Red Hat’s Technology Roadmap
Red Hat Ceph Storage versions

**RHCS 4**
- **Q1**
  - Upstream: Nautilus
  - Platform: RHEL 7 & 8 RPM & Container
  - EoL: 1/2023

**RHCS 5**
- **Q3**
  - Upstream: Octopus
  - Platforms: RHEL 8 container
  - Beta 1: 6/2020

**Regular 6 week Z-stream updates**

- Vault support
- OSP 16.1 support
Data Reduction

- **2015**
  - RHCS 1.2
  - RGW EC

- **2018**
  - RHCS 3.1
  - RBD EC preview

- **2020 Q1**
  - RHCS 4
  - RBD EC
  - CephFS EC preview

- **2021**
  - Distributed deduplication (Pacific)

- **2017**
  - RHCS 3.0
  - RGW Inline Compression

- **2019**
  - RHCS 3.3
  - Bluestore Compression

- **2020 Q3**
  - RHCS 5
  - CephFS EC

**CEPH STORAGE TECHNOLOGY ROADMAP**

**STRA TEGIC ROADMAP – SUBJECT TO CHANGE**
Security

- **2015**
  - RHCS 1.2
  - Dmcrypt

- **2016**
  - RHCS 1.3.2
  - SELinux policies (1.3.2)
  - PIE (2.0)

- **2017**
  - RHCS 3.0
  - RGW inline encryption

- **2018**
  - RHCS 3.2
  - Security Guide

- **2020 Q1**
  - RHCS 4
  - FIPS-140
  - Messenger v2 encryption
  - Namespaces

- **2020 Q3**
  - RHCS 5
  - Support for NVMe self-encrypting drive key management in MON (TP)
  - SSE-KMS Support (Barbican, Vault and KMfP)
  - SSE-S3 support Server Managed data encryption (Tech Preview)
  - S3 STS (IAM identity interop)

- **2021 Q1**
  - RHCS 6
  - S3 WORM (TP)
CephFS

**2017**
- RHCS 3
  - Support begins

**2020 Q1**
- OCS 4.2
- RHCS 4
  - Kubernetes and Rook
    - PV RWX
    - CSI driver
  - 10 Developers

**2021**
- RHCS 6
  - SMB in Tech Preview
    - Scale by user

**2018**
- Key Customers
  - [chipmaker]
  - Monash

**2020 Q3**
- OCS 4.6
  - Snapshot clones

**2020 Q3**
- RHCS 5
  - Scale to 10000 PVs turning
  - NFS
  - Key Customers: (round 2)
    - [chipmaker]
    - [major hardware OEM]
Business Continuity

- **2015**
  - RHCS 1.2
  - RHCS 2.0
  - Cinder
  - RBD Snapshots
  - Stretch clusters

- **2016**
  - RHCS 2.0
  - RBD Mirror
  - RGW Multisite

- **2017**
  - RHCS 3.0
  - RBD Trash
  - Snapshot provisioning

- **2019 Q3**
  - RHCS 3.3
  - Backup ISV certifications

- **2020 Q1**
  - RHCS 4
  - RGW Archive Zone (TP)

- **2020 Q3**
  - RHCS 5
  - OCS 4.6
  - RBD mirror
  - Snapshot mode
  - CephFS snapshot clones
  - Stretch cluster mode

- **2021**
  - CephFS Geo Rep (Pacific)

CEPH STORAGE TECHNOLOGY ROADMAP
Performance & Scale

- **2015**
  - RHCS 1.3
    - "Petabyte release"
    - Bucket sharding
    - Scrubbing window
    - Alloc and cache hinting

- **2016**
  - RHCS 1.3.2
    - First support for DBMS
    - Thread cache tuning
    - 1.8 PB deployed in one hour (1040 OSDs)
  - RHCS 1.3.3
    - 10PB cluster
  - RHCS 2.0
    - 1.8 PB deployed in one hour (1040 OSDs)
    - 10PB cluster

- **2017**
  - RHCS 3.0
    - Consistent IO on rebalance
  - RHCS 3.1
    - First support for DBMS
  - RHCS 3.2
    - RocksDB journaling

- **2018**
  - RHCS 3.3
    - 2X performance
    - 1 billion objects
    - Bluestore
    - Beast.ASIO
    - 12TB drive support

- **2019**
  - RHCS 3.2
    - 2X performance
  - RHCS 3.3
    - 1 billion objects
    - Bluestore
    - Beast.ASIO
    - 12TB drive support

- **2019-20**
  - OCS 4.2
    - Consistent IO on rebalance
  - RHCS 4.0
    - Consistent IO on recovery
  - RHCS 4.2
    - Async Messenger
    - Consistent IO on recovery

