

Name Space Quotas Management

What is the problem?

The number of names in DFS is constrained by the local memory of the name node. If the name space is exhausted, the name node is likely to exhibit ugly behavior, perhaps to the point of becoming unmanageable by an administrator.

On at least one occasion, an ill-considered job had thousands of tasks that each created thousands of files that exhausted the name space and made the cluster unavailable to other users. Therefore, there should be protection against the rogue, but not necessarily malicious, job.

Solution: Directory Quota

The use of name space could be limited either by users, jobs, or directories. This proposal chooses the directory quotas approach. It separates quota management from user identity. This would allow in an implementation independent of the permissions model.

Quotas on individual directories are set explicitly by an administrator. Newly created directories have no associated quota. The largest quota is `Integer.MaxValue`. A quota of one forces a directory to remain empty.

The directory quota is a hard limit on the number of names in the tree rooted at that directory. File/directory creations fault if the quota would be exceeded. Quotas stick to renamed directories; the rename operation faults if operation would result in a quota violation. The attempt to set a quota faults if the directory would be in violation of the new quota.

Directory nodes with a quota keep count of the number of names in the tree rooted at the node in addition to the quota value as set. A creation operation decrements counts on the way down, and increments counts on the way up if a quota violation is discovered. Similarly, a successful delete operation increments counts. A rename operation increments counts above the source and decrements counts above the target.

Quotas will be persistent with the fsimage. Actual counts can be reconstructed when the name node starts. When starting, if the fsimage is immediately in violation of a quota (perhaps the fsimage was surreptitiously modified), startup operation fails with an error report. Setting/removing a quota creates a journal entry.

The following new commands or new options are added to support quotas. The first two are administration commands.

dfsadmin -setquota <N> <directory>

Set the quota N for each directory. Best effort for each directory, with faults reported if N is not a positive integer, user is not an administrator, the directory does not exist or it is a file, or the directory would immediately exceed the new quota.

dfsadmin -clrquota [directory]

Remove any quota for each directory. Best effort for each directory, with faults reported if the directory does not exist or it is a file, or if user is not an administrator. It is not a fault if the directory has no quota.

fs -count -q [directory]

With the -q option, also report the quota value set for each directory, and the available quota remaining. If the directory does not have a quota set, the reported values are "none" and "inf".

Test Plan

Test setquota/cclrquota/count -q commands

All operations are issued by an administrator if not otherwise noted.

Test cases	Description	Expected Behaviors
1	Create a directory /test and set its quota to be 3 by an administrator	The operation succeeds
2	Create a subdirectory /test/data0	The operation succeeds
3	Create a file /test/datafile0	The operation succeeds
4	Count -q /test	Returns a quota of 3 and a count of 3
5	Count -q /test/data0	Return an undefined quota and a count of 1
6	Create a subdirectory /test/data1	The operation fails with QuotaExceededException
7	Create a file /test/datafile1	The operation fails with QuotaExceededException
8	Clrquota /test	The operation succeeds
9	Clrquota /test/data0	The operation succeeds
10	Create a file /test/datafile1	Operation succeeds
11	Set the quota of /test to be 1	The operation fails
12	Set the quota of /test/data0 to be 1	The operation succeeds. /test/data0 has a count of 1.
13	Create a directory /test/data0/in	The operation fails

14	Setquota on a. a non-existent directory /test1 b. a file /test/data0	Command fails with a non-existent directory error
15	Clrquota on a. a non-existent directory /test1 b. file /test/data0	Command fails with a non-existent directory error
16	Set the quota of /test to be an illegal quota: a. 0 b. -1 c. a number greater than Integer.MAX_VALUE d. a non integer	Command fails with a IllegalArgumentException
17	Setquota /test issued by a non-administrator	Permission Denied exception
18	Clrquota /test/data0 issued by a non-administrator	Permission Denied exception

