

After you have defined the rule, use it when creating a replicated pool:

```
[ceph: root@node /]# ceph osd pool create myfirstpool 50 50 inDC2
pool 'myfirstpool' created
```

For erasure coding, Ceph automatically creates rules for each erasure coded pool you create. The name of the rule is the name of the new pool. Ceph uses the rule parameters you define in the erasure code profile that you specify when you create the pool.

The following example first creates the new `myprofile` erasure code profile, then creates the `myecpool` pool based on this profile:

```
[ceph: root@node /]# ceph osd erasure-code-profile set myprofile k=2 m=1 \
crush-root=DC2 crush-failu re-domain=rack crush-device-class=ssd
[ceph: root@node /]# ceph osd pool create myecpool 50 50 erasure myprofile
pool 'myecpool' created
[ceph: root@node /]# ceph osd crush rule ls
replicated_rule
erasure-code
myecpool
```

Customizing the CRUSH Map by Decompiling the Binary Version

You can decompile and manually edit the CRUSH map with the following commands:

Command	Action
<code>ceph osd getcrushmap -o <i>binfile</i></code>	Export a binary copy of the current map.
<code>crushtool -d <i>binfile</i> -o <i>textfilepath</i></code>	Decompile a CRUSH map binary into a text file.
<code>crushtool -c <i>textfilepath</i> -o <i>binfile</i></code>	Compile a CRUSH map from text.
<code>crushtool -i <i>binfile</i> --test</code>	Perform dry runs on a binary CRUSH map and simulate placement group creation.
<code>ceph osd setcrushmap -i <i>binfile</i></code>	Import a binary CRUSH map into the cluster.



Note

The `ceph osd getcrushmap` and `ceph osd setcrushmap` commands provide a useful way to back up and restore the CRUSH map for your cluster.

Optimizing Placement Groups

Placement groups (PGs) allow the cluster to store millions of objects in a scalable way by aggregating them into groups. Objects are organized into placement groups based on the object's ID, the ID of the pool, and the number of placement groups in the pool.