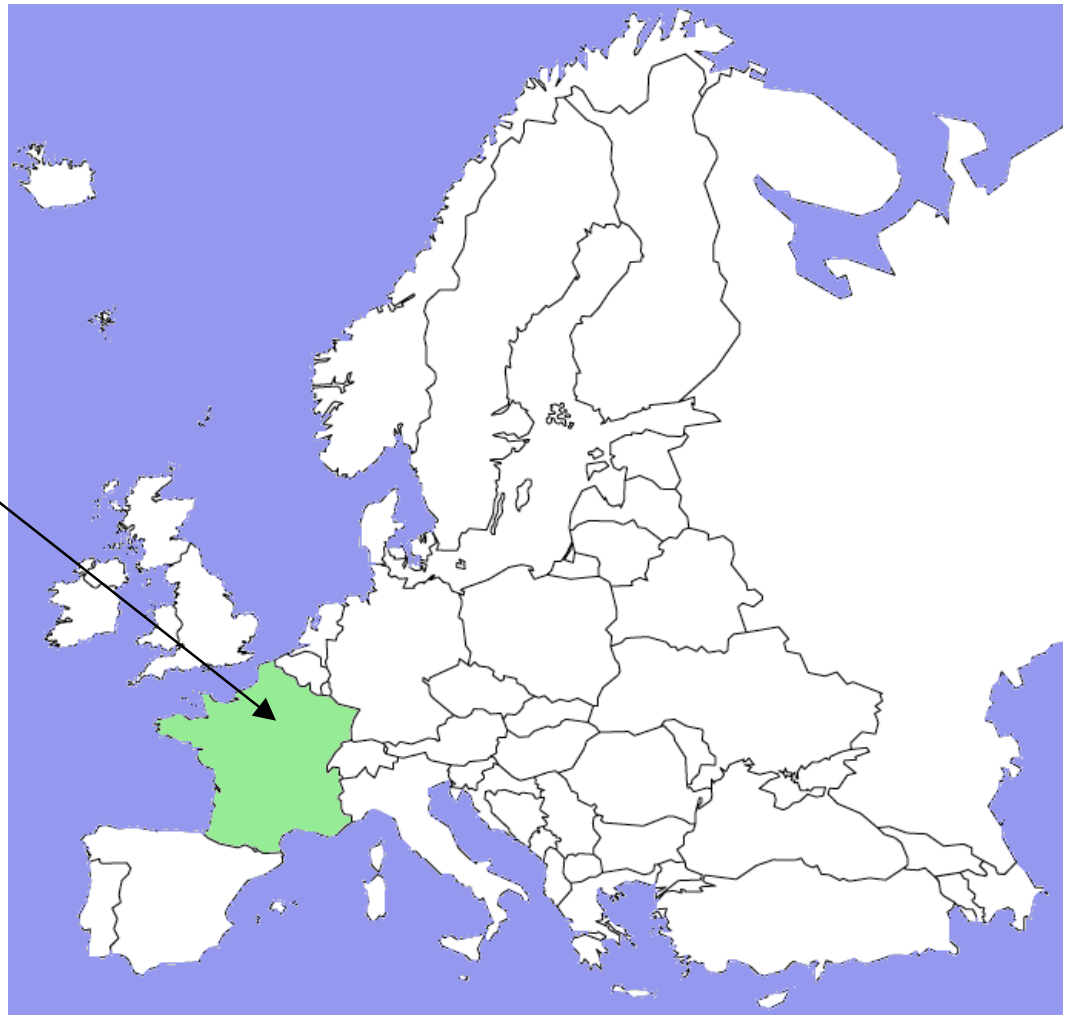
LCE Reports

French Atomic Energy Commission (CEA)

