Warning
Deleting a pool removes all data in the pool and is not reversible. You must set `mon_allow_pool_delete` to `true` to enable pool deletion.

- Prevent pool deletion for a specific pool by using the `ceph osd pool set` *`pool_name`* `nodelete true` command. Set `nodelete` back to `false` to allow deletion of the pool.
- View and modify pool configuration settings by using the `ceph osd pool set` and `ceph osd pool get` commands.
- List pools and pool configuration settings by using the `ceph osd lspools` and `ceph osd pool ls detail` commands.
- List pools usage and performance statistics by using the `ceph df` and `ceph osd pool stats` commands.
- Enable Ceph applications for a pool by using the `ceph osd pool application enable` command. Application types are `cephfs` for Ceph File System, `rbd` for Ceph Block Device, and `rgw` for RADOS Gateway.
- Set pool quotas to limit the maximum number of bytes or the maximum number of objects that can be stored in the pool by using the `ceph osd pool set-quota` command.

Important
When a pool reaches the configured quota, operations are blocked. You can remove a quota by setting its value to 0.

Configure these example setting values to enable protection against pool reconfiguration:

`osd_pool_default_flag_nodelete`
: Sets the default value of the `nodelete` flag on pools. Set the value to `true` to prevent pool deletion.

`osd_pool_default_flag_nopgchange`
: Sets the default value of the `nopgchange` flag on pools. Set the value to `true` to prevent changes to `pg_num`, and `pgp_num`.

`osd_pool_default_flag_nosizechange`
: Sets the default value of the `nosizechange` flag on pools. Set the value to `true` to prevent pool size changes.

Pool Namespaces

A *namespace* is a logical group of objects in a pool. Access to a pool can be limited so that a user can only store or retrieve objects in a particular namespace. One advantage of namespaces is to restrict user access to part of a pool.

Namespaces are useful for restricting storage access by an application. They allow you to logically partition a pool and restrict applications to specific namespaces inside the pool.

You could dedicate an entire pool to each application, but having more pools means more PGs per OSD, and PGs are computationally expensive. This might degrade OSD performance as load increases. With namespaces, you can keep the number of pools the same and not dedicate an entire pool to each application.