

How do i ... with BeeGFS?

... use fhgfs-ctl ...

■ fhgfs-ctl

- is part of „fhgfs-utils“
- can be found in /usr/bin/fhgfs-ctl
- central administration tool

... use fhgfs-ctl ...

```
[root@master1 ~]# fhgfs-ctl --help
FhGFS Command-Line Control Tool (http://www.fhgfs.com)
Version: 2014.01-r9

GENERAL USAGE:
$ fhgfs-ctl --<modename> --help
$ fhgfs-ctl --<modename> [mode_arguments] [client_arguments]

MODES:
--listnodes           => List registered clients and servers.
--listtargets         => List storage targets and node mappings.
--listpools           => List free space classes of servers.
--removenode (*)      => Remove (unregister) a node.
--maptarget (*)       => Map target to storage node.
--unmaptarget (*)     => Remove (unregister) a storage target.

--getentryinfo        => Show file system entry details.
--setpattern (*)      => Set a new striping configuration.
--mirrormd (*)        => Enable metadata mirroring. (EXPERIMENTAL)
--find                => Find files located on certain servers.
--refreshentryinfo    => Refresh file system entry metadata.

--createfile (*)      => Create a new file.
--createdir (*)       => Create a new directory.
--migrate              => Migrate files to other storage servers.
--disposeunused (*)   => Purge remains of unlinked files.

--serverstats         => Show server IO statistics.
--clientstats         => Show client IO statistics.
--userstats           => Show user IO statistics.
--storagebench (*)    => Run a storage targets benchmark.

--getquota            => Show quota information for users or groups.
...
```

... use fhgfs-ctl ...

```
[root@master1 ~]# fhgfs-ctl --listnodes --help
GENERAL USAGE:
  $ fhgfs-ctl --<modename> [mode_arguments] [client_arguments]

CLIENT ARGUMENTS:
Optional:
  --mount=<path>           Use config settings from this FhGFS mountpoint
                          (instead of "--cfgFile").
  --cfgFile=<path>        Path to FhGFS client config file.
                          (Default: /etc/fhgfs/fhgfs-client.conf)
  --logEnabled            Enable detailed logging.
  --<key>=<value>         Any setting from the client config file to override
                          the config file values (e.g. "--logLevel=5").

MODE ARGUMENTS:
Mandatory:
  --nodetype=<nodetype>  The node type (management, metadata,
                          storage, client).

Optional:
  --details              Print additional node details, such as network ports
                          and interface order.
  --nicdetails           Print additional network interconnect details, such as
                          IP address of each node interface.
  --showversion          Print node version code.
  --reachable           Check node reachability (from localhost).
  --reachretries=<num>  Number of retries for reachability check.
                          (Default: 6)
  --reachtimeout=<ms>   Timeout in ms for reachability check retry.
                          (Default: 500)
  --route               Print IP and protocol type through which the localhost
                          can connect to a node.
```

...

... use fhgfs-ctl ...

```
demo-io1:~ # fhgfs-ctl --listnodes --nodetype=storage
demo-io2 [ID: 1255]
demo-io1 [ID: 44966]
```

```
demo-io1:~ # fhgfs-ctl --listnodes --nodetype=meta
demo-io2 [ID: 1255]
demo-io1 [ID: 44966]
```

```
demo-io1:~ # fhgfs-ctl --listnodes --nodetype=storage --nicdetails
demo-io2 [ID: 1255]
  Ports: UDP: 18003; TCP: 18003
  Interfaces:
  + ib0[ip addr: 10.12.201.204; type: RDMA]
  + ib0[ip addr: 10.12.201.204; type: TCP]
  + eth1[ip addr: 10.10.201.204; type: TCP]
demo-io1 [ID: 44966]
  Ports: UDP: 18003; TCP: 18003
  Interfaces:
  + ib0[ip addr: 10.12.201.203; type: RDMA]
  + ib0[ip addr: 10.12.201.203; type: TCP]
  + eth0[ip addr: 10.10.201.203; type: TCP]

Number of nodes: 2
```