Stéphane Thiell
stephane.thiell@cea.fr

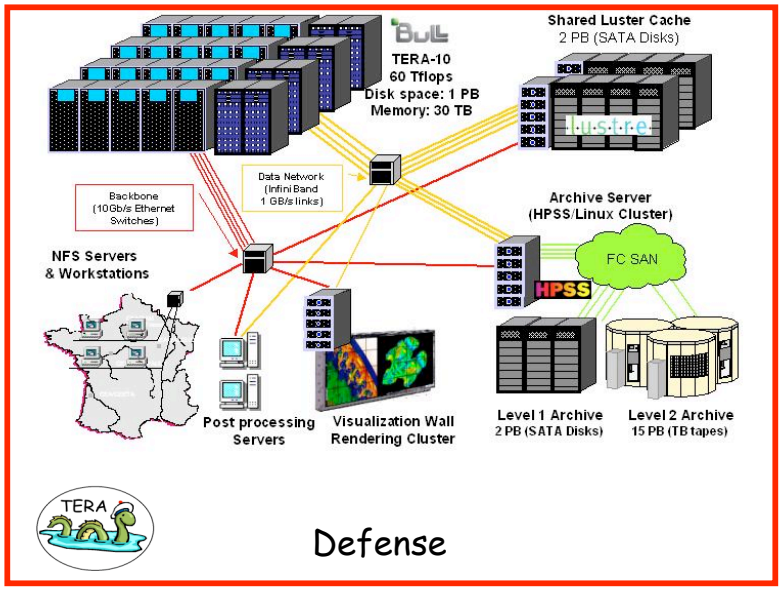
CEA Computing Center

CEA/DAM Computing Center



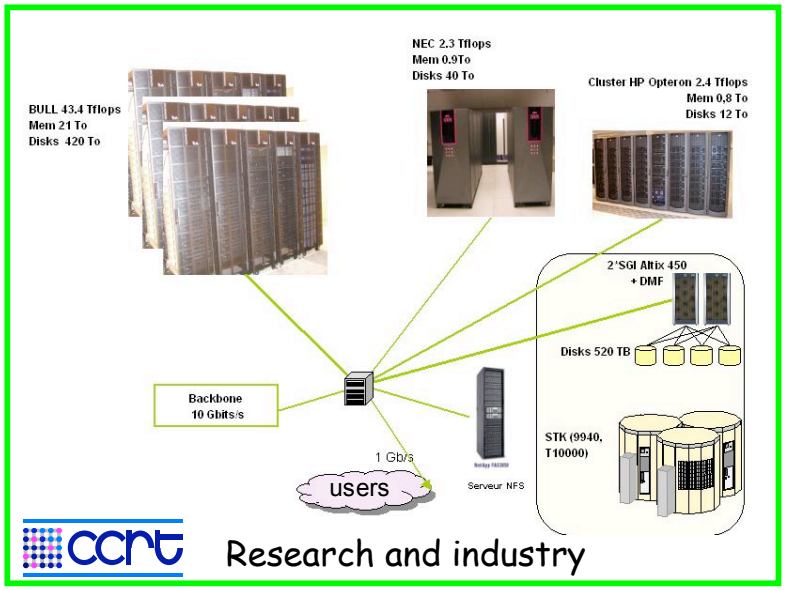
Current supercomputers at CEA/DIF

TERA-10



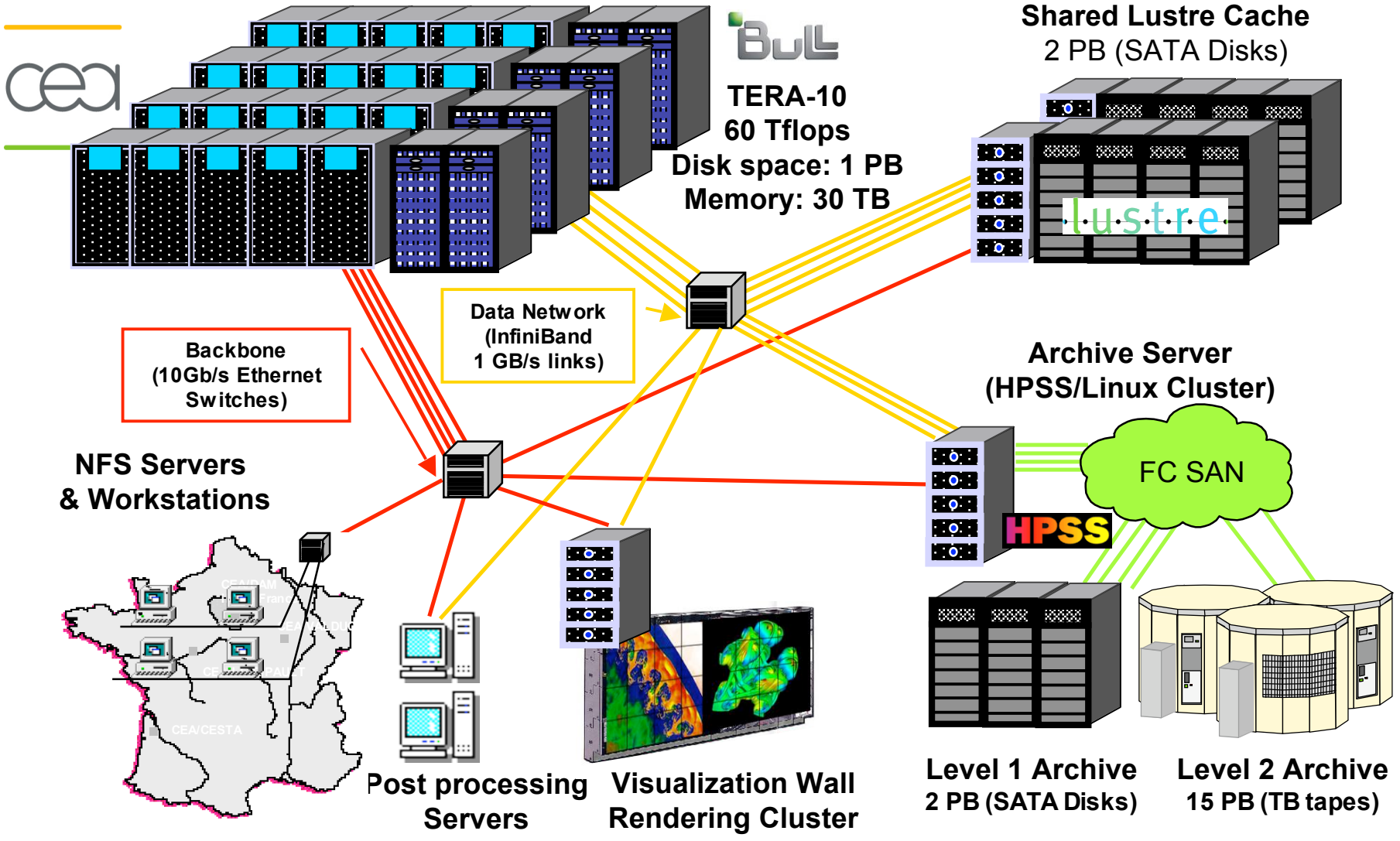
- Linux Kernel 2.6
- Lustre 1.4.7 and Lustre 1.6 Beta 7
- Quadrics and Infiniband interconnects

