"Quotas for Projects"
A Proposed New Feature

Shuichi Ihara
Li Xi
DataDirect Networks Japan
Lustre Quota

- Quota is the most basic but useful storage management mechanism
- Lustre has been supporting Quota since Lustre-1.4
- It's cluster wide UID/GID based Quota and scalable.
- Quota codes are cleaned up and updated in Lustre-2.3.
- Lustre's use cases are extending which makes UID/GID Quota limit become insufficient. (e.g. “Project” oriented directory or files)
What is "project" and "Quota for project"?

- Some people belong to project1, other some people belong to project2, or maybe both project1 and project2 and use specific directories.
- Administrator wants to control Quota limit per project
- UID/GID based quota doesn't help
Why not OST pool help?

- Some cases maybe work.
- It's not Quota and size limitation can be only controlled by N x OST.
- Big performance impacts if number of OST is small.
- Can't control if real OST pool is exist (e.g. SSD pool)
A Quick Summary of Project Quota

- A growing needs of "project" oriented quota control
  - user/group quota doesn't work in some use cases. (e.g. project based storage volume allocation)
  - Quota for small groups in a Filesystem helps administrator to make a capacity plan of entire storage's volume
  - XFS supports per-directory or per-project quota and GPFS also supports fileset based quota which is conceptually similar
  - Patch which introduces subtree quota support for ext4 has existed for years, but was not merged.
Architecture of Quota in Lustre

► Quota “slaves”
  • All the OSTs and MDT(s) are quota slaves
  • Manage local quota usage/hardlimit and acquire/release quota space from the master

► Quota “master”
  • A centralized server hold the cluster wide limits
  • Guarantees that global quota limits are not exceeded and tracks quota usage on slaves
  • Stores the quota limits for each uid/gid
  • Accounts for how much quota space has been granted to slaves
  • Single quota master running on MDT0 currently
Architecture of Quota in Lustre

Clients

LOV

Bulk write request/reply

MDS(QMT)

DQACQ request/reply

OSS(QSD)

Local quota check
New quota type "Project" requirements

► Integrated in current quota framework
  • Support new accounting (new id) as well as today’s UID/GID
  • Ability to enforce both block and inode quotas
  • Support hard and soft limits
  • Same CLI to manage Quota limit
► Keep compatibility with old versions of Lustre
► No significant performance impact
Subtasks of "Quota for Project"

Lustre-2.6/Lustre-2.7

- Today's Lustre Quota
- Cleanup Lustre Quota

Lustre-2.7 and 2.8

- Update UserSpace tool
- Enhancement of llog
- Cleanup Lustre Quota
- New quota type in OSDs
Cleanup the Quota codes

► Current quota codes only recognize UID/GID quota, and not extendable enough for new quota type
  • Involved software: general quota support of Linux kernel, ext4, zfs, Lustre, e2fsprogs, quota-tools, glibc head file

► Finished work
  • Cleanup patch of e2fsprogs is pushed into ext4 community
  • Cleanup patch of Lustre is pushed into Lustre community
Project/Directory Quota support in OSDs

- Today, ext4 only supports UID/GID based quota
- It has been some discussion on linux-ext4 alias to support "project quota" in ext4.
- Recently, RFC of "project Quota" was posted by Zheng Liu.
- Patches of "subtree based quota" was posted by Dmity Monakhov, 2012. (http://lwn.net/Articles/506064/)
- ZFS supports subtree level quota as well as UID/GID based quota
Management of "project" id/name

- Need a cluster-wide mechanism to manage project
  - Need list/add/remove/set-quota/get-quota operations for daily management
  - User/Group management already has local /etc/passwd /etc/group databases and LDAP
1. Use "(OST) pool name" as "project name"
   - Save both name and its unique ID of pool into llog and use them as project id/name
   - It's simple way to do, but need to be careful of compatibility.
   - The OST pools can manage project id/name as well as traditional OST pool.
2. Develop all "project id/name" management mechanism
   - Store Project ID/Name into llog and create new management tool for them
   - A database of map for project id and "directories/files"
   - It is a bit complex to implement
3. And more...
Prototype of "Quotas of Projects"

- Since "Quotas of Projects" is complex, we are working on make a prototype version of "Project Quota" feature.
- We re-used "subtree" quota patches into EXT4 to support project/directory Quota in the backend filesystem
- Adding OST pool id and use pool id/name as project id/name
  - Simplest implementation, but less compatibility
  - Full functionality, but only based on ldiskfs
  - Reusable codes even design changes
Prototype: Flow of setting pool ID

Client
- Create a new file
- Write data to the file for the first time
  - LOV
  - Get pool ID from pool name

MDS
- Inherit pool attribute from parent directory
- LOD
- Map pool name to pool ID
- OSD
  - Save pool ID as project ID of that file
  - Pool ID

OSS
- OSD
  - Save pool ID as project ID of the data objects
Prototype: Data Flow of Write Request

Client
- Send RPC to write to an object of a file

OSS
- Lookup quota entry though pool ID from object xattr
- Check whether local quota is enough
  - Yes: Send RPC to acquire more quota from master
  - No: Grant space quota to the slave
- Write data to object
- Write RPC completed

MDS
- Lookup quota entry though pool ID from RPC
- Send reply to client
Prototype:
Status

► The most of components are done
  • Cleanup and refresh of subtree support patches in EXT4 and LDIFKS
  • Cleanup E2fsprogs
  • Cleanup Lustre Quota

► A prototype is ready in End of April, 2014
  • Full functionality as well as user space utilities, but only based on ldiskfs
  • llog changes to store new pool id, but less compatibility

► LU-4017 quota: Add pool support to quota
Prototype:
How to use project based quota

► Enable Quota
   
   # lctl conf_param lustre.quota.ost=ugp; lctl conf_param lustre.quota.mdt=ugp

► Create new pools
   
   # lctl pool_new lustre.proj1; lctl pool_new lustre.proj2
   # lctl pool_add lustre.proj1 lustre-OST[0-3]; lctl pool_add lustre.proj2 lustre-OST[0-3]

► Assign pool to an directory
   
   # lfs setstripe -p proj1 /lustre/proj1

► Set/Check Quota to pool
   
   # lfs setquota -p lustre.proj1 -b10240 -B 20480 –i 2000 –I 3000 /lustre
   # lfs quota -p lustre.proj1 /lustre

► Enforced Quota
   
   # dd if=/dev/zero of=/lustre/proj1/file bs=1M count=100
   dd: writing `/lustre/proj1/file': Disk quota exceeded
   # lfs quota -p lustre.proj1 /lustre
   Pool: lustre.proj1
   Disk quotas for subtree lustre.proj1 (sid 319124861):
   
<table>
<thead>
<tr>
<th>Filesystem</th>
<th>kbytes quota</th>
<th>limit</th>
<th>grace</th>
<th>files</th>
<th>quota</th>
<th>limit</th>
<th>grace</th>
</tr>
</thead>
<tbody>
<tr>
<td>/lustre/proj1</td>
<td>19460* 10240 20480 1w 1 2000 3000 -</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Further work

- Compatibility with older versions
  - LLOG record format has changed?
  - Disk format has changed by adding project id into EA
  - Quota control API has changed
  - Wire format has changed

- Clustered meta-data support
  - MDT pool support of quota

- ZFS supports

- Update test suites and supports "project" quota
Summary

► We designed new quota type "Quotas for Projects"
► Many new use cases will be available with this new Quota type (Project Quota)
► A prototype version of codes are ready very soon
► Continue to have discussion based on prototype. The most of codes should be re usable even changes the design.
► Any feedback are Welcome!