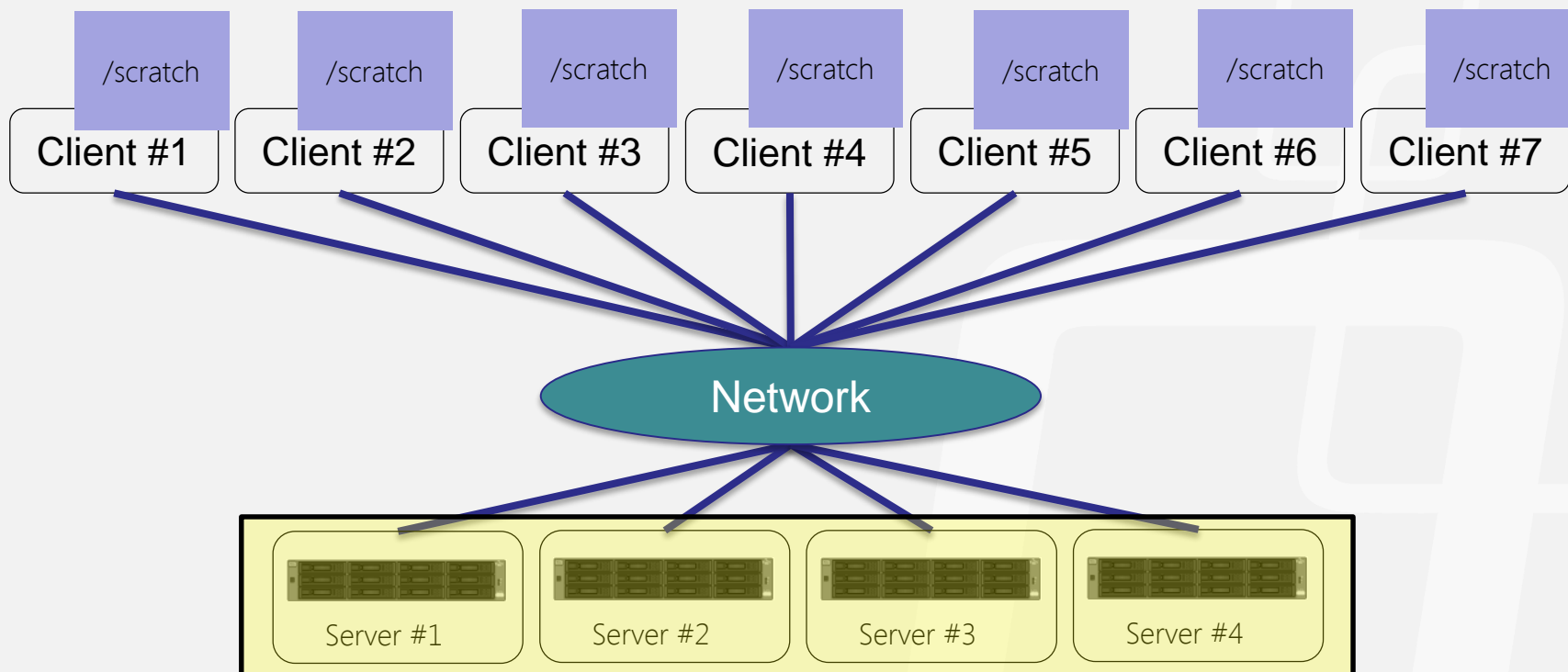
```
demo-io1:~ # fhgfs-ctl --listnodes --nodetype=storage --reachable
demo-io2 [ID: 1255]
  Reachable: <yes>
demo-io1 [ID: 44966]
  Reachable: <yes>
```

... query free space ...

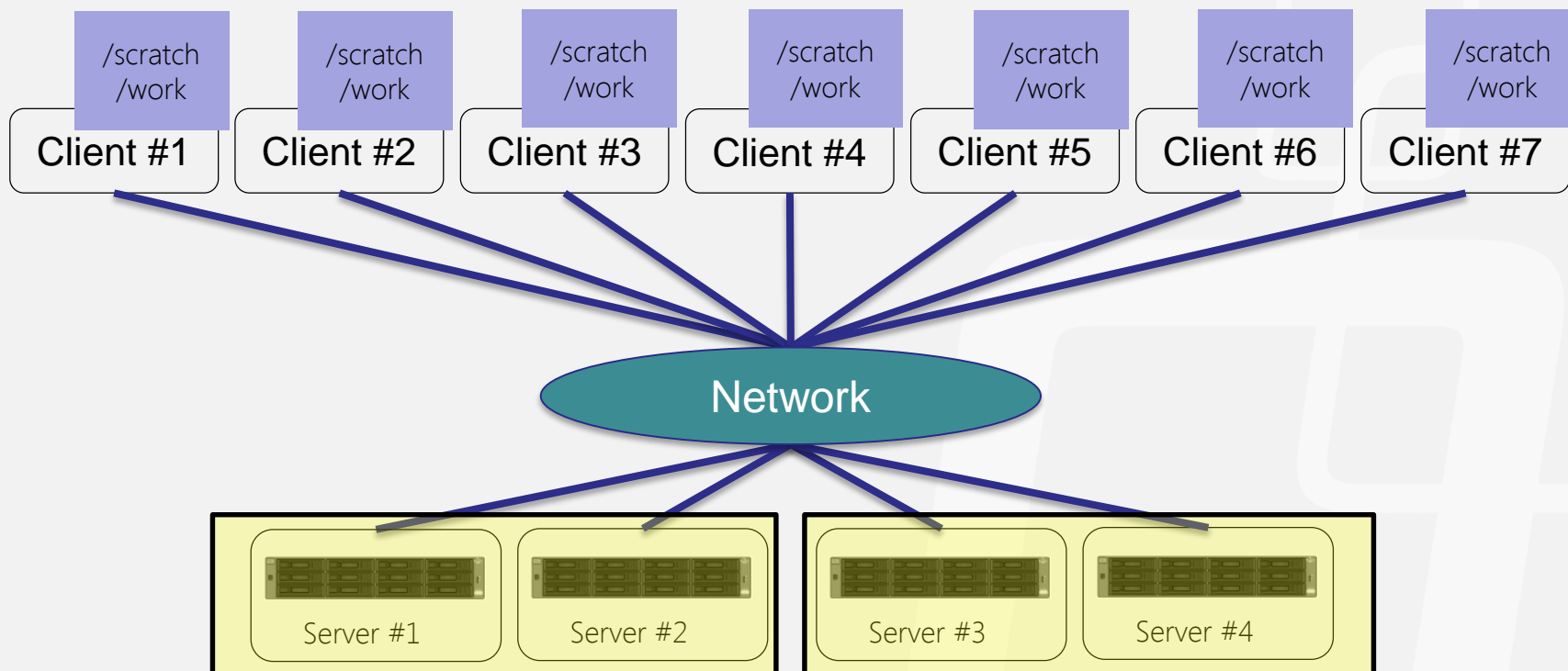
```
demo-iol:~ # fhgfs-df
METADATA SERVERS:
TargetID      Pool      Total      Free      ITotal     IFree
=====      =====
1255          [normal]  238.1GB    73.0GB    158.8M     158.8M
44966         [normal]  238.1GB    72.5GB    158.8M     158.7M

STORAGE TARGETS:
TargetID      Pool      Total      Free      ITotal     IFree
=====      =====
76            [normal]  9168.7GB   9168.5GB  582.2M     582.2M
14358         [normal]  9168.7GB   9168.5GB  582.2M     582.2M
48173         [normal]  9168.7GB   9168.5GB  582.2M     582.2M
61042         [normal]  9168.7GB   9168.5GB  582.2M     582.2M
```