CCRT-B

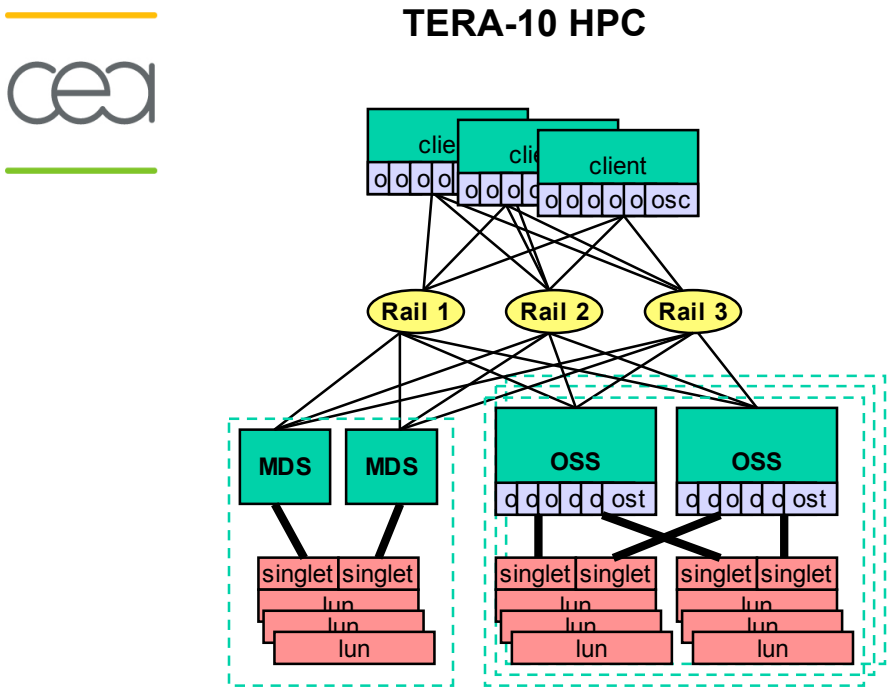


- Linux Kernel 2.6
- Lustre 1.4.8
- Infiniband interconnect

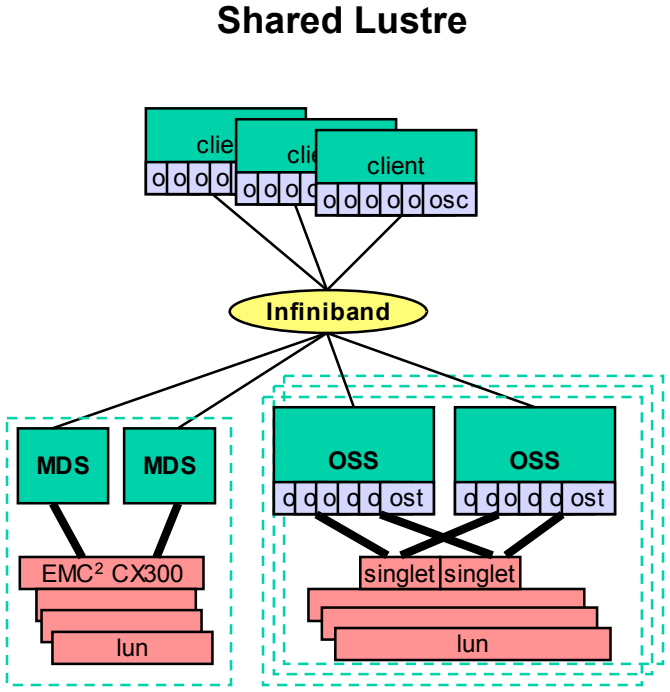
TERA-10 Computing Center Architecture



Two Lustre architectures



- Quadrics Elan4 interconnect, 3 rails each 900 MB/s
- DDN 9550, FC disks
- 16 LUNs of 1 TB (1 LUN/Tier)
- 565 clients



