**Rook**

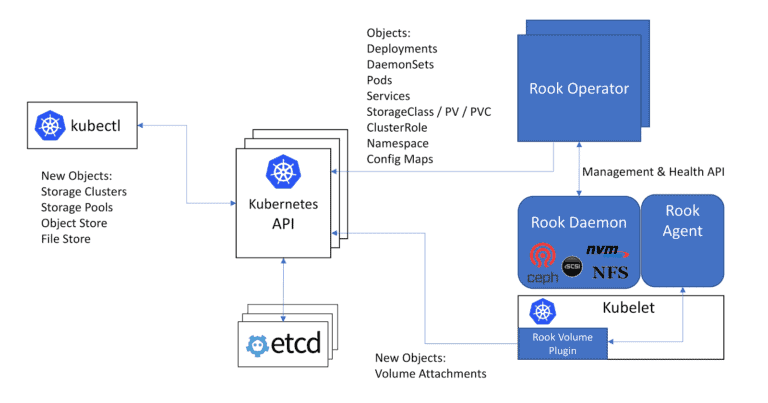
**Introduction:-**

Rook is the 15th hosted inception level project alongside Kubernetes, Prometheus, OpenTracing, Fluentd, Linkerd, gRPC, CoreDNS, containerd, rkt, CNI, Envoy, Jaeger, Notary and TUF by CNCF

Rook turns distributed storage systems into self-managing, self-scaling, self-healing storage services. It automates the tasks of a storage administrator: deployment, bootstrapping, configuration, provisioning, scaling, upgrading, migration, disaster recovery, monitoring, and resource management  
Rook uses the power of the Kubernetes platform to deliver its services via a Kubernetes Operator for Ceph.

Instead of building an entirely new storage system which requires many years to mature, Rook focuses on turning existing battle-tested storage systems like Ceph into a set of cloud-native services that run seamlessly on-top of Kubernetes. Rook integrates deeply into Kubernetes providing a seamless experience for security, policies, quotas, lifecycle management, and resource management.

They have added a concept of a storage cluster, a storage pool, an object store and a file system through rook



|  |  |
| --- | --- |
| **website** | <https://www.rook.io> |
| **Organization/foundation name** | Cloud Native Computing Foundation(CNCF) |
| **License** | Apache License 2.0 |
| **Open/Proprietary** | Open source project |
| **Source Path(if open source)** | <https://github.com/rook/rook> |
| **Brief description** | It is an open source cloud-native storage orchestrator for Kubernetes, providing the platform, framework, and support for a diverse set of storage solutions to natively integrate with cloud-native environments.  Rook is a storage system for Kubernetes which improves data storage management.  It is an open-source project that was started in September 2017.  Rook is the first storage system for Kubernetes based on Ceph, an open-source distributed object store and file system  It provides continuous data protection and it integrates with the Kubernetes API.  Rook brings File, Block and Object storage systems into the Kubernetes cluster, running them seamlessly alongside other applications and services that are consuming the storage. By doing so, the cloud-native cluster becomes self-sufficient and portable across public cloud and on-premise deployments |

**Project Summary:**

**Project Details**

**Key features**

* **Simple and reliable automated storage management**

Whichever supported storage technologies you choose, Rook ensures that resources can be deployed and managed automatically.

* **Hyper-scale or hyper-converge your storage clusters**

With Rook you can either build dedicated storage clusters or hyper-converged clusters where your apps run alongside storage.

* **Efficiently distribute and replicate data to minimize loss**

Rook efficiently distributes and replicates your data across your cluster to minimize the risk of data loss. With snapshots, cloning and versioning, no more losing sleep over your data.

* **Provision, file, block, and object storage**

Rook integrates Ceph with multiple storage presentations including object storage (compatible with S3 and swift), block storage, and POSIX-compliant shared file system.