... mount multiple BeeGFS' ...



... mount multiple BeeGFS' ...

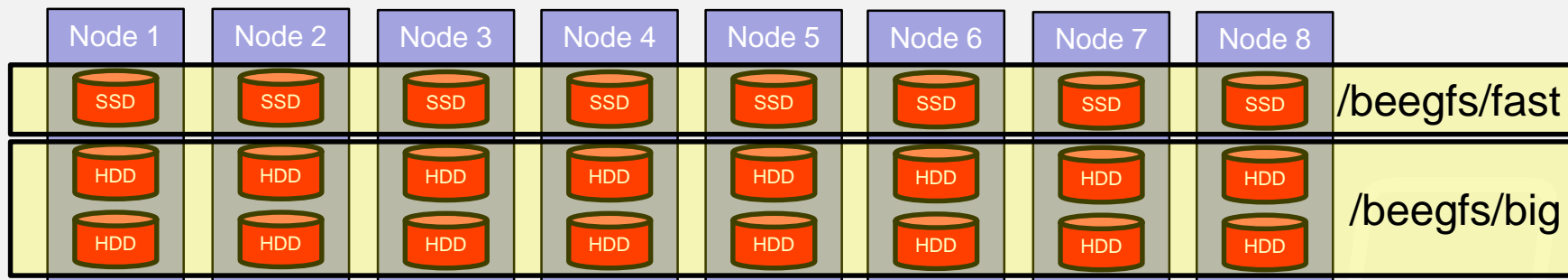


... mount multiple BeeGFS' ...

- create a second fhgfs-client.conf...
 - ... with a new name
 - ... pointing to a second management server
- edit /etc/fhgfs/fhgfs-mounts.conf
 - with a new name

```
demo-iol:~ # cat /etc/fhgfs/fhgfs-mounts.conf
/scratch      /etc/fhgfs/fhgfs-client-scratch.conf
/work         /etc/fhgfs/fhgfs-client-work.conf
```

... export multiple BeeGFS' ...



- enable „multi mode“ in /etc/default/fhgfs- ...

```
demo-iol:~ # cat /etc/default/fhgfs-storage
```

```
# Set to "NO" if you do not want to start the Fhgfs storage server
START_SERVICE="YES"
```

```
# Set to "YES" if you want to start multiple storage servers with different
# configuration files on this machine.
```

```
#
# Create a subdirectory with the ending ".d" in "/etc/fhgfs/" for every config
# file. The subdirectory name will be used to identify a particular server
# instance for init script start/stop.
```

```
#
# Example: /etc/fhgfs/scratch.d/fhgfs-storage.conf
```

