



# Lustre HSM Project

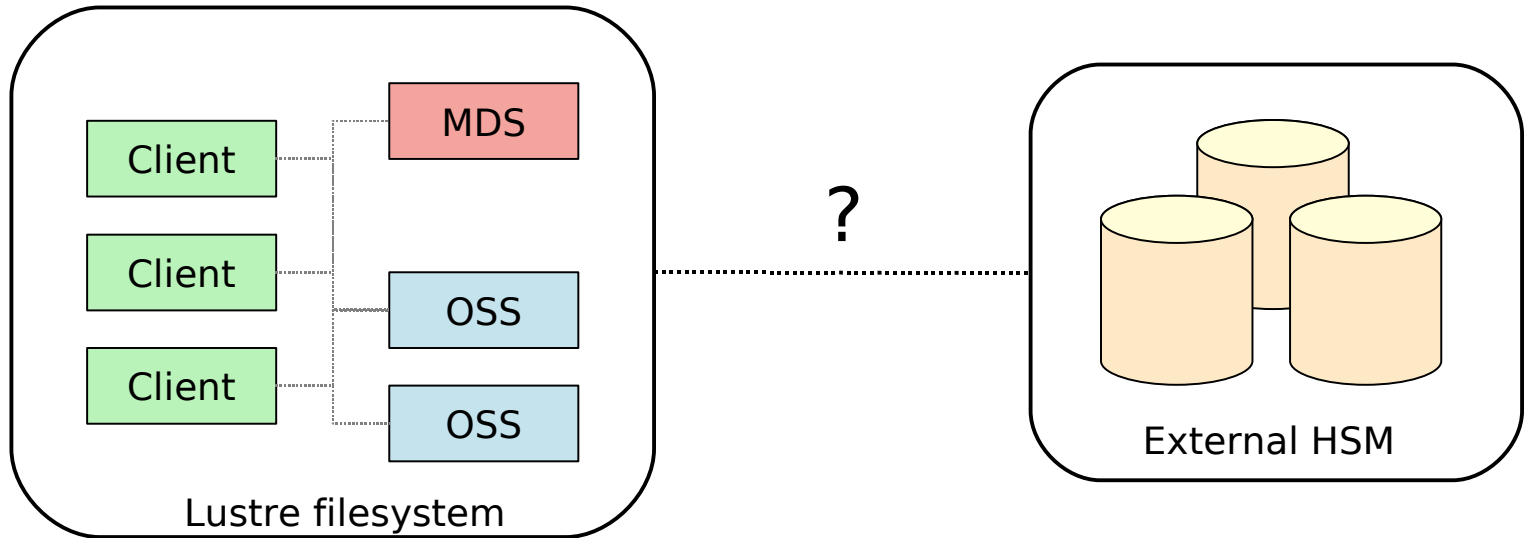
***Aurélien Degrémont***  
*aurelien.degreumont@cea.fr*

- Introduction
- Lustre HSM Project
- Architecture
- Features
- PolicyEngine/Robinhood
- HSM bindings
- Project status

# Introduction (1/2)



- **Two powerful components**



- Size from some TB to few PB
- Fast and parallel

- Size from lots of TB to many PB
- Slow data accesses

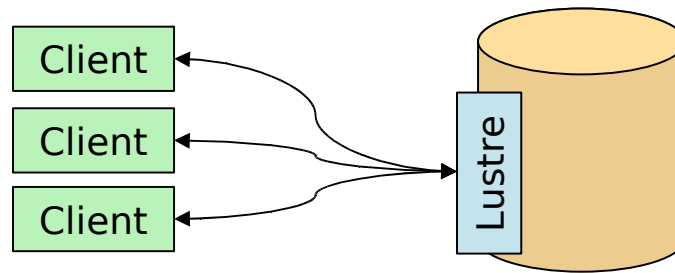
## Introduction (2/2)

---

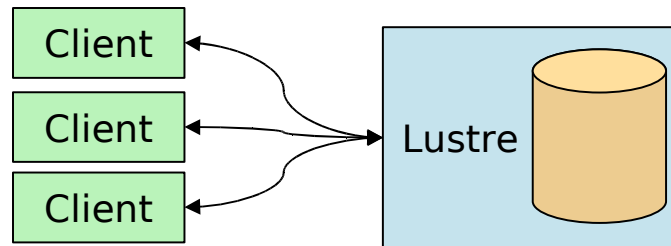


- **Interesting cooperations**

- Add a highly performant and scalable access to your HSM.



