

DEPLOYING A DEVELOPMENT CLUSTER

In order to develop on ceph, a Ceph utility, *vstart.sh*, allows you to deploy fake local cluster for development purpose.

USAGE

It allows to deploy a fake local cluster on your machine for development purpose. It starts rgw, mon, osd and/or mds, or all of them if not specified.

To start your development cluster, type the following:

```
vstart.sh [OPTIONS] . . .
```

In order to stop the cluster, you can type:

```
./stop.sh
```

OPTIONS

- i** *ip_address*
Bind to the specified *ip_address* instead of guessing and resolve from hostname.
- k**
Keep old configuration files instead of overwriting theses.
- l, --localhost**
Use localhost instead of hostname.
- m** *ip[:port]*
Specifies monitor *ip* address and *port*.
- n, --new**
Create a new cluster.
- o** *config*
Add *config* to all sections in the ceph configuration.
- nodaemon**
Use ceph-run as wrapper for mon/osd/mds.
- smallmds**
Configure mds with small limit cache size.
- x**
Enable Cephx (on by default).
- X**
Disable Cephx.
- d, --debug**
Launch in debug mode
- valgrind[_{osd,mds,mon}]** '*valgrind_toolname [args...]*'
Launch the osd/mds/mon/all the ceph binaries using valgrind with the specified tool and arguments.
- bluestore**
Use bluestore as the objectstore backend for osds
- memstore**

Use memstore as the objectstore backend for osds

--cache <pool>

Set a cache-tier for the specified pool

ENVIRONMENT VARIABLES

{OSD,MDS,MON,RGW}

These environment variables will contain the number of instances of the desired ceph process you want to start.

Example:

```
OSD=3 MON=3 RGW=1 vstart.sh
```

DEPLOYING MULTIPLE DEVELOPMENT CLUSTERS ON THE SAME MACHINE

In order to bring up multiple ceph clusters on the same machine, *mstart.sh* a small wrapper around the above *vstart* can help.

USAGE

To start multiple clusters, you would run *mstart* for each cluster you would want to deploy, and it will start monitors, rgws for each cluster on different ports allowing you to run multiple mons, rgws etc. on the same cluster. Invoke it in the following way:

```
mstart.sh <cluster-name> <vstart options>
```

For eg:

```
./mstart.sh cluster1 -n -r
```

For stopping the cluster, you do:

```
./mstop.sh <cluster-name>
```