

# ROOK DEEP DIVE CEPH

Sébastien Han  
KubeCon - 21 Nov 2019



CEPH?

*Ceph is an open source distributed storage software-defined solution that allows you to consume your data through several interfaces such as object, block and file.*

APP



HOST/VM



CLIENT



RGW

A web services gateway  
for object storage,  
compatible with S3 and  
Swift

RBD

A reliable,  
fully-distributed block  
device with cloud  
platform integration

CEPHFS

A distributed file  
system with POSIX  
semantics and  
scale-out metadata  
management

LIBRADOS

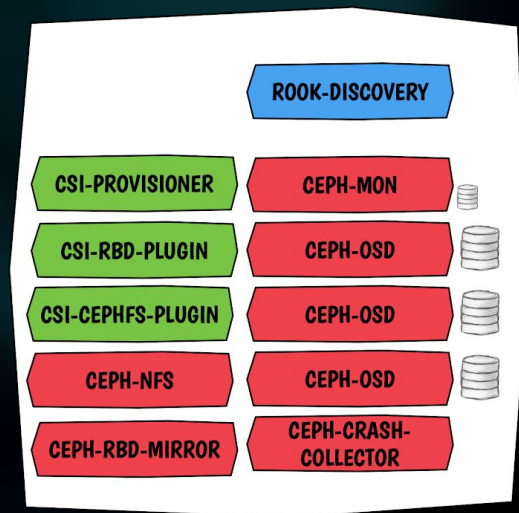
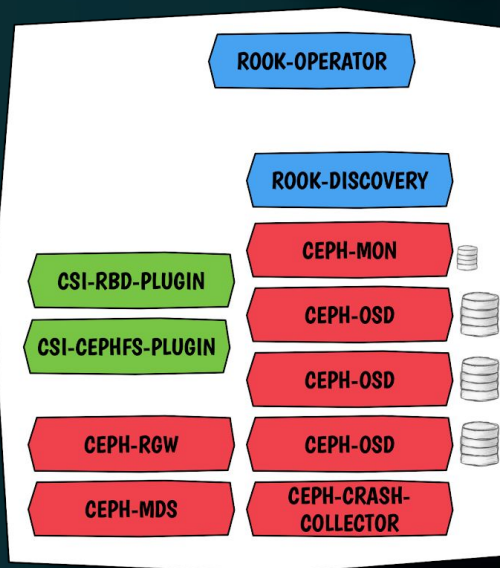
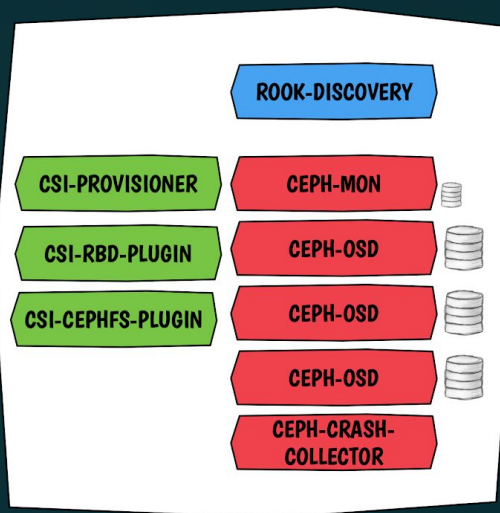
A library allowing apps to directly access RADOS (C, C++, Java, Python, Ruby, PHP)