- Extend your Lustre disk space cheaply.



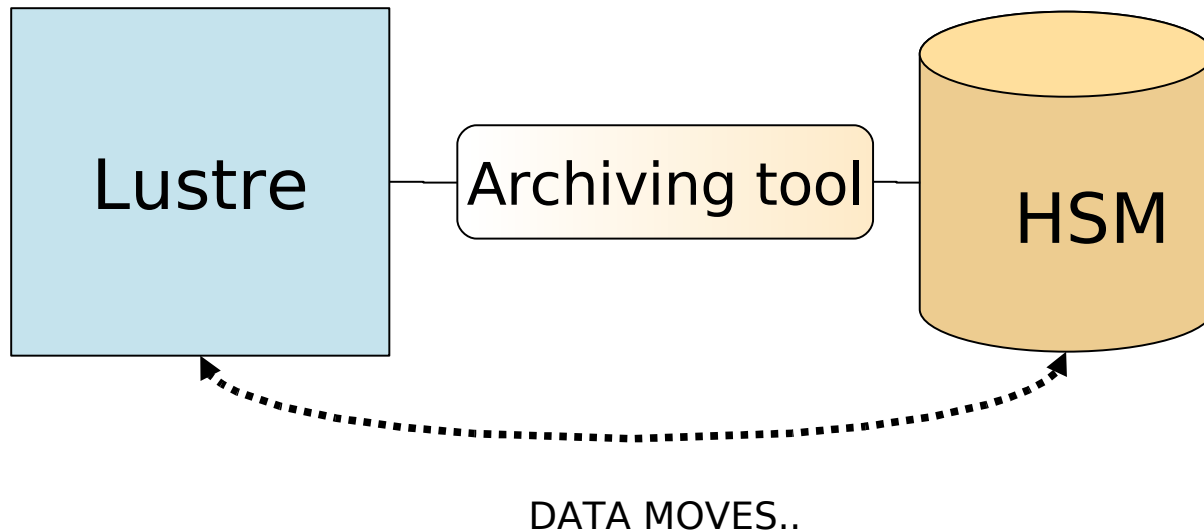
- Backup a Lustre data in a resilient and stable archiving system.