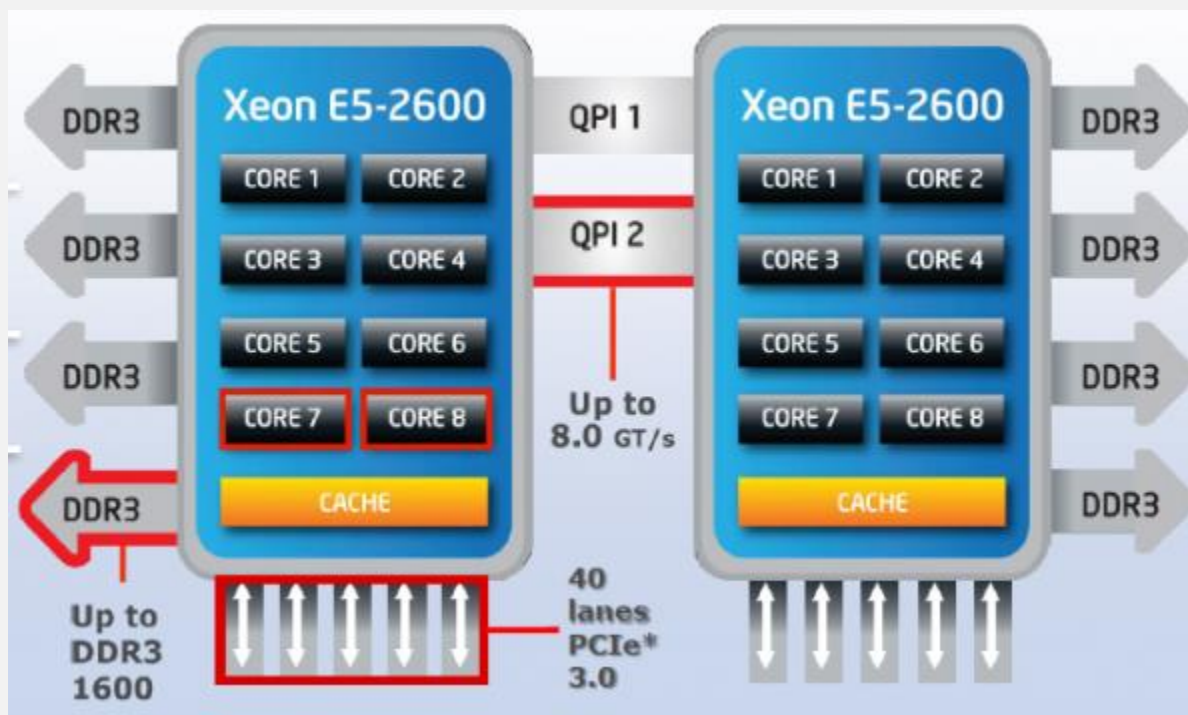
```
# $ /etc/init.d/fhgfs-storage start scratch
MULTI_MODE="YES"
```

... export multiple BeeGFS' ...

- create a separate directory in /etc/fhgfs for each instance
 - /etc/fhgfs/scratch.d/
 - /etc/fhgfs/work.d/
- create a configuration file for each daemon in the matching directory
 - fhgfs-meta.conf, fhgfs-storage.conf, fhgfs-mgmt.d.conf...
- adjust the parameters in the .conf files

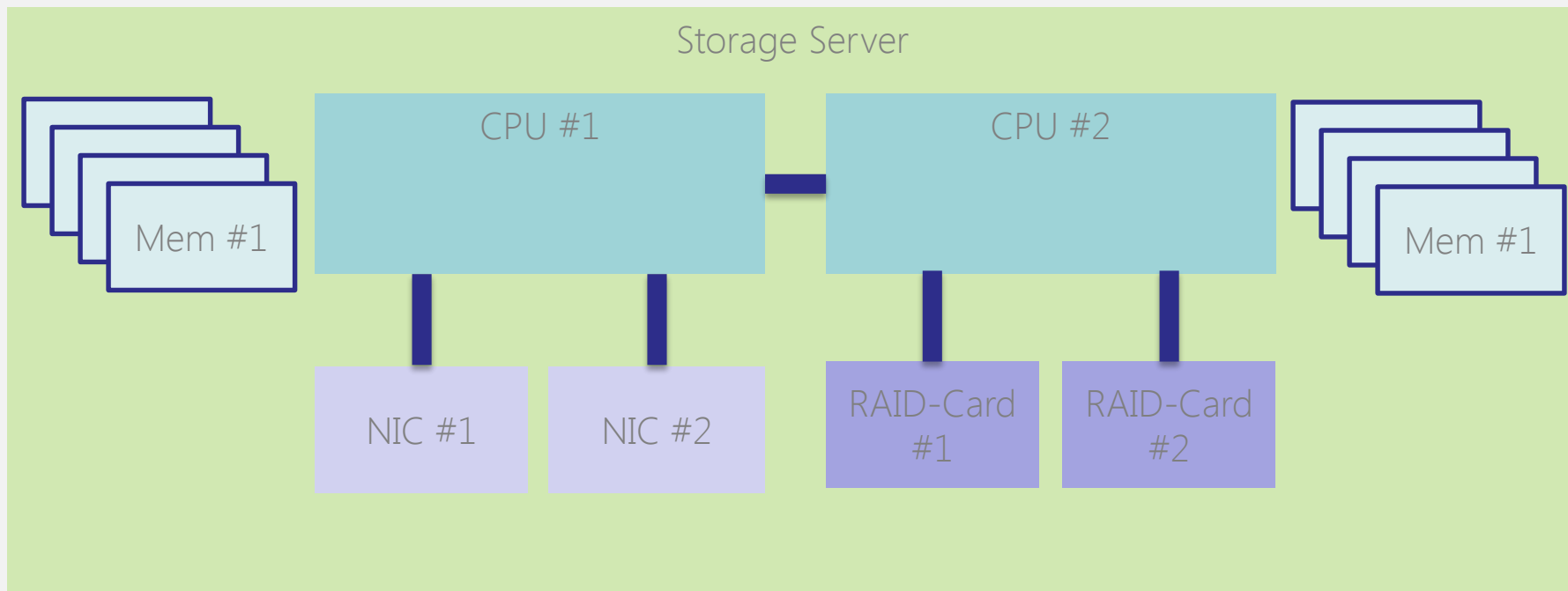
```
...
sysMgmtHost           = 10.10.201.203
storeStorageDirectory = /meta/ost-ssd
...
connPortShift         = 0
connMgmtPortUDP       = 28008
connMgmtPortTCP       = 28008
connStoragePortUDP    = 28003
connStoragePortTCP    = 28003
...
```

... pin BeeGFS' processes ...

