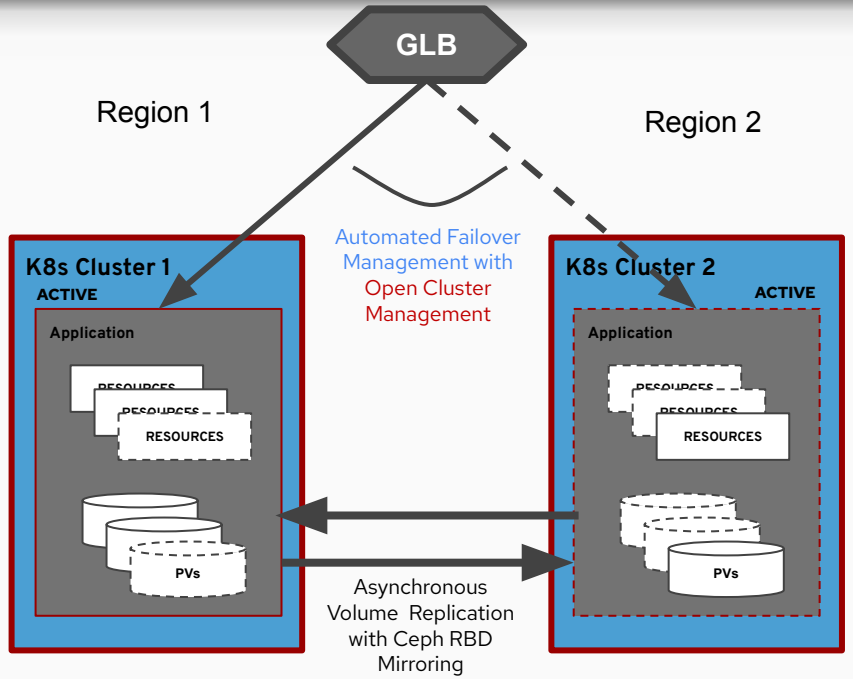
- Provides:
  - Cluster registry, Work distribution, Content placement, Vendor neutral APIs
- Leveraged for:
  - Cluster configuration
  - Application lifecycle management
  - Application placement using OCM Placement CRDs, which determine cluster(s) to deploy the application to
- Upstream Project:
  - https://github.com/open-cluster-management-io

RamenDR

Kubernetes orchestrator that provides "Instant Cloud-Native Workload Recovery and Relocation Across Kubernetes Clusters".

- Orchestrates workload placement and PVC replication across k8s clusters
  - Enhances OCM Placement scheduling for DR workflows
  - Groups PVCs in an application and orchestrates their replication, leveraging VolumeReplication and VolumeReplicationClass
- Ramen CRDs:
  - DRPolicy, DRClusters, DRPlacementControl are used on the hub cluster
    VolumeReplicationGroup is on managed cluster and used for DR actions
- Upstream Project:
  - https://github.com/RamenDR/ramen

