

Lustre HSM

**Arun
Ramakrishnan
Technical Lead,
Storage Software
Engineering**

©2012 SGI



Lustre HSM Architecture

- Policy Engine.
 - Implements space management policies.
 - Sends requests to HSM coordinator.
 - Uses Lustre changelogs instead of filesystem scans.
 - Robinhood engine used for default config.
 - Manages pre-migration and purge policies.
- User support commands.
 - Invoke HSM operations like migrates, recalls etc.
 - Sends requests to HSM coordinator
- HSM Coordinator.
 - Processes requests from policy engine and user commands.
 - Dispatches incoming requests to HSM Agents.

Lustre HSM Architecture contd..

- HSM Agent
 - Processes HSM coordinator requests.
 - Runs on one or more Lustre clients.
 - Runs custom copytool to transfer data.
 - Informs Lustre core about request progress periodically.
- HSM backend
 - Custom application managing secondary storage.
 - Maps lustre HSM request into backend requests using custom copytool.

Lustre File ID

- Persistent and unique on a per filesystem basis.
- HSM backend must be able to map File IDs to native internal objects.
- File ID logically mapped in filesystem namespace to ease file metadata and data access.
 - `<mount-point>/.lustre/fid/<fid_string>`
- File ID can be mapped to actual lustre pathnames using `ioctl`s.

Lustre HSM states

- Archived (Dual State)
- Released (Offline)
- Exists
 - File modified while being archived.
- Dirty
 - File modified since archival.
- Lost
- No-Release / No-Archive

Lustre HSM Operations

- Archive (Migrate)
 - Store copy in secondary storage.
 - Attrs (except for stripe info) preserved in lustre object after archival.
 - Optionally backup extended attrs, stripe info, path and posix attrs in HSM backend for disaster recovery.
- Restore (Recall)
 - Retrieve copy from secondary storage.
 - Optionally restore stripe info, extended attrs.

Lustre HSM Operations contd ...

- Remove (Soft Delete)
 - Remove data from lustre and mark secondary storage copy for removal.
 - Logically equivalent to destroy dmapi event.
- Cancel
 - Cancel request using given File ID.
- Import
 - Import file data from existing HSM backend into new Lustre filesystem.
- Recover
 - Recover a given File ID from existing HSM backend into new Lustre filesystem.

Backup using Robinhood-backup

- Custom policy engine developed by CEA.
- Registered as a changelog reader.
- Uses MySQL database for managing file state.
- Uses remote cp/rm to perform the data movement via agents.
- Can NFS mount exported DMF filesystems from CXFS clients and issue data movement commands.

Lustre HSM Integration Status

- Lustre HSM Client support in Lustre 2.4.
- Lustre HSM Server support planned for Lustre 2.5.
- Dmarchive/Robinhood-backup can be used to manage lustre data using DMF.
- Copytool C++ interfaces need to be stabilized in order to complete DMF copytool effort.
- Additional tools needed to import DMF data directly into new Lustre filesystems.
- Need Policy Engine mechanisms to send data to specific DMF VGs/MGs.
- DMF I/O stack needs to be optimized for dealing with lustre filesystems.
- Lustre awareness in dmfsapi layer.

Misc Lustre HSM User Commands

- hsm_state
 - Determine state of file (dmls equivalent).
- hsm_set / hsm_clear
 - Set/Clear flags like norelease/noarchive.
 - Some flags need root privileges.
- fid2path / path2fid
 - Map between Lustre File IDs and Lustre pathnames.