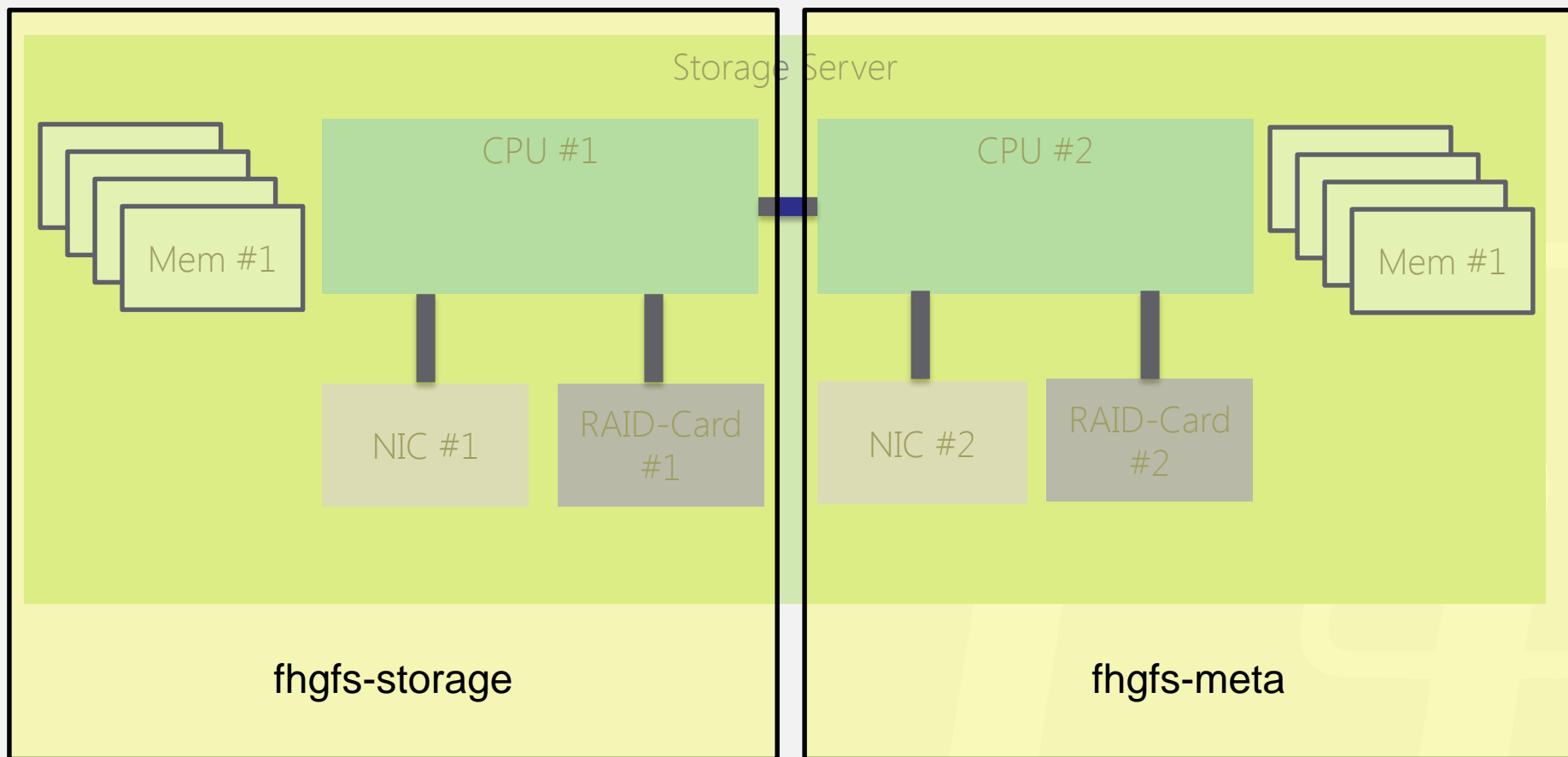


- use „tuneBindToNumaZone“
 - `ls -al /sys/devices/system/node/node{0,1}`