* Manage open-source Ceph storage

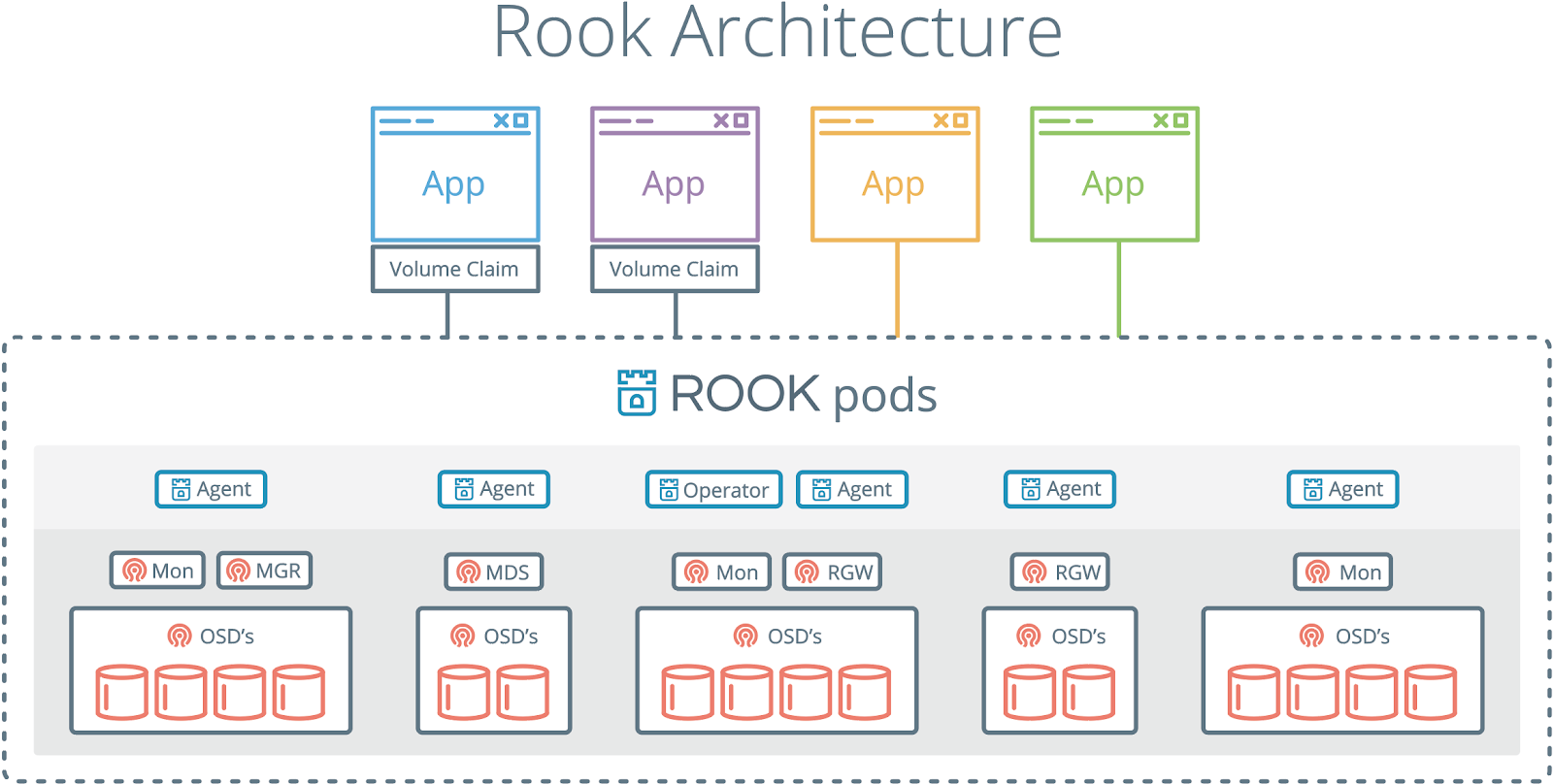
* **Easily enable elastic storage in your datacenter**

Scale is now possible in your datacenter. Get started with a few terabytes, and easily to scale up to petabytes. Simply add more nodes and Rook takes care of the rest.

* **Optimize workloads on commodity hardware**
* Runs on commodity hardware to optimize your workload. Take advantage of SSD or NVMe for high performance, or lower your costs with HDD.

* **Open source software released under the Apache 2.0 license**

**Architecture:-**

****

Rook consists of multiple components:

* **Rook Operator** is the core of Rook.The Rook operator is a simple container that automatically bootstraps the storage clusters and monitors the storage daemons to ensure the storage clusters are healthy.
* **Rook Agents** run on each storage node and configure a FlexVolume plugin that integrates with Kubernetes’ volume controller framework. Agents handle all storage operations such as attaching network storage devices, mounting volumes on the host, and formatting the filesystem.
* **Rook Discovers** detect storage devices attached to the storage node.

Rook also deploys MON, OSD and MGR daemons for the Ceph clusters as Kubernetes pods.

The Rook Operator enables you to create and manage your storage clusters through CRDs. Each type of resource has its own CRD defined.

* A Rook clusters provides the settings of the storage cluster to serve block, object stores, and shared file systems. Each cluster has multiple pools.
* A pool manages the backing store for a block store. Pools are also used internally by object and file stores.
* An object store exposes storage with an S3-compatible interface.

     A file system provides shared storage for multiple Kubernetes pods.