RADOS

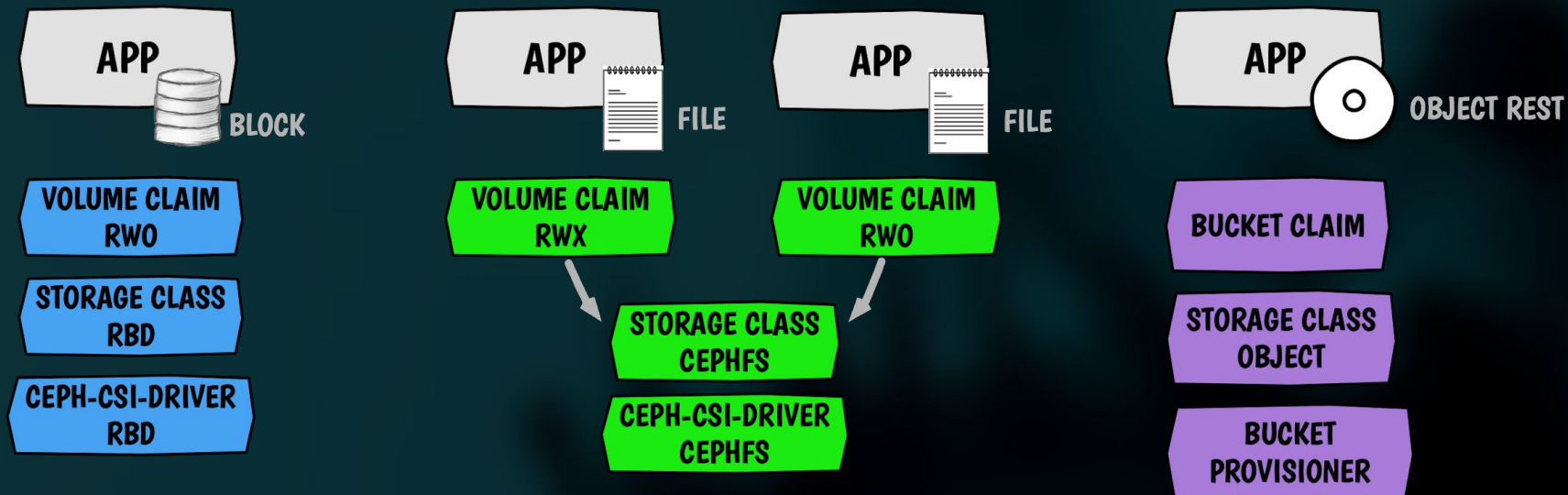
A software-based, reliable, autonomous, distributed object store comprised of self-healing, self-managing, intelligent storage nodes and lightweight monitors

