#### Coordinatool

A copytool to rule them all

**Dominique Martinet** 

dominique.martinet@codewreck.org

September 28-30, 2021

Dominique Martinet | September 28-30, 2021 | PAGE 1/29



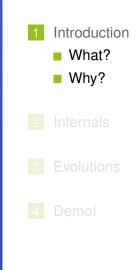
Introduction

Evolutions

Demo!

# Big thanks to CEA for organizing LAD and funding this development and talk

(And also to everyone involved in the lustre community for making this possible)





Introduction

Evolutions

- A lustre copytool...
- that acts as a userspace coordinator...
- and gives work to real, existing copytools through LD\_PRELOAD overloading

#### Overview

Introduction

Evolutions

Demo!

#### Two main components:

- Lustre copytool "server":
  - accept requests from lustre
  - schedules requests to coordinatool clients
- LD\_PRELOAD client library
  - overloads llapi HSM calls
  - allows using existing copytools as coordinatool clients
- administrative client
  - query status, lock states, requeue lost requests...

Still WIP (young "free time" project), but already works



Reaching the limits of in-kernel coordinator

Set limits for each request type per agent

CEA agents allow more remove than the rest: lustre keeps banging its head on EAGAINs

Better request scheduling, retries...

 Stepping stone for the real userspace coordinator work (LU-13384)



- 2 Internals
  - Lustre copytool "server"
  - Client protocol
  - LD\_PRELOAD client
  - coordinatool\_client
  - Tricky bits





#### Server code flow

Introduction

Internals

Evolutions

Demo!

Single thread process:

- register llapi copytool
- bind/listen/accept TCP connections from clients
- epoll loop on llapi\_hsm\_copytool\_get\_fd/clients
  - Receive & process HSM requests
  - Reply to client requests
  - Hopefully won't block...

#### Server in depth

Introduction 0000 Internals

Evolutions

- What do we use to queue actions ?
  - performance bottleneck with catalogs on real coordinator
  - might as well use it for recovery: work in memory

#### Quick reminder of lustre types: HAI

Introduction

Internals

Evolutions

struct hs	sm_action_item {
u32	hai_len; /* valid size of this struct */
u32	hai_action;
struct	<pre>lu_fid hai_fid; /* Lustre FID to operate on */</pre>
struct	lu_fid hai_dfid; /* fid used for data access */
struct	hsm_extent hai_extent;
	<pre>/* byte range to operate on */</pre>
u64	hai_cookie; /* action cookie */
u64	hai_gid; /* grouplock id */
char	hai_data[0];
}attri	ibute((packed));

#### Quick reminder of lustre types: HAL

Introduction

Internals

Evolutions

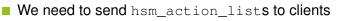
<pre>struct hsm_action_list {</pre>	
u32 hal_version;	
u32 hal_count; /* number of hais to follow */	
u64 hal_compound_id;	
u64 hal_flags;	
u32 hal_archive_id; /* which archive backend */	
u32 padding1;	
<pre>char hal_fsname[0]; /* null-terminated */</pre>	
<pre>/* struct hsm_action_item[hal_count] follows, aligned</pre>	
on 8-byte boundaries. See hai_zero */	
}	

#### Queueing actions: requirements 1/2

Introduction

Internals

Evolutions



- queues must group by identical hal flags/archive\_id
- for now (v0), single queue only checking identity on append
- code is structured to allow multiple queues:
  - hsm\_action\_queues\_get(internal\_state, archive\_id, flags, fsname)

#### Queueing actions: requirements 2/2

Introduction

Internals

Evolutions

Demo!

#### We need to quickly find a request by cookie if required

- for HSMA\_CANCEL (TODO)
  - easy if not sent yet to a client
  - find which client if there is one
- deduplication for recovery mechanism

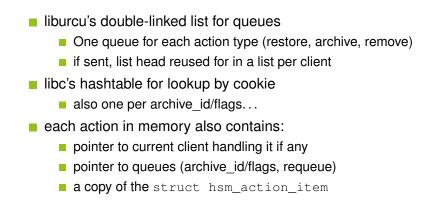
Lustre copytool "server"

### Queueing actions: implementation

Introduction

Internals

Evolutions



#### **Client protocol**

Introduction

Internals

Evolutions

- simple json serialization
- server only replies to clients
  - possibly not right away (e.g. no work to distribute)
- code is shared as much as possible
  - both clients and server
  - a new client would only need to implement a few callbacks

#### Commands:

Introduction

Internals

Evolutions

Demo!