# Lustre HSM Project

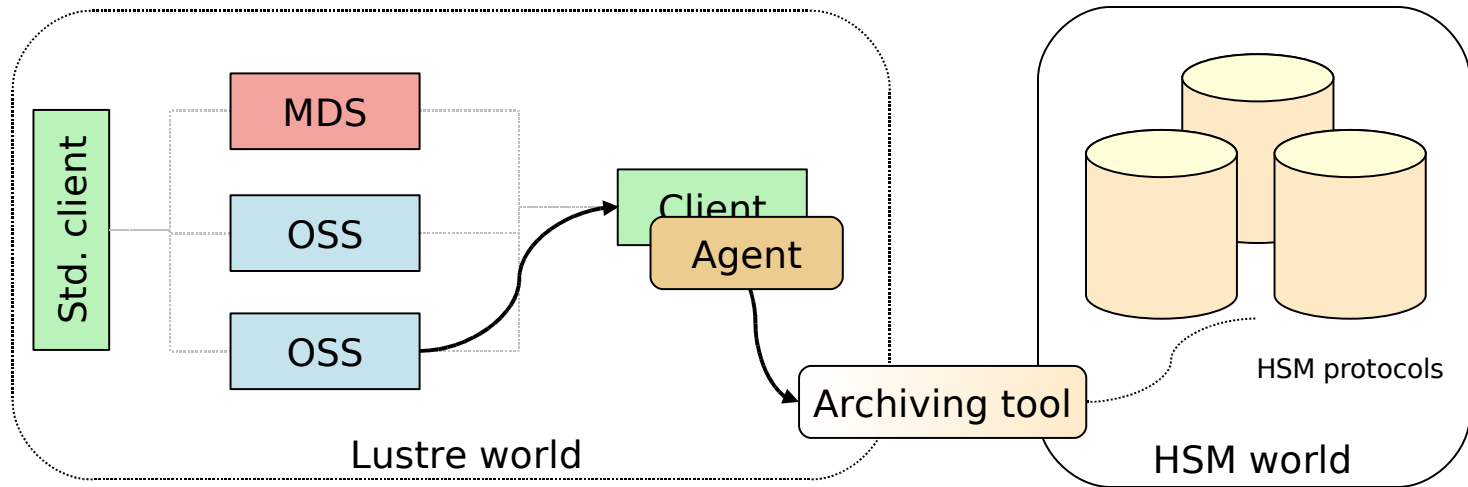
---



- Associate Lustre and an external archiving system.
- Migrate your file data between them.
- Data migration is managed by policies.



# Architecture: Copy it!



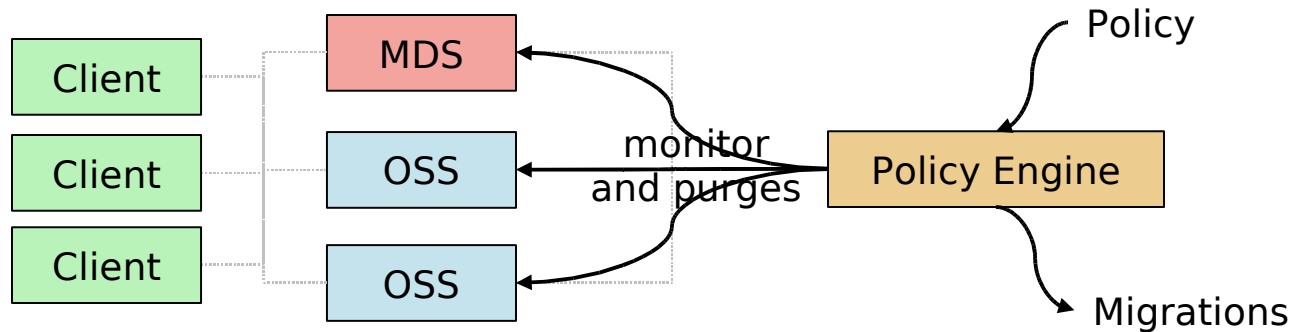
- **Agent**

- A service used to move data, to cancel such movement and to remove external storage files.

- **Archiving/Copy tool**

- Run on identified client nodes.
- Interface Lustre and the HSM.
- Know how to communicate with a specific HSM.