# ROOK ARCHITECTURE

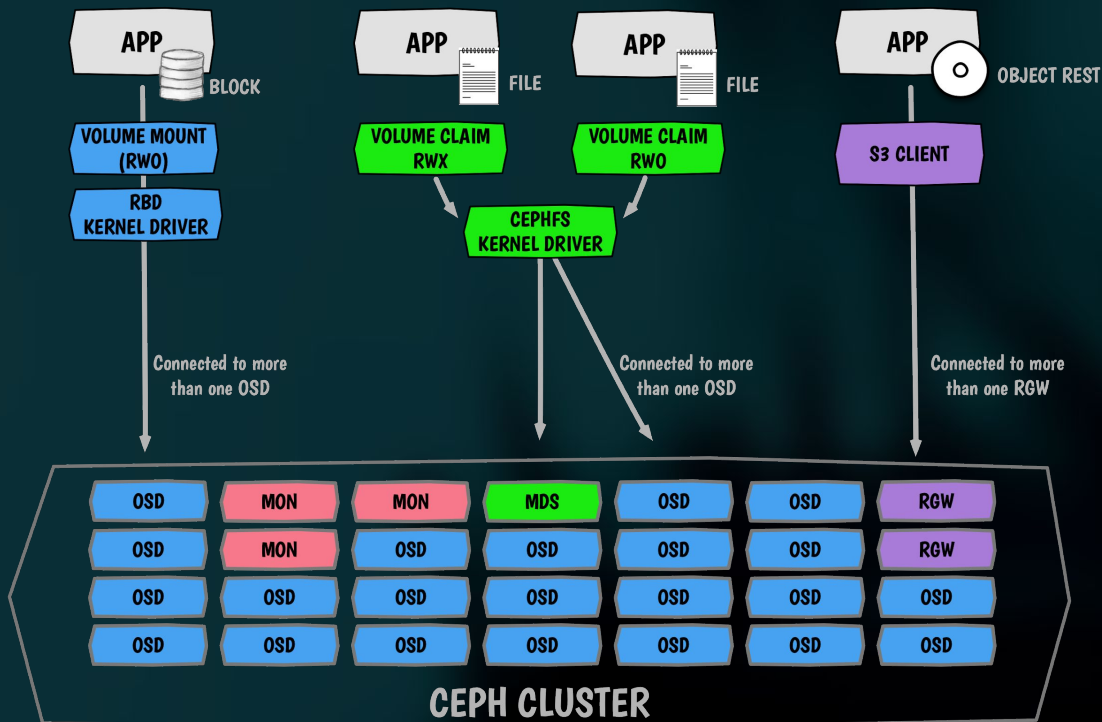
# Rook Components: Pods



# Application Storage: Provisioning



# Application Storage: Data Path

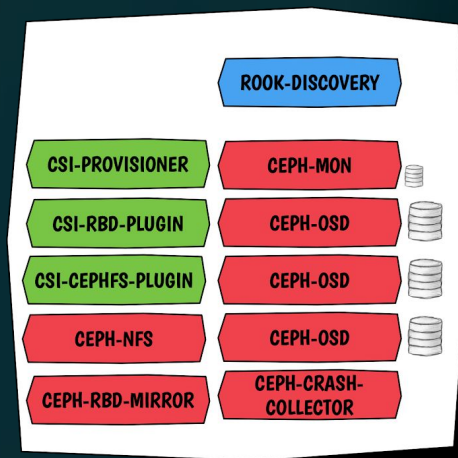




# ROOK NEW FEATURES

# Configuration for Cloud Environments

- Ceph Monitors and OSDs running on PVCs
- Ease deployments in the Cloud
- Makes storage more portable in Cloud environments



# Ceph CSI Driver

- Ceph CSI Driver is stable and deployed by default with Rook 1.1
  - Dynamic provisioning of RWO/RWX/ROX (RBD)
  - Dynamic provisioning of RWO/RWX/ROX (CephFS)
- Snapshots are still alpha in the CSI spec
- Flex driver is still supported but will be deprecated once CSI reaches feature parity

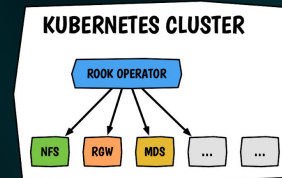
# Object Bucket Provisioning

- Define a Storage Class for object storage
- Create an “object bucket claim”
  - The operator creates a bucket when requested
  - Similar pattern to a Persistent Volume Claim (PVC)

# External Cluster Connection

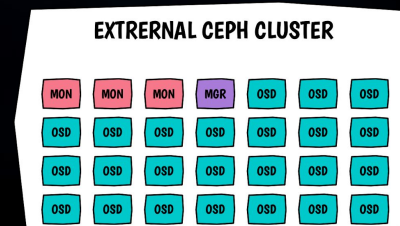
Connect to a Ceph cluster that you've configured separately from Kubernetes

- Inject the following in Kubernetes:
  - Monitors list
  - Admin keyring
  - Cluster FSID
- Create the cluster-external CR



```
kind: CephCluster
spec:
  external:
    enabled: true
  name: rook-ceph-external
```

Bonus: you can bootstrap  
Ceph stateless resources in Kubernetes!



# Cluster Topology Awareness

- Monitors will be spread across zones
- OSD's CRUSH hierarchy will be automatically populated based on node labels
- Rook labels: `topology.rook.io/<level>`
  - chassis, rack, row, pdu, pod, room, datacenter
- K8s labels: `failure-domain.beta.kubernetes.io/<level>`
  - zone, region

```
sh-4.2# ceph osd tree
ID CLASS WEIGHT TYPE NAME STATUS REWEIGHT PRI-AFF
-1 0.02637 root default
-5 0.02637 region us-east-1
-10 0.00879 zone us-east-1a
-9 0.00879 host set1-1-data-rlr88
0 ssd 0.00879 osd.0 up 1.00000 1.00000
-4 0.00879 zone us-east-1b
-3 0.00879 host set1-2-data-jw6db
1 ssd 0.00879 osd.1 up 1.00000 1.00000
-14 0.00879 zone us-east-1c
-13 0.00879 host set1-0-data-qvvq4
2 ssd 0.00879 osd.2 up 1.00000 1.00000
```



# Configure Ceph Manager modules

- Enable ceph-mgr modules from the cluster CR

```
apiVersion: ceph.rook.io/v1
kind: CephCluster
metadata:
  name: rook-ceph
  namespace: rook-ceph
spec:
  mgr:
    modules:
      - name: pg_autoscaler
        enabled: true
```

# ROOK v1.2 FEATURES

mid December



# Collect Crash Dumps

- New ceph-crashcollector controller
- Runs on nodes where Ceph daemons exist
- When a Ceph daemon crashes, it puts a crash log in `/var/lib/ceph/crash`
- `ceph-crash` scraps `/var/lib/ceph/crash` and sends the crash in the `ceph-mgr`
- Crashes are centralized and can be accessed via the Ceph CLI

