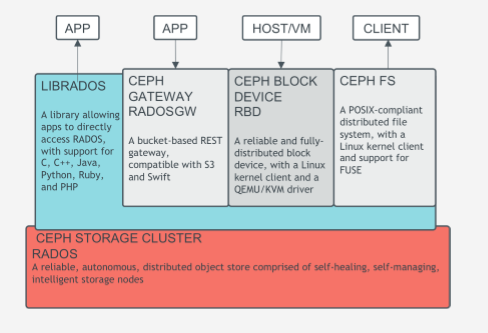
**CEPH Architecture**



**RADOS**

Reliable Autonomic Distributed Object Store (RADOS) is an open source object storage service that is an integral part of the Ceph distributed storage system. RADOS has the ability to scale to thousands of hardware devices by making use of management software that runs on each of the individual storage nodes. The software provides storage features such as thin provisioning, snapshots and replication. An algorithm called controlled replication under scalable hashing (CRUSH) determines how the data is replicated and mapped to the individual nodes.

**Communication Methods**

* **LIBRADOS**
* **CEPH GATEWAY(RADOS GW)**
* **CEPH BLOCK DEVICE(RBD)**
* **CEPH FS**

**LIBRADOS**

The [Ceph Storage Cluster](http://docs.ceph.com/docs/giant/glossary/" \l "term-ceph-storage-cluster) provides the basic storage service that allows [Ceph](http://docs.ceph.com/docs/giant/glossary/" \l "term-ceph) to uniquely deliver object, block, and file storage in one unified system. However, you are not limited to using the RESTful, block, or POSIX interfaces. Based upon RADOS, the librados API enables you to create your own interface to the Ceph Storage Cluster.

The librados API enables you to interact with the two types of daemons in the Ceph Storage Cluster:

* The [Ceph Monitor](http://docs.ceph.com/docs/giant/glossary/" \l "term-ceph-monitor), which maintains a master copy of the cluster map.
* The [Ceph OSD Daemon](http://docs.ceph.com/docs/giant/glossary/" \l "term-ceph-osd-daemon) (OSD), which stores data as objects on a storage node.

**CEPH GATEWAY(RADOS GW)**

[Ceph Object Gateway](http://docs.ceph.com/docs/giant/glossary/#term-ceph-object-gateway) is an object storage interface built on top of librgw to provide applications with a RESTful gateway to Ceph Storage Clusters. [Ceph Object Storage](http://docs.ceph.com/docs/giant/glossary/" \l "term-ceph-object-storage) supports two interfaces:

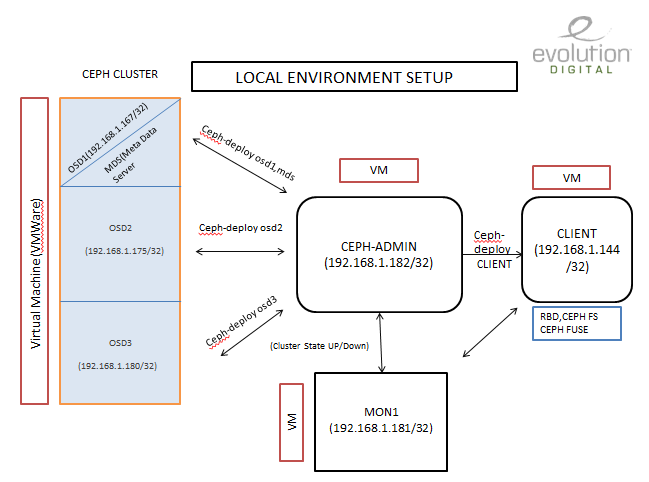
1. S3-compatible: Provides object storage functionality with an interface that is compatible with a large subset of the Amazon S3 RESTful API.
2. Swift-compatible: Provides object storage functionality with an interface that is compatible with a large subset of the OpenStack Swift API.