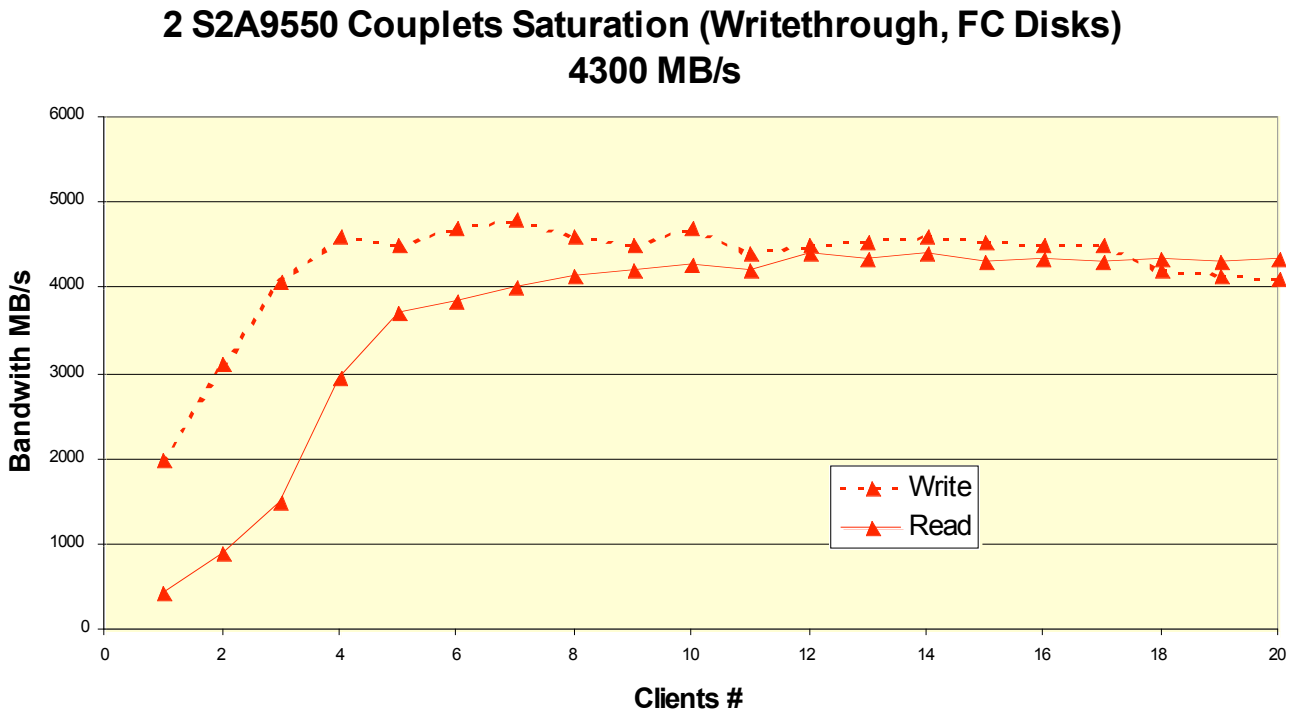
- Infiniband SDR interconnect
- DDN 9550, SATA disks
- 48 LUNs of 8 TB (2 LUNs/Tier)
- Use of device-mapper multipath
- ~60 clients

Lustre Usage (I)