The Rook daemons (Mons, OSDs, MGR, RGW, and MDS) are compiled to a single binary rook, and included in a minimal container. The rook container includes Ceph daemons and tools to manage and store all data – there are no changes to the data path. Rook does not attempt to maintain full fidelity with Ceph. Many of the Ceph concepts like placement groups and crush maps are hidden so you don’t have to worry about them. Instead Rook creates a much simplified UX for admins that is in terms of physical resources, pools, volumes, filesystems, and buckets.

**Current Usage:-**

Following are the tools/products that integrate with Rook. Rook provides an operator to deploy and manage these products

* **Minio object storage**

MinIO is a high performance distributed object storage server, designed for large-scale private cloud infrastructure. Rook provides an operator to deploy and manage MinIO clusters.

* **CockroachDB database**

CockroachDB is an open-source distributed SQL database that is highly scalable across multiple global regions and also highly durable, able to survive multiple hardware failures with minimal disruption. Its creators refer to it as “the SQL database for global cloud services.”

To simplify the deployment and management of CockroachDB, Rook has extended Kubernetes with a dedicated CockroachDB API object and operator. This effort was greatly simplified due to the functionality and benefits provided by the Rook framework.

* **EdgeFS**

Rook runs as a cloud-native service in Kubernetes clusters for optimal integration with applications in need of storage, and handles the heavy-lifting behind the scenes such as provisioning and management. Rook orchestrates innovating, next generation clould-connecting storage solution EdgeFS.

EdgeFS is high-performance, low latency and fault-tolerant object storage system with Geo-Transparent data access to file, block or object.

* **NFS Server**

Rook ceph storage system is used to deploy the NFS server

* **Naxenta**

Naxentaedge was built using EdgeFS and rook.io

Companies like Gini,Foretag, Feelway.inc, Melio Consulting, Lunar Ops.inc ,Codelab ,UIB , Kubernetes ,DevOPS and infrastructure etc use Rook in tech stacks

**Technical Details:-**

Rook runs the Kubernetes slave nodes and uses the disks as specified in the configuration. SSD's used for journaling of the HDD for more and faster IOPS. It creates separate pools only containing SSD's applications which require more IOPS and speed like databases and uses HDD for web applications or backups. For each disk, a pod runs which manages that disk and for each service like a monitor, a metadata server, Rados gateway and monitor a separate pod created.

Rados gateway is for providing object storage. S3 and swift API's are compatible with the Ceph object storage. RGW integrated with LDAP for user management. Rook deployed either in a hyper-converged manner (running in the same node as other applications) or hyperscale by giving the complete node to Ceph services pods only. The hyper-converged solution provides better utilization of resources and reducing infrastructure cost. When running out of storage add more disks and updated the deployment file. For more compute and storage add a new node in the cluster. It creates a storage class based on the pools. It can create separate storage class for SSD, HDD, and filesystem. Then use the storage class according to requirement while provisioning PVC.

**Project comparison:-**

**Ceph**

In computing,It is a free-software storage platform, implements object storage on a single distributed computer cluster, and provides interfaces for object-, block- and file-level storage.

**Amazon S3**

Amazon Simple Storage Service provides a fully redundant data storage infrastructure for storing and retrieving any amount of data, at any time, from anywhere on the web

**Google Cloud Storage**

Google Cloud Storage allows world-wide storing and retrieval of any amount of data and at any time. It provides a simple programming interface which enables developers to take advantage of Google's own reliable and fast networking infrastructure to perform data operations in a secure and cost effective manner. If expansion needs arise, developers can benefit from the scalability provided by Google's infrastructure.

**Azure Storage**

Azure Storage provides the flexibility to store and retrieve large amounts of unstructured data, such as documents and media files with Azure Blobs; structured nosql based data with Azure Tables; reliable messages with Azure Queues, and use SMB based Azure Files for migrating on-premises applications to the cloud.

**Amazon EBS**

Amazon EBS volumes are network-attached, and persist independently from the life of an instance. Amazon EBS provides highly available, highly reliable, predictable storage volumes that can be attached to a running Amazon EC2 instance and exposed as a device within the instance. Amazon EBS is particularly suited for applications that require a database, file system, or access to raw block level storage.

Reference

[www.rook.io](http://www.rook.io)  - official site of rook project(details of rook)

[www.cncf.io](http://www.cncf.io)  -official site of cncf

[www.github.com](http://www.github.com) - <https://github.com/rook/rook>

[www.thenewstack.io](http://www.thenewstack.io)  - news update of rook

[www.infoq.com](http://www.infoq.com) - blog on rook