NAME	READY	STATUS	RESTARTS	AGE
csi-cephfsplugin-provisioner-7c494c799-6pjh6	3/3	Running	1	108m
csi-cephfsplugin-provisioner-7c494c799-cbd6m	3/3	Running	0	108m
csi-cephfsplugin-pvjjb	3/3	Running	0	16d
csi-rbdplugin-9rwz6	3/3	Running	0	16d
csi-rbdplugin-provisioner-667b98cdf-d82q5	4/4	Running	1	108m
csi-rbdplugin-provisioner-667b98cdf-g26c8	4/4	Running	0	108m
rook-ceph-crashcollector-minikube-574858b99c-sw2g9	1/1	Running	0	73m
rook-ceph-mgr-a-c9bdd499-rc4xm	1/1	Running	0	73m
rook-ceph-mon-a-647b468b9f-dpwlw	1/1	Running	0	73m
rook-ceph-mon-b-59b75795d6-6gggc	1/1	Running	0	73m
rook-ceph-mon-c-54bc9f44cc-97lxl	1/1	Running	0	73m
rook-ceph-operator-5fd85794d7-5867p	1/1	Running	0	74m
rook-ceph-osd-0-6df4d9bfcc-jj449	1/1	Running	0	71m
rook-ceph-osd-1-f65c86f7f-grtwq	1/1	Running	0	71m
rook-ceph-osd-2-67f9cc6cdd-ftlqx	1/1	Running	0	71m
rook-ceph-osd-prepare-minikube-c4vs5	0/1	Completed	0	72m
rook-discover-bsn55				



# Priority Classes

- Kubernetes feature stable since 1.14
- Add support for priority classes defined in the CephCluster CR
- Priority indicates the importance of a Pod relative to other Pods
- Ceph monitors, Ceph Manager and OSDs should have a high priority
- Ceph MDS / rbd-mirror / Ceph Rgw / Ceph NFS can have a lower priority if redundant and spread across hosts

# Ceph-CSI v2.0

- Topology aware provisioning - data affinity/locality
- Extra level of locking for RWO volumes
- RBD-NBD support
  - Uses librbd, then provides more image features
- Manage RBD mirroring via a StorageClass parameter
- Snapshot support for RBD and CephFS
- Restore a snapshot to a new PVC for CephFS

# DEMO CLUSTER ON PVC

# Thanks!

seb@redhat.com



**BONUS**

# Collect Crash Dumps flow

```
[leseb@tarox~/] kubectl -n rook-ceph exec -ti rook-ceph-crashcollector-minikube-574858b99c-zvg4z bash

[root@rook-ceph-crashcollector-minikube-574858b99c-zvg4z /]# ceph crash ls

[root@rook-ceph-osd-2-7644f99695-cljzh /]# pidof ceph-osd
13258 5415 5397

[root@rook-ceph-osd-2-7644f99695-cljzh /]# kill -SIGABRT 13258

[root@rook-ceph-osd-2-7644f99695-cljzh /]# ls /var/lib/ceph/crash/
2019-11-12_12:56:51.404109Z_39f060e1-776d-4605-8f28-85c97e53de96      posted

... wait maximum 10 min (ceph-crash scraps every 10 minutes)
... the container will exit

[root@rook-ceph-osd-2-7644f99695-cljzh /]# exit

[leseb@tarox~/] kubectl -n rook-ceph exec -ti rook-ceph-crashcollector-minikube-574858b99c-zvg4z bash

[root@rook-ceph-crashcollector-minikube-574858b99c-zvg4z /]# ceph crash ls
2019-11-12_12:56:51.404109Z_39f060e1-776d-4605-8f28-85c97e53de96  osd.1

[root@rook-ceph-crashcollector-minikube-574858b99c-zvg4z /]# ls /var/lib/ceph/crash/

posted

[root@rook-ceph-crashcollector-minikube-574858b99c-zvg4z /]# ls /var/lib/ceph/crash/posted/
2019-11-12_12:56:51.404109Z_39f060e1-776d-4605-8f28-85c97e53de96

[root@rook-ceph-crashcollector-minikube-574858b99c-zvg4z /]# tail /var/lib/ceph/crash/posted/2019-11-
12_12\.:56\.:51.404109Z_39f060e1-776d-4605-8f28-85c97e53de96/Log
 1/ 5 mgr
 1/ 5 mgrc
 1/ 5 dpdk
 1/ 5 eventtrace
-2/-2 (syslog threshold)
-1/-1 (stderr threshold)
max_recent      10000
max_new         10000
log_file /var/lib/ceph/crash/2019-11-12_12:56:51.404109Z_39f060e1-776d-4605-8f28-85c97e53de96/log
--- end dump of recent events ---
```