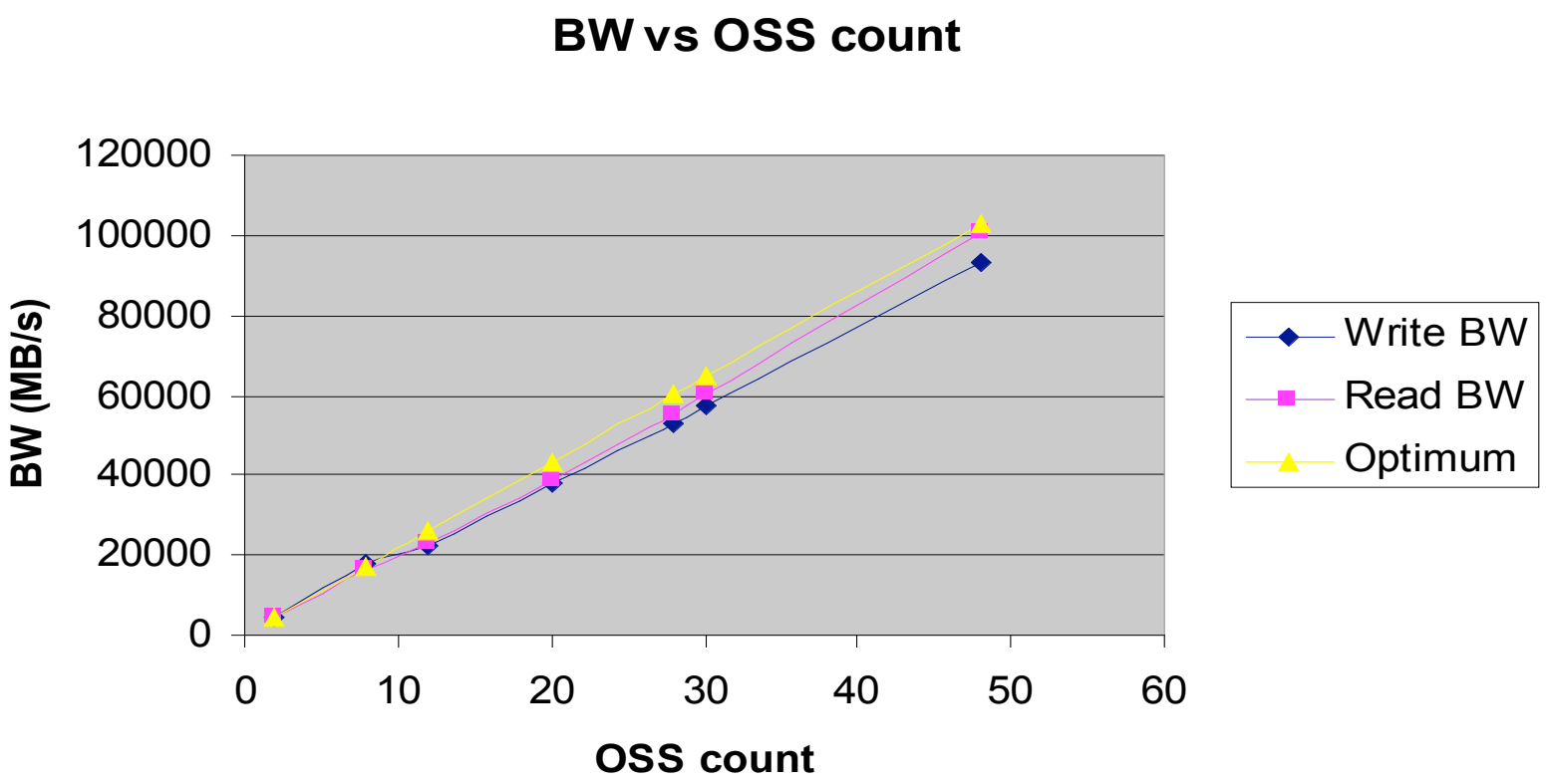
- **HPC Cluster File System**
  - Lustre 1.4.7 and 1.4.8 with vendor support (Bull)
  - Quadrics Elan4 or InfiniBand interconnect
    - Native Quadrics or OpenFabrics LND (qsnet/o2ib)
    - Network is dedicated to a cluster
  - Dual attached DDN 9550 with fibre channel disks (WT, 8+1+1)
    - 16 LUN of 1 TB
    - 2.3 GB/s per DDN couplet
    - OSS are 16 or 8-cores Itanium servers
  - Performance oriented
    - 100 GB/s on checkpoint/restart like benchmark
    - Single client performance: 2.2 GB/s Write, 1.4 GB/s Read
  - TERA-10 day production is 30 TB

HPC Lustre Performance

- **IO's**
  - The global BW increases linearly with the number of clients and saturates



