STORAGE DEVICES

There are two Ceph daemons that store data on disk:

- **Ceph OSDs** (or Object Storage Daemons) are where most of the data is stored in Ceph. Generally speaking, each OSD is backed by a single storage device, like a traditional hard disk (HDD) or solid state disk (SSD). OSDs can also be backed by a combination of devices, like a HDD for most data and an SSD (or partition of an SSD) for some metadata. The number of OSDs in a cluster is generally a function of how much data will be stored, how big each storage device will be, and the level and type of redundancy (replication or erasure coding).
- **Ceph Monitor** daemons manage critical cluster state like cluster membership and authentication information. For smaller clusters a few gigabytes is all that is needed, although for larger clusters the monitor database can reach tens or possibly hundreds of gigabytes.

OSD BACKENDS

There are two ways that OSDs can manage the data they store. Starting with the Luminous 12.2.z release, the new default (and recommended) backend is *BlueStore*. Prior to Luminous, the default (and only option) was *FileStore*.

BLUESTORE

BlueStore is a special-purpose storage backend designed specifically for managing data on disk for Ceph OSD workloads. It is motivated by experience supporting and managing OSDs using FileStore over the last ten years. Key BlueStore features include:

- Direct management of storage devices. BlueStore consumes raw block devices or partitions. This avoids any intervening layers of abstraction (such as local file systems like XFS) that may limit performance or add complexity.
- Metadata management with RocksDB. We embed RocksDB's key/value database in order to manage internal metadata, such as the mapping from object names to block locations on disk.
- Full data and metadata checksumming. By default all data and metadata written to BlueStore is protected by one or more checksums. No data or metadata will be read from disk or returned to the user without being verified.
- Inline compression. Data written may be optionally compressed before being written to disk.
- Multi-device metadata tiering. BlueStore allows its internal journal (write-ahead log) to be written to a separate, high-speed device (like an SSD, NVMe, or NVDIMM) to increased performance. If a significant amount of faster storage is available, internal metadata can also be stored on the faster device.
- Efficient copy-on-write. RBD and CephFS snapshots rely on a copy-on-write *clone* mechanism that is implemented efficiently in BlueStore. This results in efficient IO both for regular snapshots and for erasure coded pools (which rely on cloning to implement efficient two-phase commits).

For more information, see BlueStore Config Reference and BlueStore Migration.

FILESTORE

FileStore is the legacy approach to storing objects in Ceph. It relies on a standard file system (normally XFS) in combination with a key/value database (traditionally LevelDB, now RocksDB) for some metadata.

FileStore is well-tested and widely used in production but suffers from many performance deficiencies due to its overall design and reliance on a traditional file system for storing object data.

Although FileStore is generally capable of functioning on most POSIX-compatible file systems (including btrfs and ext4), we only recommend that XFS be used. Both btrfs and ext4 have known bugs and deficiencies and their use may lead to data loss. By default all Ceph provisioning tools will use XFS.

For more information, see Filestore Config Reference.