Test mkdir/rename/delete

Test cases	Description	Expected Behaviors
1	Create directory /nqdir0/qdir1/qdir20/nqdir30	The operation succeeds
2	Setquota /nqdir0/qdir1 6	The operation succeeds. The directory has a quota of 6 and a count of 3
3	Setquota /nqdir0/qdir1/qdir20 7	The operation succeeds. The directory has a quota of 7 and a count of 2.
4	Create directory /nqdir0/qdir1/qdir21 and set its quota to be 2	The operation succeeds. /nqdir0/qdir1/qdir21 has a quota of 2 and a count of 1
5	Create directory /nqdir0/qdir1/qdir21/nqdir32	The operation succeeds. /nqdir0/qdir1 has a quota of 2 and a count of 2.
6	Create directory /nqdir0/qdir1/qdir21/nqdir33	Failed with QuotaExceededException
7	Create directory /nqdir0/qdir1/qdir20/nqdir31	The operation succeeds. /nqdir0/qdir1/qdir20 has a quota of 7 and a count of 3. /nqdir0/qdir1 has a quota of 6 and count of 6.
8	Create directory	Failed with

	/nqdir0/qdir1/qdir20/nqdir33	QuotaExceededException
9	Move /nqdir0/qdir1/qdir21/nqdir32 /nqdir0/qdir1/qdir20/qdir30	The operation succeeds. /nqdir0/qdir1/qdir20 has a quota of 7 and a count of 4. /nqdir0/qdir1 has a quota of 6 and count of 6.
10	Move /nqdir0/qdir1/qdir20/nqdir30 /nqdir0/qdir1/qdir21	Failed with QuotaExceededException
11	Move /nqdir0/qdir1/qdir20/nqdir30 /nqdir0	The operation succeeds. /nqdir0/qdir1/qdir20 has a quota of 7 and a count of 2. /nqdir0/qdir1 has a quota of 6 and count of 4.
12	Create directory /nqdir0/nqdir30/nqdir33	The operation succeeds.
13	Move /nqdir0/qdir30 /nqdir0/qdir1/qdir20/qdir30	Failed with QuotaExceededException
14	Move /nqdir0/qdir1/qdir21 /nqdir0/qdir1/qdir20	The operation succeeds. /nqdir0/qdir1/qdir20 has a quota of 7 and a count of 3. /nqdir0/qdir1 has a quota of 6 and count of 4. /nqdir0/qdir1/qdir20/qdir21 has a quota of 2 and a count of 1.
16	Delete /nqdir0/qdir1/qdir20/qdir21	The operation succeeds. /nqdir0/qdir1/qdir20 has a quota of 7 and a count of 2. /nqdir0/qdir1 has a quota of 6 and count of 3.
17	Rename /nqdir0/qdir30 /nqdir0/qdir1/qdir20	The operation succeeds. /nqdir0/qdir1/qdir20 has a quota of 7 and a count of 5. /nqdir0/qdir1 has a quota of 6 and count of 6.

Test upgrade and fsimage saving and loading (manual tests)

Test cases	Description	Expected Behaviors
1	Start an dfs with pre-quota code; create directory /dir1/dir2/dir3; create files /dir1/file1, /dir1/dir2/file2, /dir1/dir2/dir3/file3;	The operation succeeds

	Shutdown the cluster;	
2	Start dfs with quota code and the pre-quota fsimage	All directories have an undefined quota
3	Setquota /dir1 7; Setquota /dir1/dir2/dir3 5;	The operation succeeds. /dir1 has a quota of 7 and a count of 6, /dir1/dir2 has an undefined quota and a count of 4, and /dir1/dir2/dir3 has a quota of 5 and a count of 2
4	Shutdown the cluster and restart it	/dir1 has a quota of 7 and a count of 6, /dir1/dir2 has an undefined quota and a count of 4, and /dir1/dir2/dir3 has a quota of 5 and a count of 2.