HPC Lustre Scalability

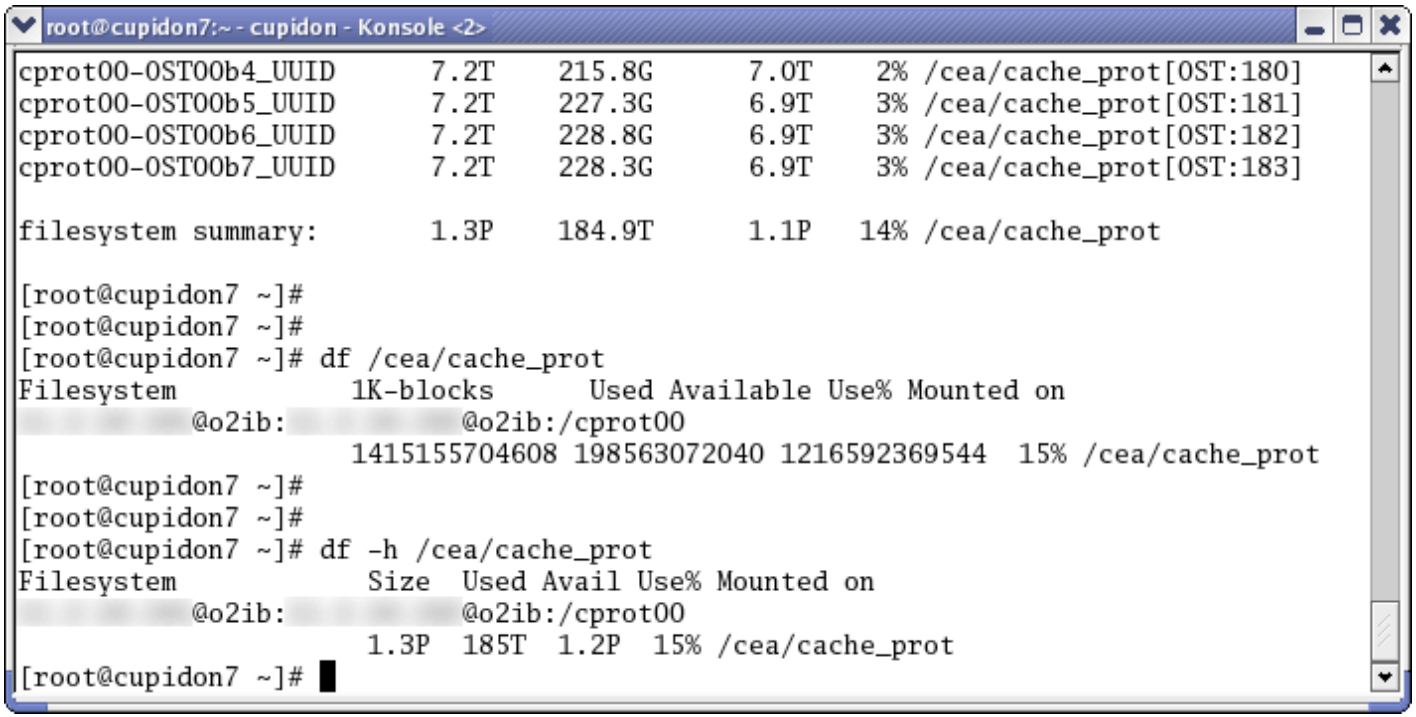


Lustre Usage (II)

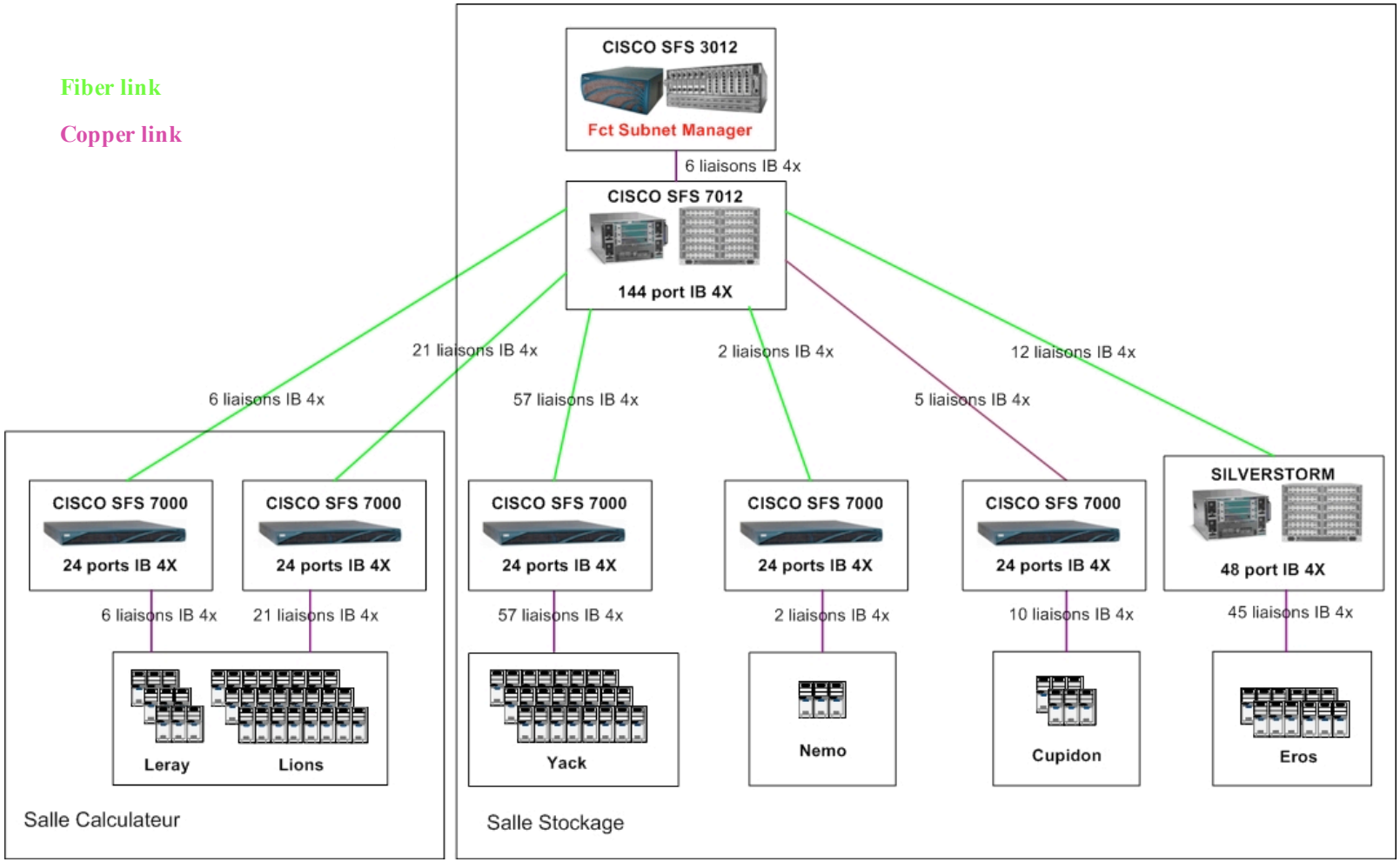
- **Shared File System**
  - Lustre 1.6 Beta 7
  - InfiniBand interconnect
    - Native OpenFabrics LND (o2ib)
    - Network is shared by several clusters
  - Dual attached DDN 9550 with SATA disks (WT, 8+2)
    - 48 LUN of 8 TB
    - 1.5 GB/s per DDN couplet
    - OSS are 4-cores Xeon servers
  - 1 FS shared by several clusters
  - Capacity oriented
    - Multi Petabytes FS (target is 2+ PB)
    - Single client performance: 472 MB/s Write, 371 MB/s Read