# Architecture: Purge it

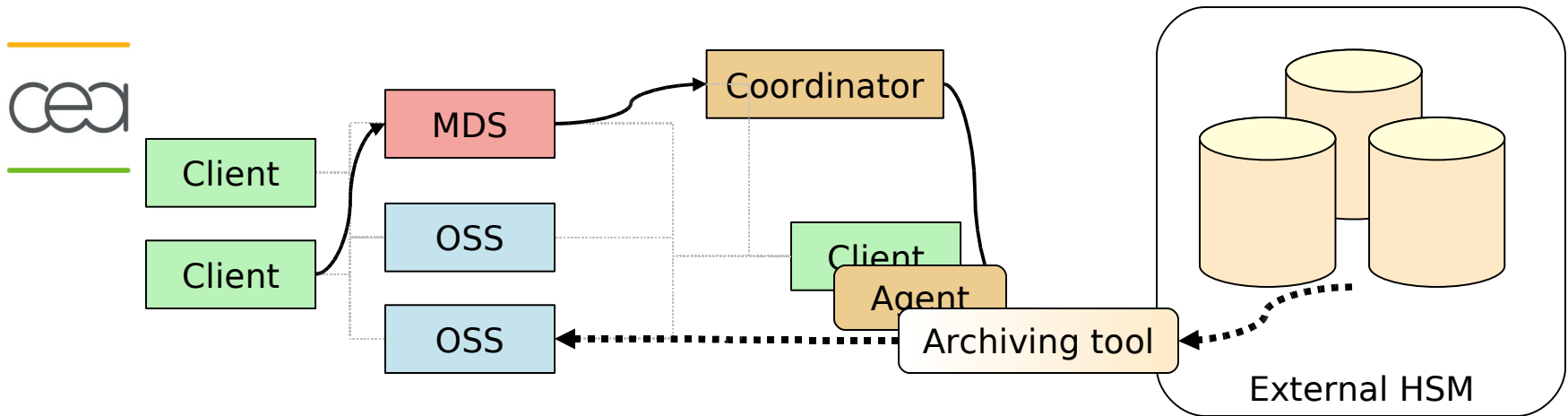


## ● Policy Engine

- Monitor filesystem disk space usage
- Pre-migrate not recently used data
- When space is lacking, purge data from files already copied in the

HSM

# Architecture: Bring them back



- **Initiating**

- Cache misses are detected on MDT and OST

- **Coordinator**

- Centralize migration requests
- Dispatch them on agents

## Features in more details

---



- **Main features for the first version will be:**
  - Whole file migration.
  - Copy data back at file open.
  - External PolicyEngine, like Robinhood, decides what to archive, what needs to be released from Lustre.
  - File aggregation possible if PolicyEngine and Copytool support it.
  
- **Next release plans to:**
  - Partial data released. Keep start and end of file in Lustre.
  - Copy data back at first I/O.



# PolicyEngine: Robinhood

---



- **Robinhood is a filesystem monitoring and purge tool.**
- **Project**
  - GPL-compatible, CEA development.
  - Website: <http://robinhood.sf.net/>
- **Main features**
  - Gather filesystem information from scans (POSIX scan) or event-based (Lustre 2.0).
  - Trigger file purges depending on policy rules.
  - Trigger file migration depending on policy rules (for Lustre HSM).
  - Policies based on file attributes and thresholds.

# Robinhood: Policy examples

---



- **Based on file attributes like: path, name, time, size, ...**
- **Rules could be combined with boolean operators.**
- **Files could be white-listed.**
- **Simple example:**

```
Purge_policy
{
    Trigger
    {
        trigger_on = OST_usage
        high_watermark_pct = 85%
        low_watermark_pct = 80%
    }

    Whitelist {
        last_access < 1h
        or
        owner = root
    }
}
```

# Binding Lustre with HPSS

---



- **Copytool: A user-space program managing data between Lustre and the HSM.**
  - Only use POSIX and few Lustre library calls.
- **CEA-HPSS copytool**
  - A CEA development.
  - Use HPSS API directly.
  - 3 tier architecture, for scalability purpose.
  - Will be freely available to all HPSS sites.
- **SAM-QFS copytool**
  - Sun will develop a copytool for Lustre/SAM-QFS binding.
- **DMF copytool**
  - SGI is also looking closely on this project

# Project status

---



## ● Project

- CEA/SUN collaboration

- ✚ Architecture design made by Lustre designers and CEA

- ✚ High Level Design/Detailed Design/Coding by community (CEA, SUN, ...)

## ● Development

- Architecture and High Level Design are done

- Detailed designs and prototypes are under progress

## ● Roadmap

- Feature availability for early adopters in Lustre 2.x

- Fully supported feature target is Lustre 3.0

- Early/beta code for end of summer '09

- Final version for end of 2009



**Questions ?**