Regional-DR (RDR) Architecture

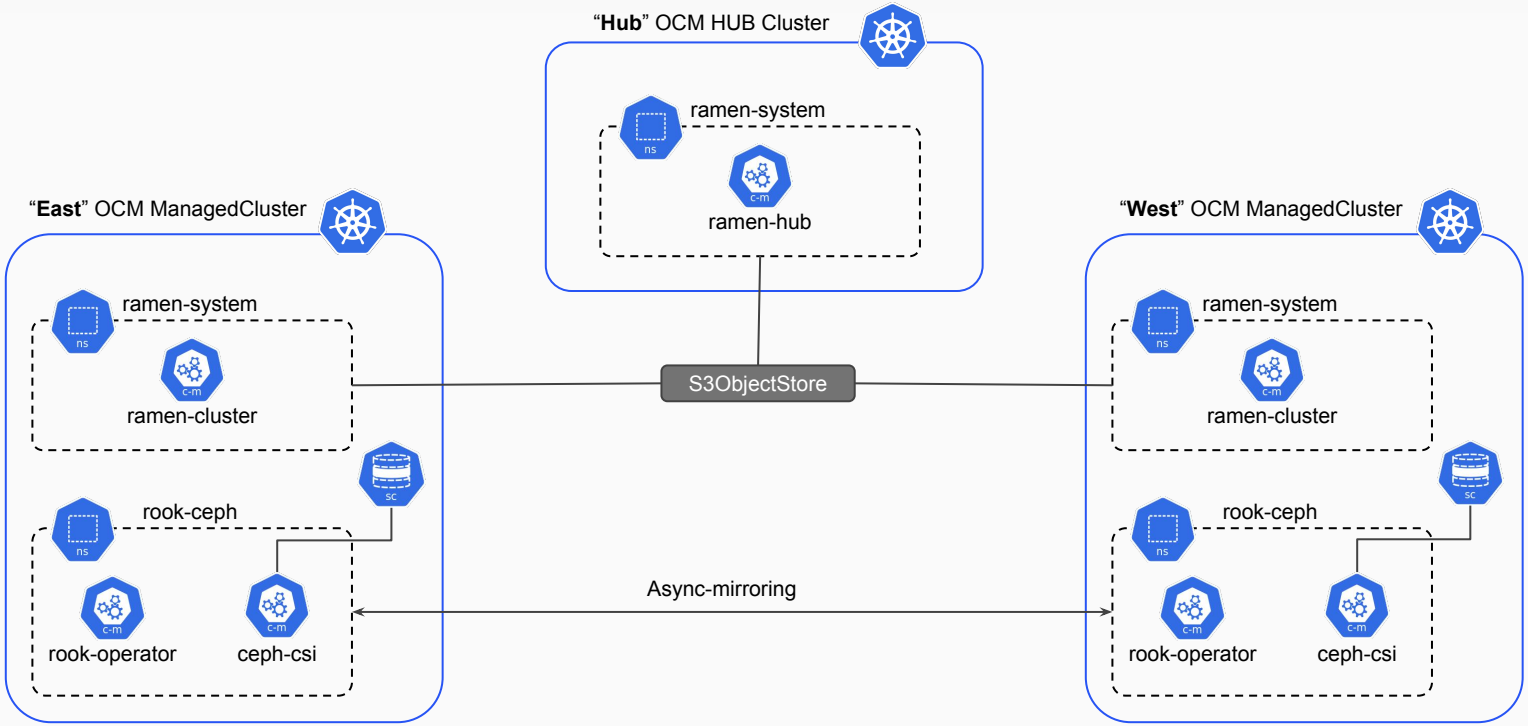


RPO – Mins
RTO – Mins

- Asynchronous Volume Replication => low RPO
  - Cross cluster replication of block volumes with replication intervals as low as 1 min defined in VolumeReplicationClass(s) and used in VolumeReplication CRs..
  - Ramen DR operators synchronizes both volume persistent data and kubernetes metadata for PVs
  - ***No distance limitations between peer clusters***
- Open Cluster Management (OCM) Failover Management => low RTO
  - OCM and Ramen DR operators enables failover and failback automation at application granularity
  - Both clusters remain active with Apps distributed and protected by the alternate cluster

Protection against Geographic Scale Disasters

Ramen, OCM & Rook Setup



RamenDR drenv testing tool

- Required resources:
  - Machine with 8 CPU and 20 GB mem
  - Machine with at least 100 GB of free space
- Required packages and tooling:
  - @virtualization group (nested virtualization enabled), minikube, kubectl, podman, cluteradm, and other tools (see the user guide)
- Regional-DR test env can create with one command:
  - **drenv start envs/regional-dr.yaml**
  - Creates minikube hub and 2 managed clusters for testing Regional DR using rbd mirroring.
- For more info see:
  - https://github.com/RamenDR/ramen/blob/main/docs/user-quick-start.md