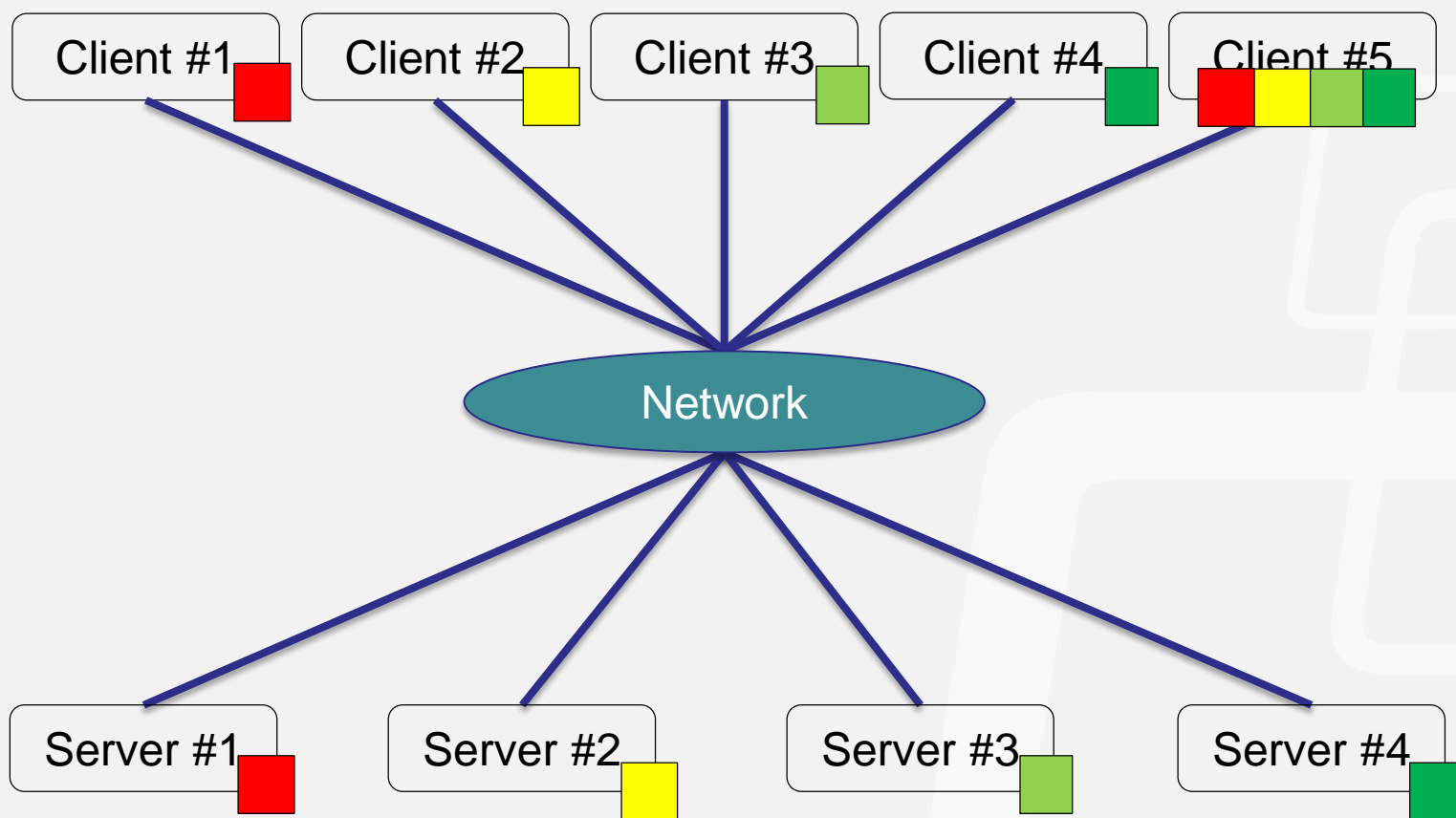
... pin BeeGFS' processes ...



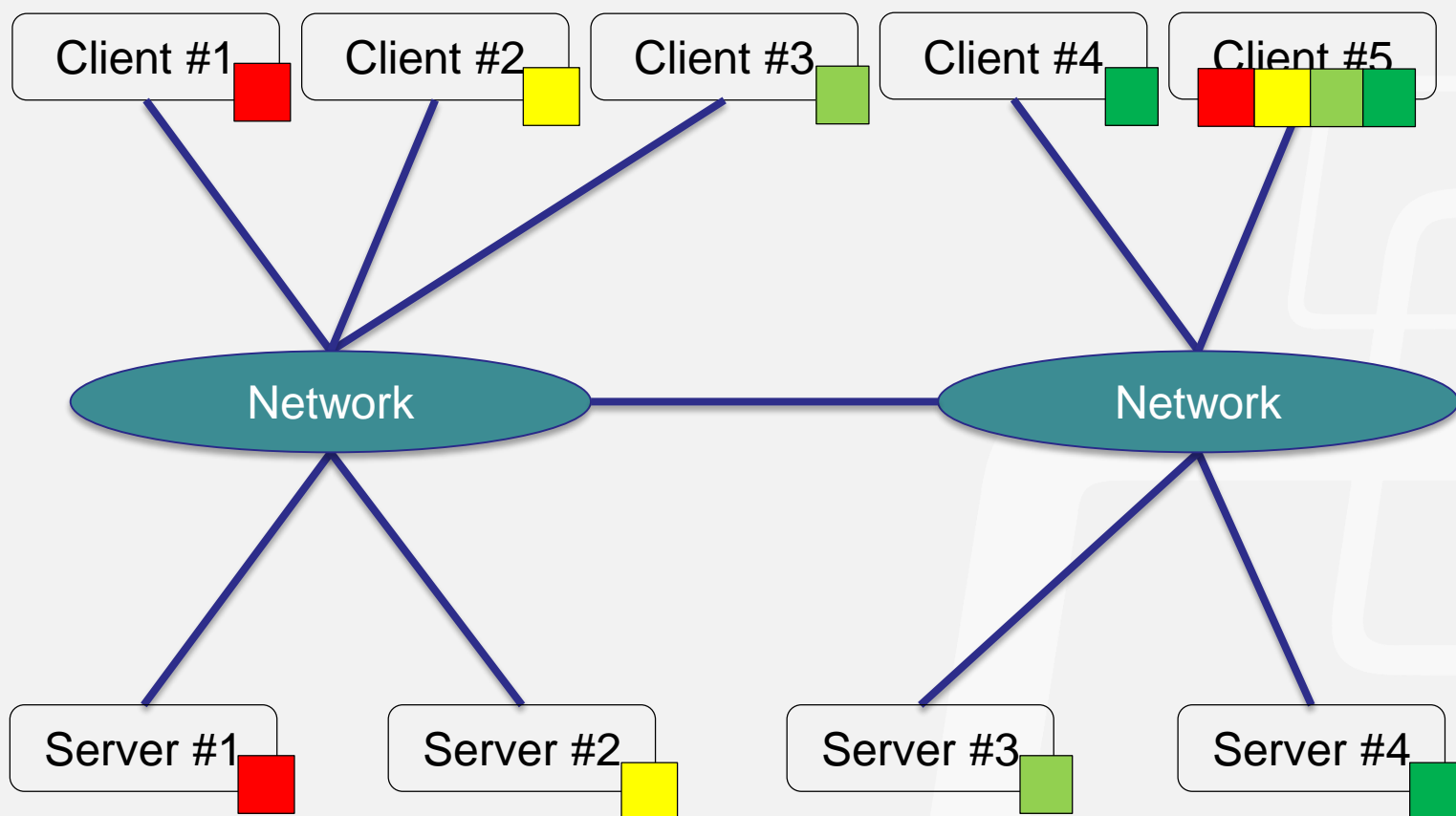
... pin BeeGFS' processes ...