Petabytes Capacity File System

- **Initially created with 2 DDN couplets = 2 * 330 TB**
- **Few weeks ago extended in half a day with 2 DDN**
  - No need to reformat
  - Now at 1.3 PB in one FS

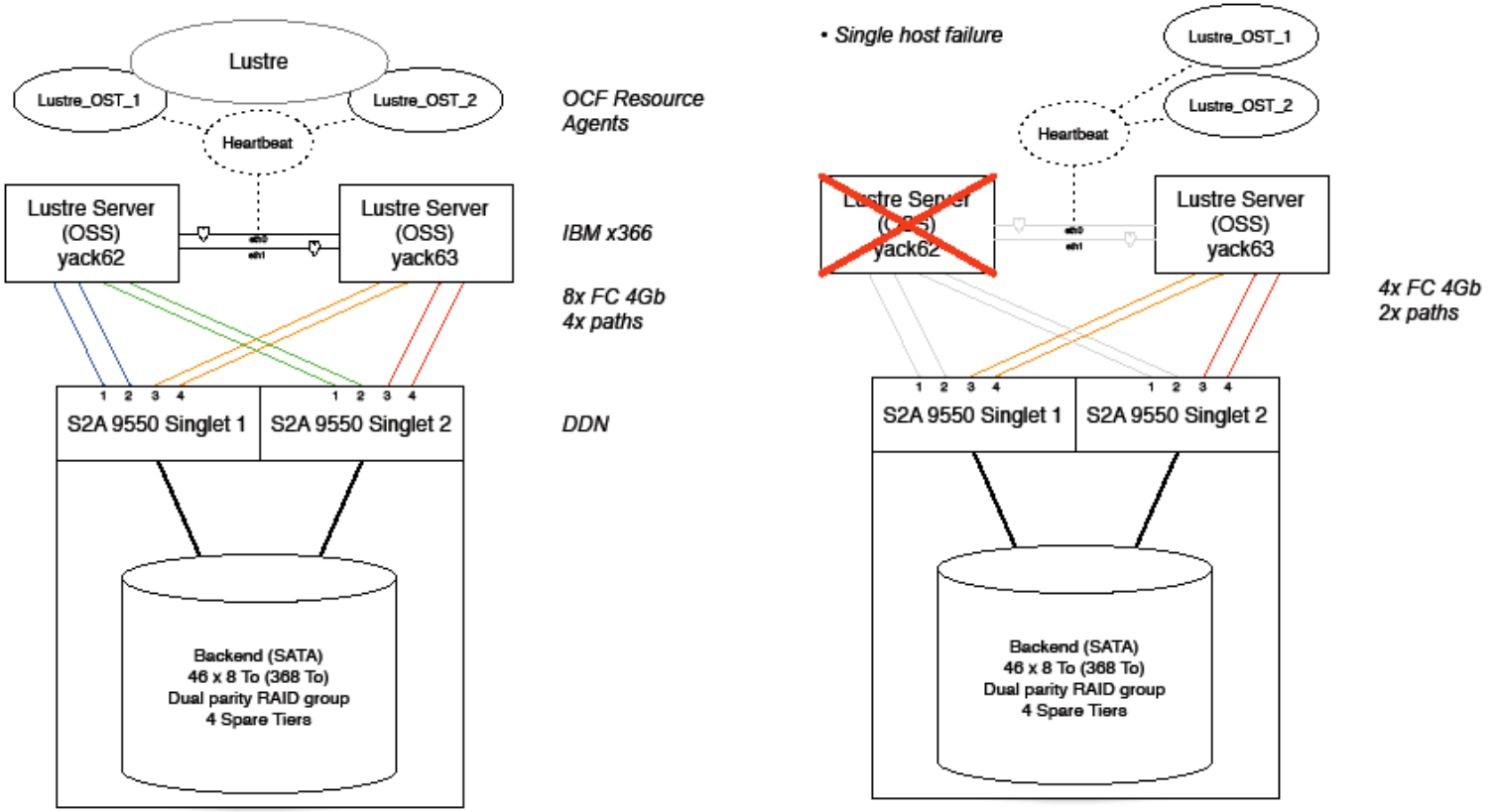


Shared InfiniBand Network Topology



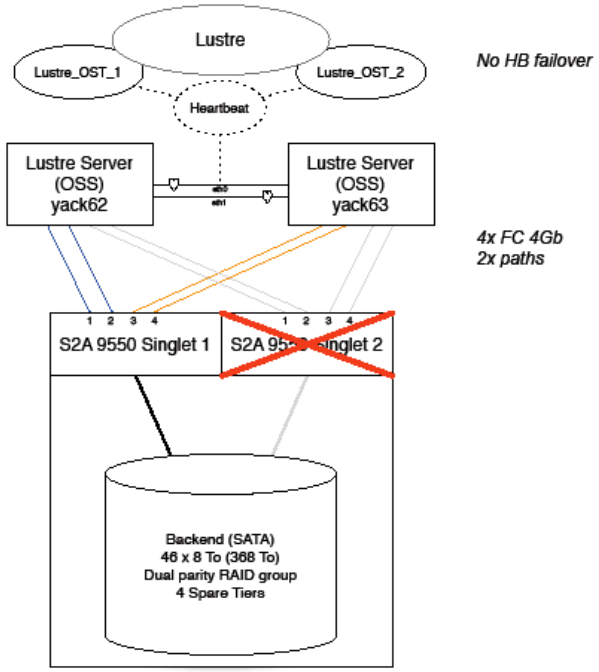
Shared FS failover setup (I)

- **1. Linux-HA + Lustre 1.6 failover setup**



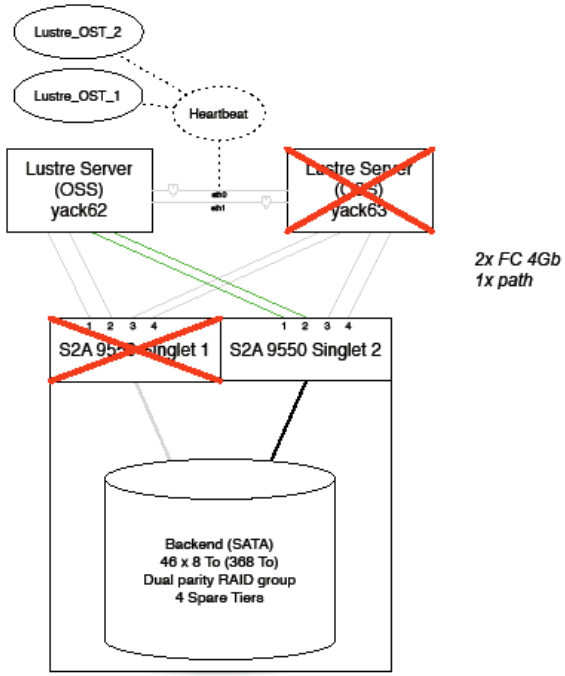
Shared FS failover setup (II)

- **2. Device-mapper multipath I/O failover (Linux DM MPIO)**
  - EMC path prioritizer on the MGS/MDS nodes
  - **clariion** path checker for EMC
  - Home made DDN path prioritizer for OSS nodes
  - **tur** path checker for DDN



No HB failover

4x FC 4Gb
2x paths



2x FC 4Gb
1x path

Lustre HSM Project Update

CFS / CEA Lustre HSM project

- **Goal: Add ILM features in Lustre (target 2008)**
  - ILM stands for Information Lifecycle Management
  - HSM stands for Hierarchical Storage Management
  - Migration inside Lustre or between Lustre and an external backend storage
- **Designed to allow interoperability with multiple external backends, including HPSS**
  - Use of userspace commands to access external backend system
- **All files are always visible in the file system, but file data can reside:**
  - On the primary storage (Lustre file system)
  - On the backend storage (internal or external storage system)
  - On both
- **Metadata (size, …) are always up-to-date**

Inside Lustre HSM

- **Involve the migration of file system objects**
  - Migration enables multiple Lustre features (HSM, caches for Lustre proxy services, space rebalancing, LAID rebuild, etc.)

- **Working at a FID granularity level**
  - MDT FID (full file)
  - OST FID (file object)
  - File access by FID feature
    - ☛ FID is used as the reference key in the backend storage system
    - ☛ Lustre namespace is independent from backend namespace

Inside Lustre HSM (cont'd)

- **File system space management will be either:**
  - Automatic
    - at OST level
    - at FS level (MDT)
  - On-demand
    - eg. based on a provided list of files to purge (e2scan)

- **Purge method**
  - Keep start/end of FID data using the punch() call
  - At OST level (ext3/ldiskfs)
  - At FS level (lustre)

Questions ?