KubeCon | CloudNativeCon

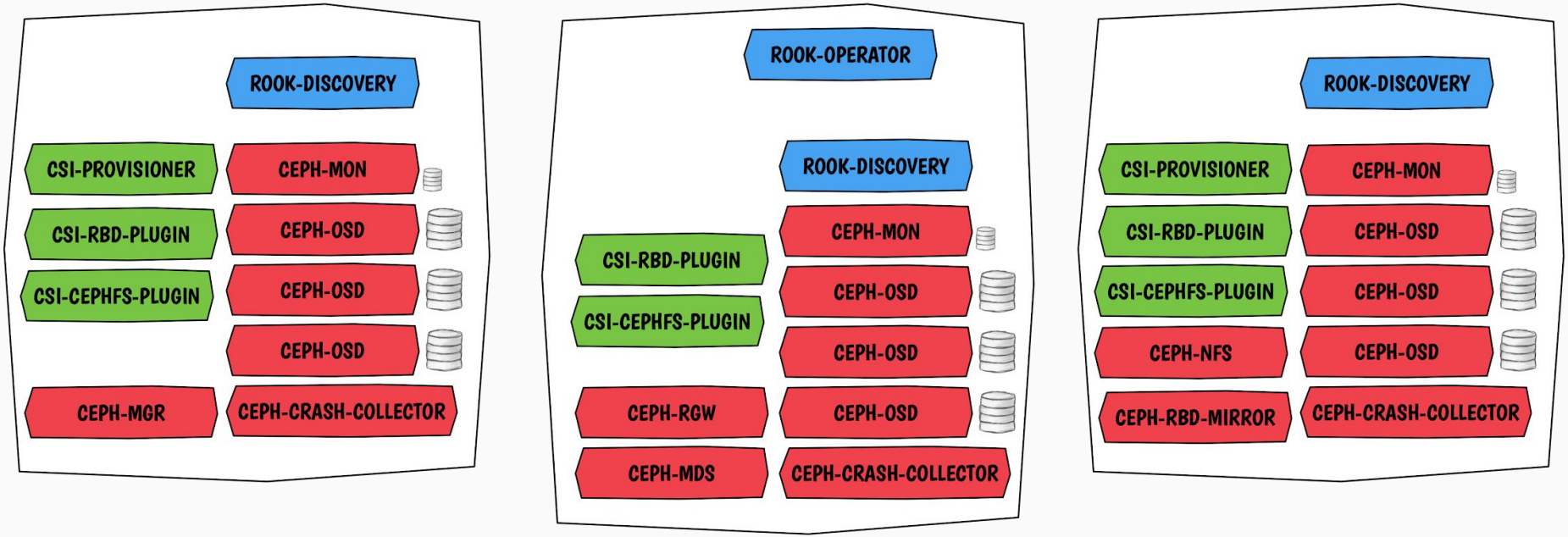
North America 2024

Thank you!

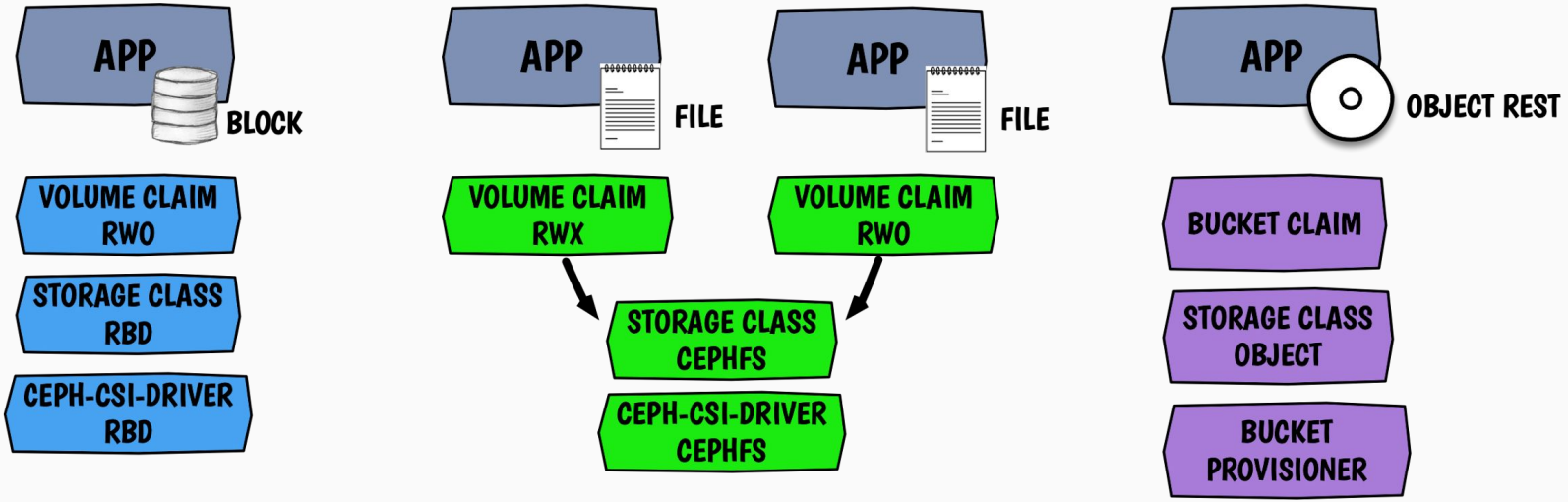
Website and Docs	https://rook.io/
Slack	https://slack.rook.io/
Twitter	@rook_io