- **2020**
  - RHCS 4.1
    - 2X performance
    - 1 billion objects
  - OCS 4.5
    - Bluestore v.2
  - RHCS 5
    - New LibRBD cache
    - 10 billion objects
    - 20,000 PVs turning

- **2021**
  - RHCS 6
    - 5,000 PVs turning
    - Async Messenger
    - Consistent IO on recovery
  - OCS 4.2
    - SeaStore (TP)
  - RHCS 6
    - Crimson OSD (TP)

---

STRATEGIC ROADMAP – SUBJECT TO CHANGE
Object Storage

- Backup ISV Certifications
- Object granular compression & encryption (SSE-C)
- Dynamic bucket index sharding

2017
RHCS 3.0

2019 Q3
RHCS 3.3

2020 Q1
RHCS 4

2020 Q3
RHCS 5

2021

- Bucket notifications
- Vault integration
- STS support
- RGW Archive Zone (TP)

- Server managed encryption (SSE-S3)
- Policy based tiering to public cloud
- Object lock (TP)
- S3 Worm (TP)

- KMIP support for key management (SSE-KMS)
- Multi-site scalability and usability enhancements

- New RGW Web server
- Performance and sizing guide
Ceph’s Community Roadmap
Stable, named release every 9 → 12 months
Backports for 2 releases
Upgrade up to 2 releases at a time
  (e.g., Luminous → Nautilus, Mimic → Octopus)
WHAT'S NEW IN CEPH

OCTOPUS
FIVE THEMES

Usability

Quality

Performance

Multi-site

Ecosystem
ORCHESTRATOR API

- End-to-end management experience
- mgr API to interface with deployment tool
  - Rook (deploy+manage via Kubernetes)
  - cephadm (deploy+manage via ssh)
- Expose provisioning functions to CLI, GUI
  - Create, destroy, start, stop daemons
  - Blink disk lights
- Pave way for cleanup of docs.ceph.com
- Automated upgrades
CEPHADM

● Easy
  ○ Simple ‘bootstrap’ to create new cluster
  ○ Most services provisioned automatically
    ■ Mon, mgr, monitoring for dashboard
  ○ Easy mode for OSDs
    ■ --all-available-devices
  ○ Everything works out-of-the-box

● Minimal dependencies
  ○ Systemd
  ○ Container runtime (podman or docker)
  ○ Python 3
  ○ LVM

● Container based
  ○ Single build artifact
  ○ Works consistently on any host OS
  ○ Easier registry-based experience
  ○ Easily enable disconnected environments

● Robust
  ○ “Declarative” management style
  ○ Automatic or controlled placement of daemons
  ○ Automated upgrades

● Fully replace ceph-ansible, ceph-deploy, puppet-ceph, DeepSea, etc.
Robust management GUI for cluster operations
  - All core Ceph services: object, block, file
  - OSD creation with DriveGroups
    - Filter by host, device properties (size/type/model)
  - Some multisite capabilities
  - Some legacy protocol support (NFS, SMB, iSCSI)

Targets “storage admins” as well as experienced Ceph power users
  - Storage management (creating pools, volumes, etc.)
  - Robust monitoring (high-level, troubleshooting, and diagnostics)
  - Cluster infrastructure management (provisioning hosts, drives, etc.)

Integrations
  - External authentication (SAML, OpenID)
  - Roles
  - External Prometheus for metrics
MISC RADOS USABILITY

- Hands-off defaults
  - PG autoscaler on by default
  - Balancer on by default
- Quality internal health alerts
- Health alert muting
  - TTL on mutes
  - Auth-unmute when alerts change, increase in severity
- Ongoing simplification and cleanup of administration/operations
- ‘ceph tell …’ and ‘ceph daemon …’ unification
  - Consistent and expanded command set via either (over-the-wire or local unix socket)
FIVE THEMES

Usability

Quality

Performance

Multi-site

Ecosystem
• Partial object recovery
  ○ Re-sync only modified portion of large object after small overwrite

• Improved prioritization of PG recovery
  ○ Focus on PGs that are inactive
  ○ Better handling of planning when both primary and replica OSDs need to do work

• Snapshot trimming improvements
  ○ Eliminate metadata in OSD map that (previously) would grow with cluster age
  ○ Simpler code; occasional scrubbing

• Close “read hole”
  ○ Eliminate very rare case where partitioned OSD + client could serve a stale read
TELEMETRY AND CRASH REPORTS

- Opt-in
  - Require re-opt-in if telemetry content expanded
  - Explicitly acknowledge data sharing license
- Telemetry channels
  - **basic** - cluster size, version, etc.
  - **ident** - contact info (off by default)
  - **crash** - anonymized crash metadata
  - **device** - device health (SMART) data
- Dashboard nag to enable
- Public dashboard launch Real Soon Now