... assign storage servers to clients ...



... assign storage servers to clients ...



... assign storage servers to clients ...

- use „tunePreferred{Meta,Storage}File“ in fhgfs-client.conf

```
demo-io1:/etc/fhgfs # fhgfs-ctl --listtargets
```

```
TargetID  NodeID
```

```
=====  =====
```

```
76        1255
```

```
14358     1255
```

```
48173     44966
```

```
61042     44966
```

```
demo-io1:/etc/fhgfs # fhgfs-ctl --listnodes --nodetype=meta
```

```
demo-io2 [ID: 1255]
```

```
demo-io1 [ID: 44966]
```

... limit traffic to a certain interface ...

- use „conn{Interfaces,NetFilter}File“ in fhgfs-client.conf

```
demo-io1:/etc/fhgfs # fhgfs-ctl --listnodes --nodetype=client --nicdetails
14E8-53E4DE10-demo-io1
Ports: UDP: 18004; TCP: 0
Interfaces:
+ eth0[ip addr: 10.10.201.203; type: TCP]
FC13-53EB9471-demo-io2
Ports: UDP: 18004; TCP: 0
Interfaces:
+ ib0[ip addr: 10.12.201.204; type: RDMA]
+ ib0[ip addr: 10.12.201.204; type: TCP]
+ eth1[ip addr: 10.10.201.204; type: TCP]

Number of nodes: 2
```

... query properties of files and directories ...

```
demo-io1:/etc/fhgfs # fhgfs-ctl --getentryinfo /mnt/fhgfs/testdir/  
Path: /testdir  
Mount: /mnt/fhgfs  
EntryID: 0-5429B25E-AFA6  
Metadata node: demo-io1 [ID: 44966]  
Stripe pattern details:  
+ Type: RAID0  
+ Chunksize: 512K  
+ Number of storage targets: desired: 4
```

```
demo-io1:/etc/fhgfs # fhgfs-ctl --getentryinfo /mnt/fhgfs/testdir/test  
Path: /testdir/test  
Mount: /mnt/fhgfs  
EntryID: 0-5429B29A-AFA6  
Metadata node: demo-io1 [ID: 44966]  
Stripe pattern details:  
+ Type: RAID0  
+ Chunksize: 512K  
+ Number of storage targets: desired: 4; actual: 4  
+ Storage targets:  
+ 61042 @ demo-io1 [ID: 44966]  
+ 48173 @ demo-io1 [ID: 44966]  
+ 76 @ demo-io2 [ID: 1255]  
+ 14358 @ demo-io2 [ID: 1255]
```

... get performance metrics ...

```
demo-iol:/etc/fhgfs # fhgfs-ctl --help
FhGFS Command-Line Control Tool (http://www.fhgfs.com)
Version: 2014.01-r8

...

--serverstats      => Show server IO statistics.
--clientstats     => Show client IO statistics.
--userstats       => Show user IO statistics.
--storagebench (*) => Run a storage targets benchmark.

...
```

```
demo-iol:/etc/fhgfs # fhgfs-ctl --serverstats --perserver --interval=2
```

```
==== time index: 1412019239 (values show last second only) ====
```

nodeID	write_KiB	read_KiB	reqs	qlen	bsy
1255	0	0	0	0	0
44966	0	0	0	0	0

```
==== time index: 1412019241 (values show last second only) ====
```

nodeID	write_KiB	read_KiB	reqs	qlen	bsy
1255	0	0	3	0	0
44966	0	0	1	0	0

... fill new (and empty) targets ...

```
demo-iol:/etc/fhgfs # cat /etc/fhgfs/fhgfs-mgmt.conf
# This is a config file for Fraunhofer parallel file system management nodes.
# http://www.fhgfs.com

...
tuneMetaSpaceLowLimit           = 10G
tuneMetaSpaceEmergencyLimit     = 3G
tuneStorageSpaceLowLimit        = 512G
tuneStorageSpaceEmergencyLimit  = 10G
...
```

```
foreach directory in /mnt/fhgfs do
```

```
  foreach file do
```