#### STATUS: query server info

- pending, running and processed action counts
- number of clients connected...
- RECV: request work.
  - Specify how many restore/archive/remove the client can process.
- DONE: report to the server that an action was processed
- QUEUE: push an hsm\_action\_list (more later)

## LD\_PRELOAD client

Introduction

Internals

Evolutions

Demo!

Overload llapi hsm functions

## LD\_PRELOAD client: overriden functions 1/2

Introduction

Internals

Evolutions

- llapi\_hsm\_copytool\_register:

   connect to server, open lustre root and .lustre/fid
   alloc hsm\_copytool\_private with non-lustre MAGIC

  llapi\_hsm\_copytool\_unregister: cleanup
- Ilapi\_hsm\_copytool\_recv: send RECV and wait

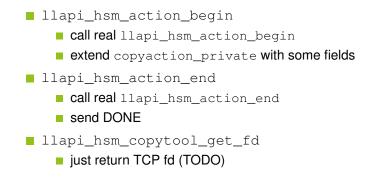
## LD\_PRELOAD client: overriden functions 2/2

Introduction

Internals

Evolutions

Demo!



That's it!

## LD\_PRELOAD client: why it works

Introduction

Internals

Evolutions

Demo!

- calling real llapi\_hsm\_action\_begin works on different client
  - Lustre does not care if a client different from the one which received request processes it
  - Don't need to do it on coordinatool/send temporary file fid...
- similarily, progress and other calls mostly work
- borderline bug: llapi\_hsm\_action\_begin does not check copytool\_private MAGIC

need mnt\_fd and open\_by\_fid\_fd at correct offset

#### coordinatool\_client

Introduction

Internals

Evolutions

- Intended for administrative tasks (stats...)
- Or debugging/tests
  - Request work and dump it on stdout
  - No done: server requeues work on client disconnect
- Very simple and unpolished (~250 lines of code)
- Mostly just reuse common init and protocol code

#### Tricky bits

Introduction

Internals

Evolutions

- Server restarting
- Client disconnecting
- Lustre tunings

#### Server restarting

Introduction

Internals

Evolutions

- Action queue only in memory
- Iustre doesn't handle a copytool disappearing really well
  - actions that had been sent are never resent
- mdt.fsname-MDT0xyz.hsm.active\_requests to the rescue
  - Parse the file and send it with "queue"
- Last problem: actions currently being handled by clients
  - (TODO) client writes in filesystem on action\_begin
  - read files on startup, give related actions a grace period

## **Client disconnecting**

Introduction

Internals

Evolutions

- currently requeues its processing actions immediately
- (TODO) give a grace period to reconnect ?
  - reclaim running actions on connect
  - trust service manager to not have duplicate on same host
  - later.

#### Lustre tunings

Introduction

Internals

Evolutions

Demo!

#### don't let requests expire

send requests to coordinatool ASAP

#### lctl set\_param -P

mdt.lustre0-MDT\*.hsm.active\_request\_timeout=31days

mdt.lustre0-MDT\*.hsm.loop\_period=1

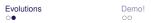
mdt.lustre0-MDT\*.hsm.max\_requests=1000



#### What next?

Introduction

Internals



- Finish TODOs listed earlier
  - server restart and HSMA\_CANCEL
- Waa-ay more tests
- Improve scheduling:
  - Got basic restore > rest like coordinator
  - Check file owners and don't let a user hog all agents

group requests by locality on tape?

- More commands? dump requests, lock/unlock...
- Lustre userspace coordinator API support
- The sky's the limit!









Dominique Martinet | September 28-30, 2021 | PAGE 28/29

#### Thanks!

Introduction

Evolutions

Demo! ○●

https://github.com/martinetd/coordinatool/

Tests, issues or PR welcome :D

Taking questions on the chat!