RBD Clones

RBD clones are read/write copies of an RBD image that use a protected RBD snapshot as a base. An RBD clone can also be *flattened*, which converts it into an RBD image independent of its source. The cloning process has three steps:

1. Create a snapshot:

```
[root@node ~]# rbd snap create pool/image@snapshot
Creating snap: 100% complete...done.
```

2. Protect the snapshot from deletion:

```
[root@node ~]# rbd snap protect pool/image@snapshot
```

3. Create the clone using the protected snapshot:

```
[root@node ~]# rbd clone pool/imagename@snapshotname pool/clonename
```

The newly created clone behaves just like a regular RBD image. Clones support COW and COR, with COW as the default. COW copies the parent snapshot data into the clone before applying a write I/O request to the clone.

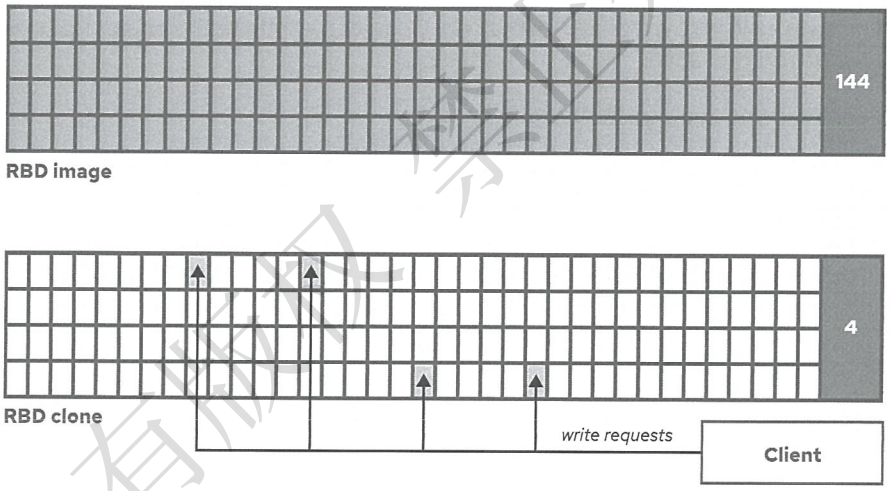

Figure 6.6: RBD clone write operation

You can also enable COR support for RBD clones. Data that is the same for the parent RBD snapshot and the clone is read directly from the parent. This can make reads more expensive if the parent's OSDs have high latency relative to the client. COR copies objects to the clone when they are first read.

If you enable COR, Ceph copies the data from the parent snapshot into the clone before processing a read I/O request, if the data is not already present in the clone. Activate the COR feature by running the `ceph config set client rbd_clone_copy_on_read true` command or the `ceph config set global rbd_clone_copy_on_read true` command for the client or global setting. The original data is not overwritten.