- Backend tools to summarize, query, browse telemetry data
- Initial focus on crash reports
  - Identify crash signatures by stack trace (or other key properties)
  - Correlate crashes with ceph version or other properties
- Improved device failure prediction model
  - Predict error rate instead of binary failed/not-failed or life expectancy
  - Evaluating value of some vendor-specific data
RADOS: BLUESTORE

- **RocksDB improvements for metadata storage**
  - Prefetching support during compaction, key iteration, object enumeration
  - Selective use of RangeDelete

- **Improved cache management**
  - Better use of cache memory
  - New inline trimming behavior (big performance bump!)

- **Per-pool omap utilization tracking**
  - To match Nautilus’ per-pool data usage (and compression) stats
MISC PERFORMANCE

**RGW**

- More async refactoring
  - Efforts started with Beast frontend a few releases ago
  - Goal is end-to-end boost::asio request processing
- Avoid omap where unnecessary
  - FIFO queues for garbage collection
  - Selective use of DeleteRange

**RBD**

- (lib)rbd cache replacement
  - Simpler IO batching, writearound cache
  - General cleanup of IO path code
  - Significant (2x+) improvement for small IO
    - e.g., ~18kIOPS → 70kIOPS for 4KiB writes
FIVE THEMES

Usability

Quality

Performance

Multi-site

Ecosystem
Today: RBD mirroring provides async replication to another cluster

- Point-in-time ("crash") consistency
- Perfect for disaster recovery
- Managed on per-pool or per-image basis

rbd-nbd runner improvements to drive multiple images from one instance

Vastly-simplified setup procedure

- One command on each cluster; copy+paste string blob

New: snapshot-based mirroring mode

- (Just like CephFS)
- Same rbd-mirror daemon, same overall infrastructure/architecture
- Will work with kernel RBD
  - (RBD mirroring today requires librbd, rbd-nbd, or similar)
Current multi-site supports
  - Federate multiple sites
  - Global bucket/user namespace
  - Async data replication at site/zone granularity

Octopus adds bucket-granularity replication
  - Finer grained control
  - Currently experimental until more testing is in place
NEW WITH CEPH-CSI AND ROOK

- Much investment in ceph-csi
  - RWO and RWX support via RBD and/or CephFS
  - Snapshots, clones, and so on
- Rook
  - Turn-key ceph-csi by default
  - Dynamic bucket provisioning
    - ObjectBucketClaim
  - Run mons or OSDs on top of other PVs
  - Upgrade improvements
    - Wait for healthy between steps
    - Pod disruption budgets
  - Improved configuration experience
WHAT’S COMING IN CEPH PACIFIC
FIVE THEMES

Usability

Quality  Performance

Multi-site  Ecosystem
ORCHESTRATION

- Cephadm improvements
  - Resource-aware service placement (memory, CPU)
  - Haproxy, NFS, SMB, RGW-NFS support
- Rook integration improvements
  - Provision RGW
  - Load balancer / Service management

- Dashboard integrations
  - Improved OSD workflows to replace failed disks, preview OSD creation, zap old devices
  - Add/configure daemons (mons, mgr,s RGW, NFS, SMB, iSCSI)
  - Initiate and monitor upgrades
MISC USABILITY AND FEATURES

RBD
- Expose snapshots via RGW (object)
- “Instant” clone/recover from external (RGW) image
- Improved rbd-nbd support
  - Expose kernel block device with full librbd feature set
  - Improved integration with ceph-csi for Kubernetes environments

RGW
- Deduplicated storage

CephFS
- ‘fs top’
- NFS and SMB support via orchestrator
FIVE THEMES

- Usability
- Quality
- Performance
- Multi-site
- Ecosystem
STABILITY AND ROBUSTNESS

RADOS

- Enable ‘upmap’ balancer by default
  - More precise than ‘crush-compat’ mode
  - Hands-off by default
  - Improve balancing of ‘primary’ role
- Dynamically adjust recovery priority based on load
- Automatic periodic security key rotation
- Distributed tracing framework
  - For end-to-end performance analysis

CephFS

- MultiMDS metadata scrub support
- MultiMDS metadata balancing improvements
- Multi-filesystem testing and auth management improvements
- Major version upgrade improvements
TELEMETRY

- Work continues on backend analysis of telemetry data
  - Tools for developers to use crash reports identify and prioritize bug fixes

- Adjustments in collected data
  - Adjust what data is collected for Pacific
  - Periodic backport to Octopus (we re-opt-in)
  - e.g., which orchestrator module is in use (if any)

- Drive failure prediction
  - Building improved models for predictive drive failures
  - Expanding data set via Ceph collector, standalone collector, and other data sources
FIVE THEMES

Usability
Quality
Multi-site
Performance
Ecosystem
CephFS

- Async unlink and create
  - Avoid client-MDS round-trip
  - `rm -r, tar xf, etc`
  - Support in both libcephfs and kernel
- Ceph-fuse performance
  - Take advantage of recent libfuse changes

RGW

- Data sync optimizations, sync fairness
- Sync metadata improvements
  - omap -> cls_fifo
  - Bucket index, metadata+data logs
- Ongoing async refactoring of RGW
  - Based on boost::asio
• **Sharded RocksDB**
  ○ Improve compaction performance
  ○ Reduce disk space requirements

• **In-memory cache improvements**

• **SMR**
  ○ Support for host-managed SMR HDDs
  ○ Targeting cold-stored workloads (e.g., RGW) only
Why
● Not just about how many IOPS we do...
● More about IOPS per CPU core
● Current Ceph is based on traditional multi-threaded programming model
● Context switching is too expensive when storage is almost as fast as memory
● New hardware devices coming
  ○ DIMM form-factor persistent memory
  ○ ZNS - zone-based SSDs

What
● Rewrite IO path in using Seastar
  ○ Preallocate cores
  ○ One thread per core
  ○ Explicitly shard all data structures and work over cores
  ○ No locks and no blocking
  ○ Message passing between cores
  ○ Polling for IO
● DPDK, SPDK
  ○ Kernel bypass for network and storage IO
● Goal: Working prototype for Pacific
FIVE THEMES

Usability
Quality
Performance
Multi-site
Ecosystem
CEPHFS MULTI-SITE REPLICATION

- Automate periodic snapshot + sync to remote cluster
  - Arbitrary source tree, destination in remote cluster
  - Sync snapshots via rsync
  - May support non-CephFS targets

- Discussing more sophisticated models
  - Bidirectional, loosely/eventually consistent sync
  - Simple conflict resolution behavior?
- Nodes scale up (faster, bigger)

- Clusters scale out
  - Bigger clusters within a site

- Organizations scale globally
  - Multiple sites, data centers
  - Multiple public and private clouds
  - Multiple units within an organization

- Universal, global connectivity
  - Access your data from anywhere

- API consistency
  - Write apps to a single object API (e.g., S3) regardless of which site, cloud it is deployed on

- Disaster recovery
  - Replicate object data across sites
  - Synchronously or asynchronously
  - Failover application and reattach
  - Active/passive and active/active

- Migration
  - Migrate data set between sites, tiers
  - While it is being used

- Edge scenarios (caching and buffering)
  - Cache remote bucket locally
  - Buffer new data locally
- Project Zipper
  - Internal abstractions to allow alternate storage backends (e.g., storage data in external object store)
  - Policy layer based on LUA
  - Initial target: tiering to cloud (e.g., S3)
- Dynamic reshard vs multisite support
FIVE THEMES

Usability

Quality

Performance

Multi-site

Ecosystem
- **External cluster support**
  - Provision storage volumes from an existing external Ceph cluster
  - Rook manages ceph-csi and provides the same CRDs for storage pools, object stores, volumes, etc.

- **Rook: RBD mirroring**
  - Manage RBD mirroring via CRDs
  - Investment in better rbd-nbd support to provide RBD mirroring in Kubernetes
  - New, simpler snapshot-based mirroring

- **Rook: RGW multisite**
  - Federation of multiple clusters into single namespace
  - Site-granularity replication
**Windows**
- Windows port for RBD is underway
- Lightweight kernel pass-through to librbd
- CephFS to follow (based on Dokan)

**Performance testing hardware**
- Intel test cluster: *officianalis*
- AMD / Samsung / Mellanox cluster
- High-end ARM-based system?

**ARM (aarch64)**
- Loads of new build and test hardware arriving in the lab
- CI and release builds for aarch64

**IBM Z**
- Collaboration with IBM Z team
- Build and test
WE INTEGRATE WITH CLOUD ECOSYSTEMS

openstack®
kubernetes
KVM
ROOK
Ceph is open source software!
- Mostly LGPL2.1/LGPL3

We collaborate via
- GitHub: https://github.com/ceph/ceph
- https://tracker.ceph.com/
- E-mail: dev@ceph.io
- #ceph-devel on irc.oftc.net

We meet a lot over video chat
- See schedule at http://ceph.io/contribute

We publish ready-to-use packages
- CentOS 7, Ubuntu 18.04

We work with downstream distributions
- Debian, SUSE, Ubuntu, Red Hat
Thank you