Creating BlueStore OSDs Using Logical Volumes

Objectives

After completing this section, you should be able to describe OSD configuration scenarios and create BlueStore OSDs using cephadm.

Introducing BlueStore

BlueStore replaced FileStore as the default storage back end for OSDs. FileStore stores objects as files in a file system (Red Hat recommends XFS) on top of a block device. BlueStore stores objects directly on raw block devices and eliminates the file-system layer, which improves read and write operation speeds.

FileStore is deprecated. Continued use of FileStore in RHCS 5 requires a Red Hat support exception. Newly created OSDs, whether by cluster growth or disk replacement, use BlueStore by default.

BlueStore Architecture

Objects that are stored in a Ceph cluster have a cluster-wide unique identifier, binary object data, and object metadata. BlueStore stores the object metadata in the *block database*. The block database stores metadata as key-value pairs in a *RocksDB* database, which is a high-performing key-value store.

The block database resides on a small BlueFS partition on the storage device. BlueFS is a minimal file system that is designed to hold the RocksDB files.

BlueStore writes data to block storage devices by utilizing the *write-ahead log* (*WAL*). The write-ahead log performs a journaling function and logs all transactions.

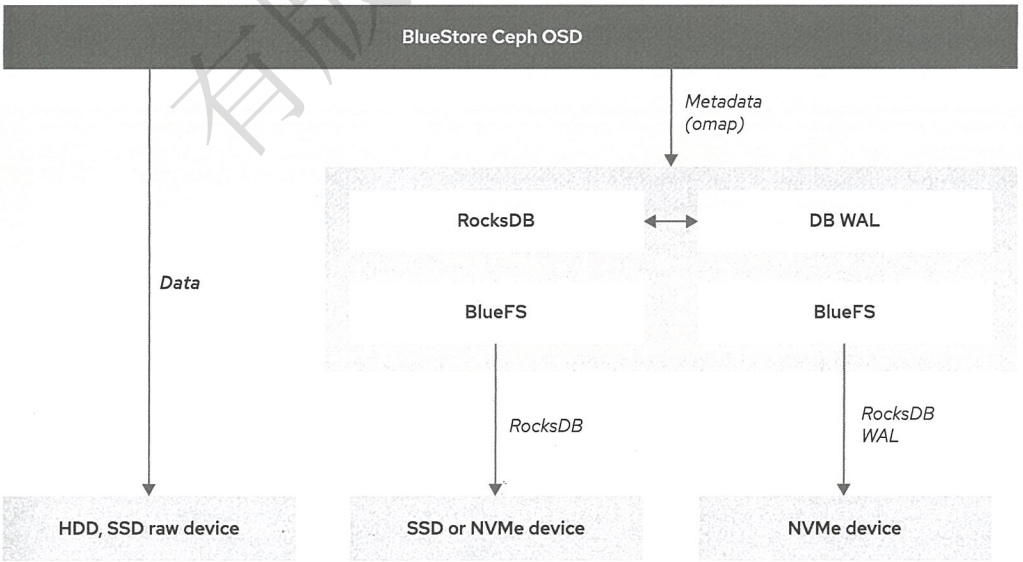

Figure 4.1: BlueStore OSD layout