Appendix

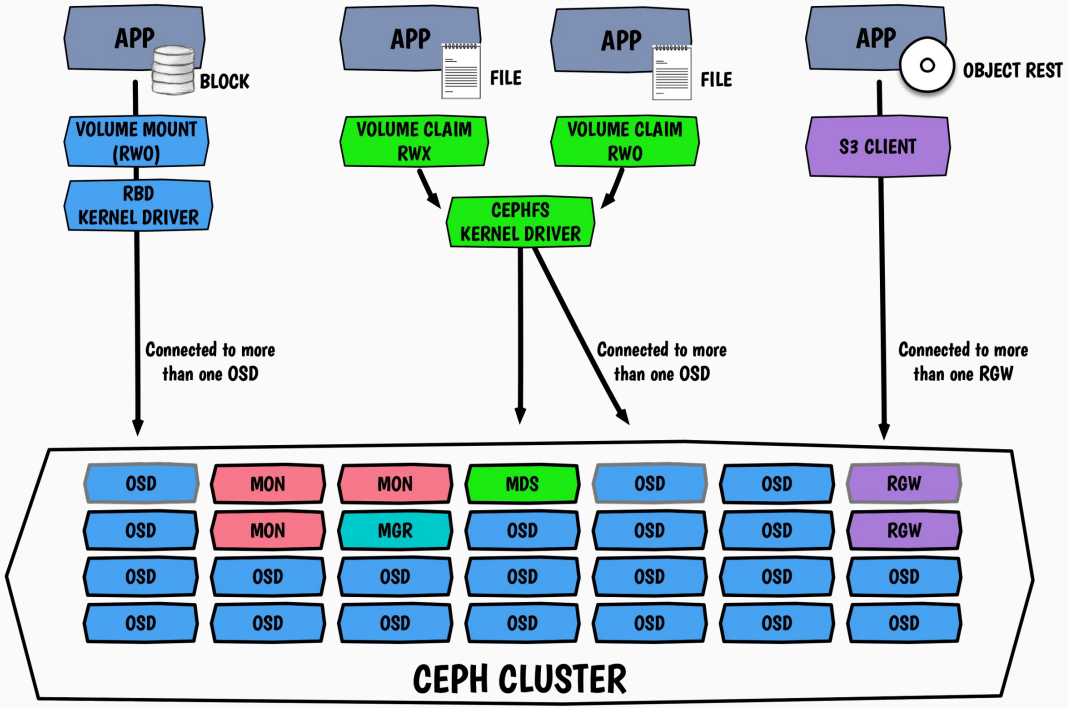
Rook Pods


ROOK-DISCOVERY
CSI-PROVISIONER
CEPH-MON
CSI-RBD-PLUGIN
CEPH-OSD
CSI-CEPHFS-PLUGIN
CEPH-OSD
CEPH-OSD
CEPH-MGR
CEPH-CRASH-COLLECTOR
ROOK-OPERATOR
ROOK-DISCOVERY
CSI-RBD-PLUGIN
CEPH-MON
CSI-CEPHFS-PLUGIN
CEPH-OSD
CEPH-OSD
CEPH-RGW
CEPH-OSD
CEPH-MDS
CEPH-CRASH-COLLECTOR
ROOK-DISCOVERY
CSI-PROVISIONER
CEPH-MON
CSI-RBD-PLUGIN
CEPH-OSD
CSI-CEPHFS-PLUGIN
CEPH-OSD
CEPH-NFS
CEPH-OSD
CEPH-RBD-MIRROR
CEPH-CRASH-COLLECTOR

Layer 2: CSI Provisioning



Layer 3: Ceph Data Path



Rook Application Disaster Recovery

To achieve RBD Mirroring, csi-omap-generator and csi-addons containers need to be deployed in the RBD provisioner pods, which are not enabled by default.

- **Volume Replication Operator**: Volume Replication Operator is a kubernetes operator that provides common and reusable APIs for storage disaster recovery. The volume replication operation is supported by the CSIAddons.
- **Omap Generator**: Omap generator is a sidecar container that when deployed with the CSI provisioner pod, generates the internal CSI omaps between the PV and the RBD image.
- Edit the rook-ceph-operator-config configmap to enable these containers.