Lazy Size on MDS

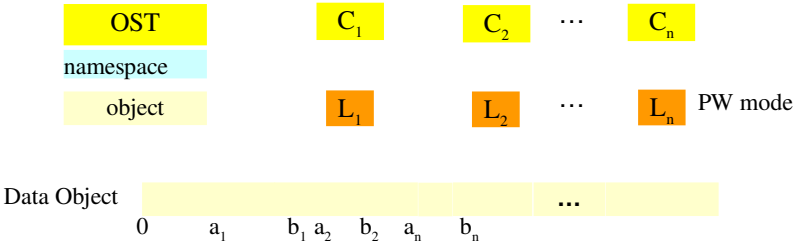
DataDirect Networks

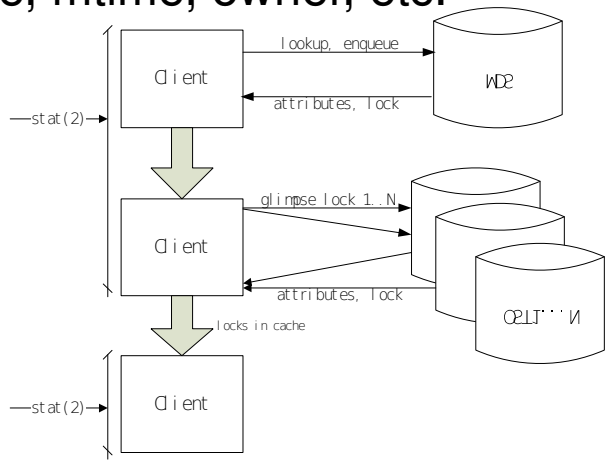
Li Xi, Dongyang Li

Why SOM?

Current approach: size on OSTs

- MDS stores some file metadata: ctime, mtime, owner, etc.





- Size and blocks information are obtained from OSTs via size glimpse lock callback.
- If file is striped into N data object, it needs N RPCs to get size and blocks for a file.
- Total N + 1 RPCs to get file attributes.
- result in “ls -l” is slow on large directory of a large system.

LSOM design

- The LSOM is saved as an EA value on MDT.
- LSOM includes both the apparent size and blocks.
- Whenever a file is being truncated, the LSOM of the file on MDT will be updated.
- Whenever a client is closing a file, it sends the size and blocks to MDS. The MDS will update the LSOM of the file if the size has been increased.
- A helper tool to sync file LSOM xattr periodically by using Lustre changelog mechanism.

Why Lazy?

Strict/Accurate SOM makes the recovery very complex

- **Keep the implementation as simple as possible.**
- **No guarantee of LSOM accuracy:**
  - A file being opened for write/append might make LSOM inaccurate.
  - Eviction or crash of client might cause incomplete process of closing a file, thus inaccurate LSOM.
- **A precise LSOM could only be read from MDT when:**
  - All possible corruption and inconsistency caused by client eviction or client/server crash have all been fixed.
  - The file is not being opened for write/append.

Client support

- Client is not aware of LSOM yet.
- lfs getsom <path>
  - Will print the size and block info on client after retrieving LSOM xattr from mdt
- getxattr/lgetxattr/fgetxattr
  - Ask for “trusted.som”