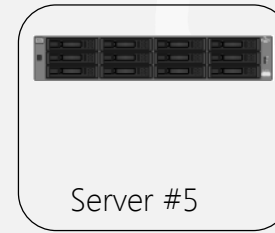
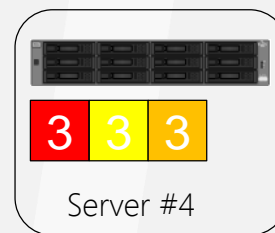
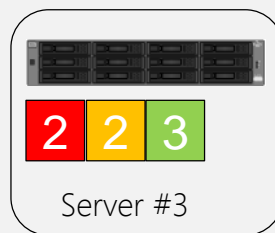
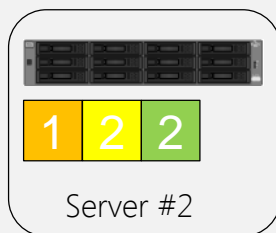
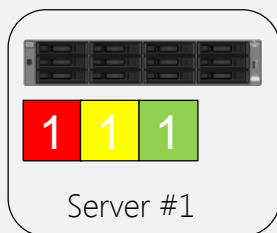
```
    cp file file.tmp
```

```
    rm file
```

```
    mv file.tmp file
```

```
  done
```

```
done
```



... drain a target ...

```
demo-iol:/etc/fhgfs # fhgfs-ctl --migrate --help
GENERAL USAGE:
  $ fhgfs-ctl --<modename> [mode_arguments] [client_arguments]

...

MODE ARGUMENTS:
Mandatory:
  <path>                Path to a file or directory.

Optional:
  --targetid=<targetID>  Migrate files away from the given targetID.
  --nodeid=<nodeID>      Migrate files away from all targets attached to this
                        nodeID.
  --nomirrors            Do not include mirror targets in search.
  --verbose              Print verbose messages, e.g. all files being migrated.

Note: Either targetID or nodeID must be specified.
Note: DO NOT access the file system while migration is in progress!

USAGE:
This mode frees up storage targets by moving the contained files to other
storage targets.
To achieve this, each matching file will be copied to a temporary file on a
different set of storage targets and then be renamed to the original file name.
Thus, migration should not be done while clients are accessing the file system.

Example: Migrate all files from storage target with ID "5" to other targets.
  $ fhgfs-ctl --migrate --targetid=5 /mnt/fhgfs
```

... do a filesystem check ...

```
demo-io1:/etc/fhgfs # fhgfs-fsck --help
FhGFS File System Check (http://www.fhgfs.com)
Version: 2014.01-r8

GENERAL USAGE:
$ fhgfs-fsck --<modename> --help
$ fhgfs-fsck --<modename> [mode_arguments] [client_arguments]

MODES:
--checkfs      => Perform a full check and optional repair of
                 a FhGFS instance.
--enablequota => Set attributes needed for quota support in FhGFS.
                 Can be used to enable quota support on an existing
                 system.

USAGE:
This is the FhGFS file system consistency check and repair tool.

Choose a mode from the list above and use the parameter "--help"
to show arguments and usage examples for that particular mode.

(Running fhgfs-fsck requires root privileges.)

Example: Show help for mode "--checkfs"
$ fhgfs-fsck --checkfs --help
```

... do a filesystem check ...

```
demo-io1:/etc/fhgfs # fhgfs-fsck --checkfs

FhGFS File System Check Version : 2014.01-r8
----

-----
Started FhGFS fsck in forward check mode [Mon Sep 29 21:47:57 2014]
Log will be written to /var/log/fhgfs-fsck.log
Database will be saved as /var/lib/fhgfs/fhgfs-fsck.db
-----

Step 1: Check reachability of nodes: Finished
Step 2: Gather data from nodes:

Fetched data > Directory entries: 5 | Inodes: 8 | Chunks: 2

Step 3: Check for errors...

* Target is used, but does not exist... Finished
* File has a missing target in stripe pattern... Finished
* Dentry-by-ID file is present, but no corresponding dentry... Finished
* Dentry-by-ID file is broken or missing... Finished
* Chunk is saved in wrong path... Finished
* Wrong owner node saved in inode... Finished
* Dentry points to inode on wrong node... Finished
* Content directory without an inode... Finished
* Dir inode without a dentry pointing to it (orphaned inode)... Finished
* File inode without a dentry pointing to it (orphaned inode)... Finished
* Chunk without an inode pointing to it (orphaned chunk)... Finished
* Dangling directory entry... Finished
* Directory inode without a content directory... Finished
* Attributes of file inode are wrong... Finished
* Attributes of dir inode are wrong... Finished
```