**CEPH BLOCK DEVICE(RBD)**

Ceph block devices are thin-provisioned, resizable and store data striped over multiple OSDs in a Ceph cluster. Ceph block devices leverage RADOScapabilities such as snapshotting, replication and consistency. Ceph’s RADOS Block Devices (RBD) interact with OSDs using kernel modules or the librbd library.

**CEPH FS**

The [Ceph Filesystem](http://docs.ceph.com/docs/giant/glossary/" \l "term-ceph-filesystem) (Ceph FS) is a POSIX-compliant filesystem that uses a Ceph Storage Cluster to store its data. The Ceph filesystem uses the same Ceph Storage Cluster system as Ceph Block Devices, Ceph Object Storage with its S3 and Swift APIs, or native bindings (librados). To mount Ceph FS we must have a running Ceph Storage Cluster with at least one [Ceph Metadata Server](http://docs.ceph.com/docs/giant/glossary/" \l "term-ceph-metadata-server) running.

**Our Local Environment CEPH Setup**

**We allocated storage to OSDs as follow**

**OSD1 - /dev/sdb 🡪 32 GB**

**OSD2 - /dev/sdb 🡪 21 GB**

**OSD3 - /dev/sdb 🡪 32 GB**

**-/dev/sdc 🡪 10 GB**

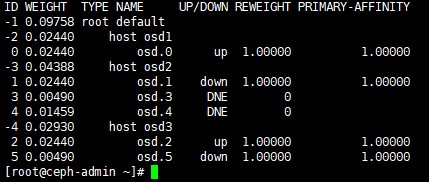
**CEPH installation and configuration**

* We already did CEPH installation along with OSD,Monitor and CEPH admin successfully on RHEL 7. As we are using RHEL 6 in our real time environment we tried to install it on RHEL 6.
* While installing CEPH on RHEL 6 we have faced following problems..

1. While running CEPH deployment command , the URL mentioned in the script /usr/lib/python2.6/site-packages/ceph\_deploy/hosts/centos/install.py is not redirected as **url = 'http://ceph.com/rpm-{version}/{repo}/'.format to 'http://download.ceph.com/rpm-{version}/{repo}/'.format’.**  So we have to change the url in the phython script manually.
2. While creating ceph repo, we have to update base url manually as it doesn’t getting redirect.
3. Radosgw library packages are not getting installed properly, So we gave up continuing installation on RHEL 6 and continued in RHEL 7.

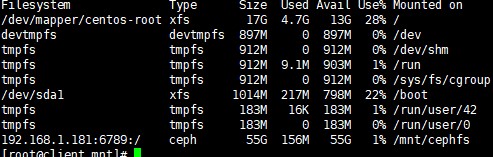
* After installing ceph on RHEL 7, We created rados block device(RBD) "/dev/rad0" from cluster and mount it to the client system under a mount point /mnt/rbdmount , to avoid mount point getting disconnected from cluster after every reboot we created a script "/usr/local/bin/rbd-mount" and systemd service (rbd-mount.service) in client machine.
* Then we studied function of CEPHFS and Metadata server and tried installation.
* Tested OSD function under different scenario and found that when more than half of OSD is down, the cluster state will be **changed from active to replay**. We can find the status of cluster in metadata server.

1. At first we down the OSD2 and keep others alive to check accessibility of files and also tested the same scenario by put down OSD1 and OSD3.
2. Please refer the image below where we put down OSD2 and keep OSD 1&3 alive.

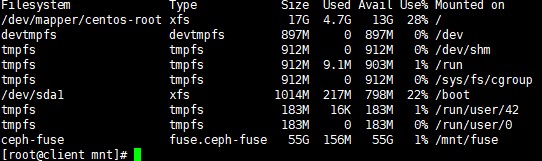


* Mounted cephfs and ceph fuse in client, to mount those file system we need metadata server and ceph monitor.

Mounting cephfs



Mounting ceph-fuse



* CRUSH alogirthm is used to define how to replicate data automatically and dynamically write on multiple OSDs. Now we are working on it.