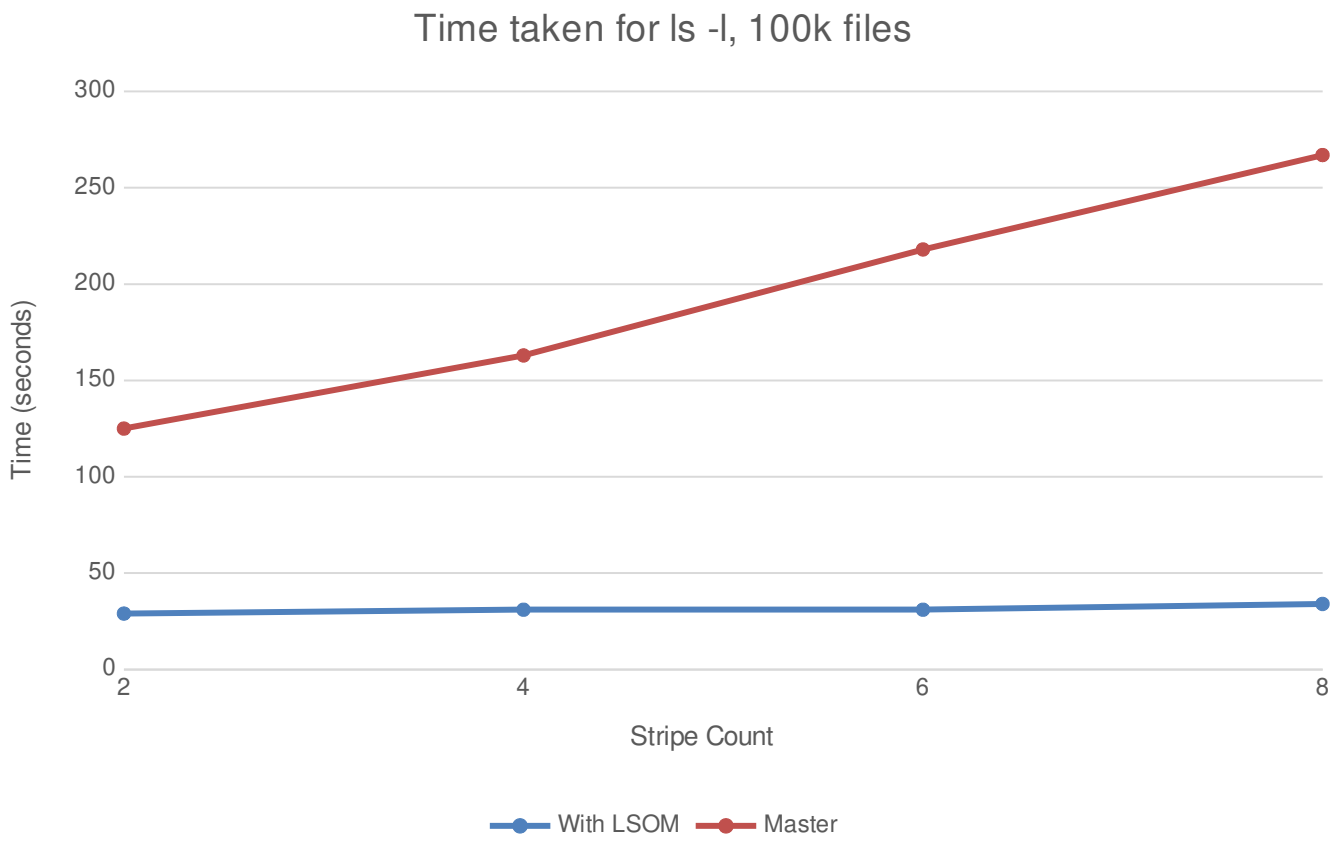
Use cases

- **statx(2) with AT_STATX_DONT_SYNC**
  - Don’t synchronize anything, just take whatever the system has cached if possible.
  - This information returned is approximate.
  - Naturally the MDT could return LSOM to the client, saving rpc round trips to OSTs.
- **archive/purge/placement decisions based on LSOM**
  - Robinhood
  - Lustre Integrated Policy Engine scans MDTs directly
    - No extra server/storage
    - No metadata duplication
    - No way to obtain object size currently

Future work

- **Add confidence flags stored on MDS for both size and blocks on a per-file basis:**
  - SOM_FL_ROUGH: Approximate, LSOM presents
  - SOM_FL_STALE: was right at some point in the past, but may be wrong now (e.g. opened for write)
  - SOM_FL_STRICT: known correct, FLR or DoM file
  - SOM_FL_UNKOWN: Unknown/no SoM, must get size form OSTs.
- **Add mount options that make the stat() behavior selectable:**
  - mount –o lazy_stat
- **Add IOCTL to get LSOM for policy engines or space rebalancing.**

Performance



LU-9538

https://review.whamcloud.com/#/c/29960/

https://review.whamcloud.com/#/c/30124/

Thank you!