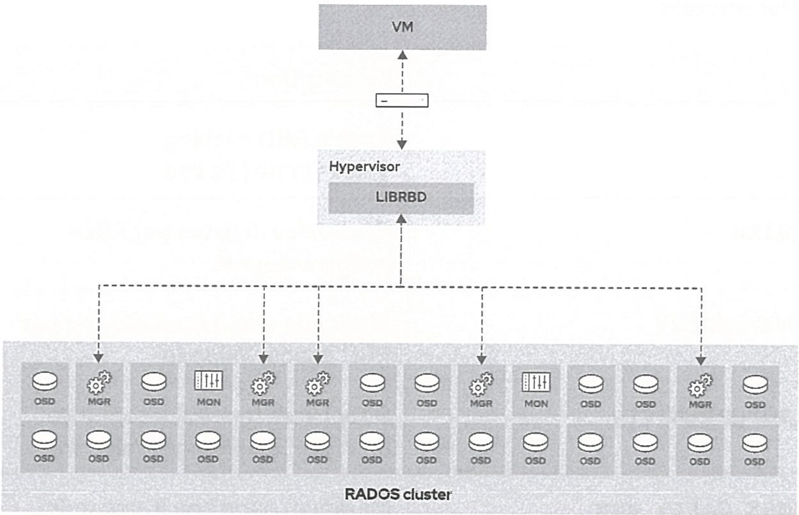

Figure 6.2: Virtual environment access

Because the user space implementation of the Ceph block device (for example, `librbd`) cannot take advantage of the Linux page cache, it performs its own in-memory caching, known as *RBD caching*. RBD caching behaves in a similar manner to the Linux page cache. When the OS implements a barrier mechanism or a flush request, Ceph writes all dirty data to the OSDs. This means that using write-back caching is just as safe as using physical hard disk caching with a VM that properly sends flushes (for example, Linux kernel >= 2.6.32). The cache uses a Least Recently Used (LRU) algorithm, and in write-back mode it can coalesce contiguous requests for better throughput.

Note

The RBD cache is local to the client because it uses RAM on the machine that initiated the I/O requests. For example, if you have Nova compute nodes in your Red Hat OpenStack Platform installation that use `librbd` for their virtual machines, the OpenStack client initiating the I/O request will use local RAM for its RBD cache.

有版权 禁止外传

RBD Caching Configurations

Caching Not Enabled

Reads and writes go to the Ceph Object Store. The Ceph cluster acknowledges the writes when the data is written and flushed on all relevant OSD journals.

Cache Enabled (write-back)

Considering two values, unflushed cache bytes `U` and maximum dirty cache bytes `M`, writes are acknowledged when `U < M`, or after writing data back to disk until `U < M`.

Write-through Caching

Set the maximum dirty byte to 0 to force write-through mode. The Ceph cluster acknowledges the writes when the data is written and flushed on all relevant OSD journals.

If using write-back mode, then the `librbd` library caches and acknowledges the I/O requests when it writes the data into the local cache of the server. Consider write-through for strategic production servers to reduce the risk of data loss or file system corruption in case of a server failure. Red Hat Ceph Storage offers the following set of RBD caching parameters: