

Silicon Graphics, Inc.

# **XFS Overview & Internals**

## **08 - Quotas**

© Copyright 2006 Silicon Graphics Inc. All rights reserved.

Permission is granted to copy, distribute, and/or modify this document under the terms of the Creative Commons Attribution-Share Alike, Version 3.0 or any later version published by the Creative Commons Corp. A copy of the license is available at <http://creativecommons.org/licenses/by-sa/3.0/us/> .

November 2006



# Quotas

- Quotas provide a method for managing disk resources.
  - Quotas can be used to report on disk usage.
  - Quotas can be used to control disk usage.
- Quotas manage filesystem resources in two ways.
  - by Managing Disk Space (blocks)
  - by Managing the Number of Files (inodes)
- These resources can be managed on a per user, group or directory (project) basis.

# Quota Types

- Quotas can be tracked or enforced for
- Users
  - Manage the disk usage of individuals or system users.
- Groups
  - Manage the disk usage of groups.
- Projects
  - Manage the disk usage of directories associated with a specific project. Projects may cross organizational 'group' boundaries
  - Provide broader levels of control than available using user or group quotas.
  - Can be used to effectively shrink a file system allowing administrators to reallocate disk space as needed.

# Reporting Vs Enforcement

Quotas can be used to monitor disk usage and optionally enforce limits.

- Quotas can be used to monitor disk usage across quota types.
  - More efficient than using unix tools like du that must traverse the filesystem.
  - Quotas are maintained as the filesystem is used.
- Quotas can be used to enforce limits on users, groups or projects. Quota limits can prevent.
  - Individual users consuming excessive resources.
  - Rogue processes bringing down a system or causing denial of services.

# Enforcement (Limits)

There are two types of limits; Hard and Soft.

- **Soft** limits are advisory.
  - Allow additional disk resources to be consumed for a period of time. this is known as the *grace* period.
  - Provide flexibility to users and processes that may occasionally need to exceed there quota temporarily
- **Hard** Prevent further allocation of disk resources.
  - Any I/O that attempt to allocate further disk resources is failed.
  - Existing data on the disk is preserved and existing disk allocations can still be written.
  - Users must remove existing files to before new resources can be used.
  - Soft limits are be enforced as hard limits after the *grace* period.
- Limits are never applied to the *root* user.

# Configuring Quotas

- XFS quotas are enabled as filesystem mount options when the filesystem is mounted.
- User, Group and Project quotas are enabled independently. Group and Project quotas are incompatible.
- Each quota type can be specified as noenforce, in this case quota reporting will be maintained but not enforced.
- Valid mount options are
  - uquota, uqnoenforce - User quotas
  - gquota, gqnoenforce - Group quotas
  - pquota, pqnoenforce - Project quota

## xfs\_quota

- Once quotas are configured the `xfs_quota` tool can be used to report on disk usage and set limits.
- Quota operation is controlled by `xfs_quota`.
- `xfs_quota` has two modes of operation, basic and expert mode
  - Basic mode includes commands for reporting disk usage to users.
  - Expert mode (`xfs_quota -x`) contains advanced commands that allow the modification of the quota system.

# Quota Reporting

```
> sudo xfs_quota -xc 'report -h' /home
User quota on /home (/dev/hdb1)
                                Blocks
User ID      Used    Soft    Hard Warn/Grace
-----
root         4.6G      0       0   00 [-----]
testuser     103.4G     0       0   00 [-----]
...
```

- Running report as an ordinary user simply reports that users disk usage while running as root allows the disk usage of all users to be reported.
- Performance
  - `sudo xfs_quota -xc 'report -h' /home` 0m0.287s
  - `sudo du -hs /home` 0m3.086s



# Quota Limits

- User Limits

- `limits xfs_quota -x -c 'limit bsoft=1000m bhard=1200m username'`
- `limits xfs_quota -x -c 'limit isoft=500 ihard=700 username'`

- Group Limits.

- `xfs_quota -x -c 'limit -g bsoft=1000m bhard=1200m groupname'`

- Project Limits

- Project controlled directories are first added to `/etc/projects`. Project names are optionally added to `/etc/projid` to provide an id/name mapping.
- `xfs_quota -c 'project -s projectname'` is run to initialize project directories.
- `xfs_quota -x -c 'limit -p bsoft=1000m bhard=1200m projectname'`

# Enabling/Disabling Quotas

- `xfs_quota` can be used to temporarily or permanently disable quotas on a filesystem using the following commands:
  - `disable` – Temporarily disable quota enforcement.
  - `enable` – Enable quota enforcement. Used to enable quotas if they have been disabled or the filesystem was mounted with `qnoenforce`.
  - `off` – Switch off quota management quotas can only be re-enabled by un-mounting and remounting the filesystem
    - Limit information is preserved
  - `remove` – remove quota extents from a filesystem. All quota limits are destroyed.

# Expert xfs commands

- xfs\_quota also contains commands to
  - Summarise the state of xfs quotas on a system using the *state* command.
  - Manipulate the grace period using the *timer* command.
  - Backup and restore the quota information contained in a filesystem using the *dump* and *restore* commands. These commands are also used internally by xfsdump

# Generic Quota Tools

- In addition to xfs\_quota xfs also works with generic quota tools provided on Linux.
- These tools include
  - quota
  - repquota
  - quotactl
  - edquota
  - quotacheck
  - setquota
  - quotaon/quotaoff (enforcement only)
  - quotawarn
- The generic tools do not understand XFS project quotas.

# Providing user feedback

- Usually systems are configured to provide a level of feedback to users if they exceed their quota limits.
- The most common method for informing users is to send email using the quotawarn command from the generic quota package.
  - Quotawarn is often run as a cron job daily.
- Feedback can be provided for interactive users by running “xfs\_quota -c quota” in users login scripts (/etc/bash.bashrc.local for example).

# Implementation

- Quota operations and data are treated as first-class operations in XFS
  - xfs quotas are implemented as part of the filesystem metadata instead of maintaining quota data in a file within filesystem
  - Quota changes are journaled
- Quotas are updated in realtime as the filesystem is changed. No “quota checks” need to be run.

## Implementation – Project Quotas.

- Project quotas are implemented using the group quota inode. This excludes project and group quotas from being enabled at the same time.
- Project configuration information is stored in /etc/projects and /etc/projid.
- `xfs_quota -s projectname` initializes the project id in inodes contained under directories specified in /etc/projects.

# Quotas Ondisk

- xfs\_db can be used to understand the internals of the quota implementation.
- Quota inodes can be seen in the filesystem superblock

```
# xfs_db -xr -c sb -c p /dev/hdb1
```

```
...
```

```
uquotino = 131
```

```
gquotino = 148
```

```
# xfs_db -xr -c 'inode 131' -c p /dev/hdb1
```



# Quotas Ondisk

- Individual quota entries can be viewed with xfs\_db's dqquot command.

```
# xfs_db -xr -c 'dqquot username' -c p /dev/hdb1  
# xfs_db -xr -c 'dqquot -p projectname' -c p /dev/hdb1
```

- inodes can be interrogated to determine which project they belong to.

```
> ls -ia  
> sudo xfs_db -xr -c 'inode 675234191' -c p /dev/hdb1  
...  
core.projid = 33  
core.uid = 0  
core.gid = 0
```

**sgi®**