Lustre-HSM

Progress report on Lustre HSM and proposal for a new design of the coordinator
Introduction: coordinator's design
Contended design

- **client**
- **client**
- **client**

- **syscalls**

- **hsm layer**

- **llog**

- **coordinator_thread**

- **coordinator**

- **agent**
- **agent**
- **agent**
History (~6 months)

bugfix / performance / feature

- f3a4152 LU-7988 hsm: update many cookie status at once
- afc9ff6 LU-7988 hsm: added coordinator housekeeping flag
- cc6ef11 LU-7988 hsm: run HSM coordinator once per second at most
- f11a502 LU-7988 hsm: mark the cdt as stopped when its thread exits
- dd4b034 LU-4640 mdt: implement Remove Archive on Last Unlink policy
- 65effa6 LU-9338 hsm: cache agent record locations
- 2d5babe LU-9312 hsm: convert cdt_llog_lock to a rw semaphore
- 958198e LU-7988 hsm: change coordinator start/stop mechanisms
Performance improvements

Lower is better

Before

Freq = 80 req / s
Freq = 160 req / s
Freq = 240 req / s
Freq = 320 req / s
Freq = 400 req / s
Freq = 480 req / s

Now

Freq = 80 req / s
Freq = 160 req / s
Freq = 240 req / s
Freq = 320 req / s
Freq = 400 req / s
Freq = 480 req / s

Lower is better
Performance improvements

Upper is better

Lower is better

- Number of requests in 60 seconds
  - Before: 10k requests
  - Now: 35k requests

- Time to process 10k requests
  - Before: 16
  - Now: 8

Upper is better (higher number of requests)
Lower is better (lower time to process)
Remaining issues

• Still not scalable (limited at 500 req/s)

• Current behaviour does not fit every workload

• Bugs: LU-5216, LU-5896
Current design

client → syscalls → hsm layer

hsm layer → llog

client → coordinator_thread

coordinator

agent

agent

agent
Current design

- client
- syscalls
- hsm layer
- llog
- coordinator_thread
- agent
- coordinator
Current design

client

syscalls

hsm layer

llog

coordinator_thread

agent

coordinator
Current design

- Client
- Syscalls
- HSM layer
- Coordinator_thread
- Llog
- Coordinator
- Agent
Current design

client

syscalls

hsm layer

coordinator_thread

lllog

agent

coordinator
Current design

client

syscalls

hsm layer

llog

coordinator_thread

agent

coordinator
Current design

client

syscalls

hsm layer

agent

llog

coordinator_thread

coordinator
Current design

client

syscalls

hsm layer

llog

coordinator_thread

agent

coordinator
Current design

- client
- syscalls
- hsm layer
- coordinator_thread
- llog
- agent
- coordinator
Current design

client

syscalls

hsm layer

llog

coordinator_thread

coordinator

agent
Current design

- client
  - syscalls
    - hsm layer
      - llog
      - coordinator_thread
        - coordinator
  - agent
Current design

- **client**
- **syscalls**
- **hsm layer**
- **llog**
- **coordinator_thread**
- **agent**
- **coordinator**
New structures

**scheduler:**
- declare_request
- cancel_request
- register_agent
- unregister_agent

**store:**
- declare_request
- cancel_request

**iterator:**
- next
Proposal

client ➔ syscalls ➔ hsm layer ➔ store ➔ scheduler ➔ coordinator

client ➔ syscalls ➔ hsm layer ➔ agent

client ➔ syscalls ➔ hsm layer ➔ agent

client ➔ syscalls ➔ hsm layer ➔ agent
Proposal

Diagram showing the flow from client, through syscalls, hsm layer, store, scheduler, and finally to agent and coordinator.
Proposal

client

syscalls

hsm layer

store

scheduler

basic fifo priority

agent

coordinator
Proposal

client

syscalls

hsm layer

store

scheduler

basic fifo priority

agent

coordinator
Proposal

client

syscalls

hsm layer

store

scheduler

basic  fifo  priority

agent

coordinator
Proposal

client

syscalls

hsm layer

store

scheduler

basic fifo priority

agent

coordinator
Proposal

client

syscalls

hsm layer

store

scheduler

basic  fifo  priority

agent

coordinator
Proposal

client

syscalls

hsm layer

store

scheduler

basic

fifo

priority

agent

coordinator
Pros and cons

- scalable
- tunable / improvable
- cleaner code
- new features

- hard to integrate
- lots of coding / review
Thank you!
Integration planning

1. Clean split between hsm and mdt
2. General code cleanup
3. Remove completed requests from the llog
4. Stop storing the requests' state in the llog
5. Replace the llog (?)
6. Move on to loadable policies