LLNL Lustre Centre of Excellence

Mark Gary

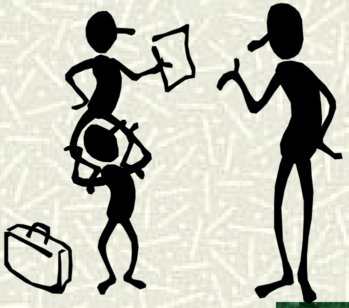
4/23/07

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

UCRL-PRES-230018

LLNL is home to a Lustre Centre of Excellence (LCE)

- We enjoy a close working partnership with CFS
- The Lustre Centre of Excellence (LCE) is written into our ongoing CFS support contract.
- I consider almost everything we do with Lustre, contractual or not, to be an LCE effort item.
- LCE activities at LLNL are many…

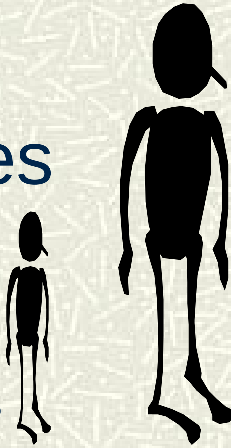


LLNL LCE Effort Areas

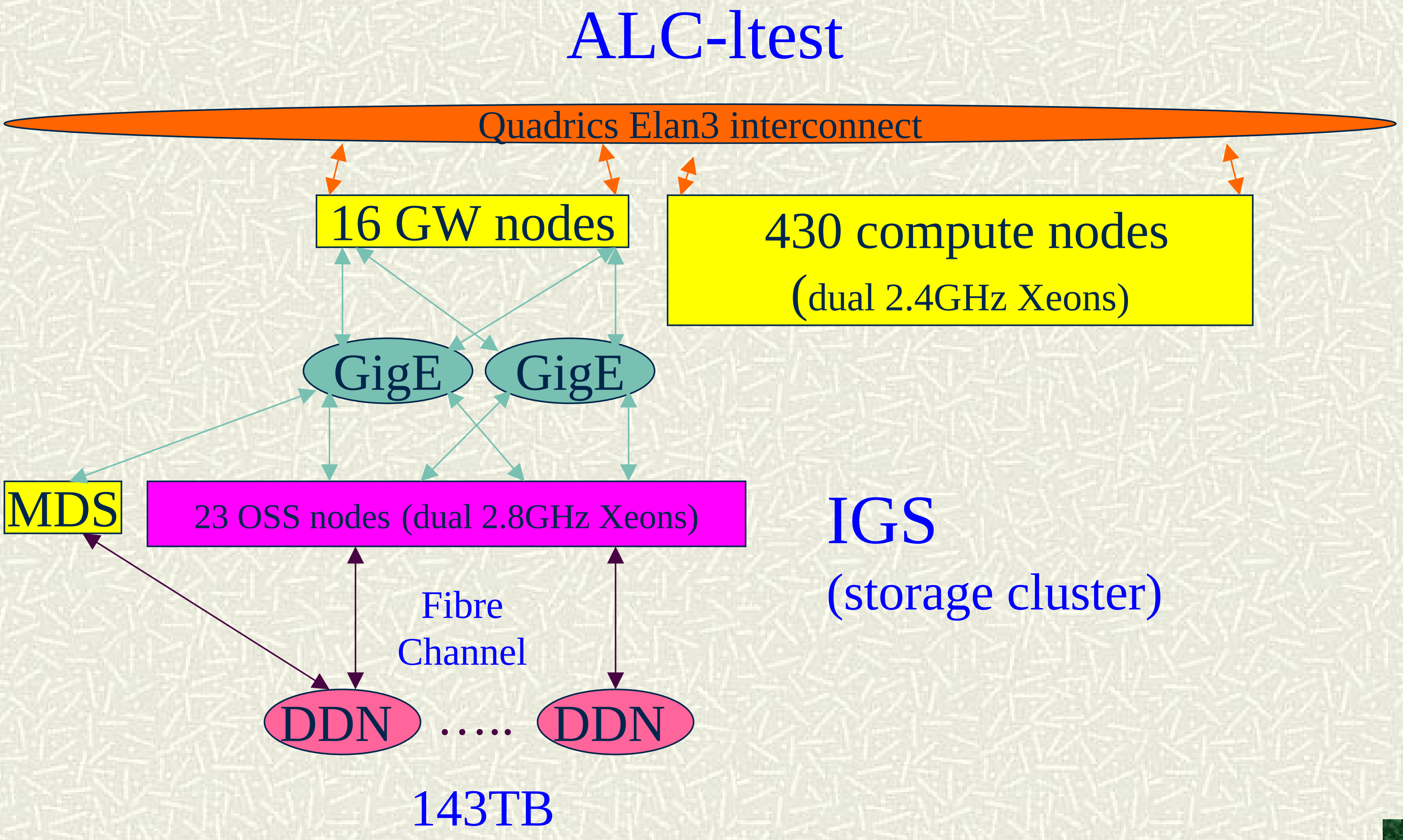
- Selected CFS/LLNL efforts
  - At-scale testing, bug fixing, performance issue analysis
  - fsck:
    - Debugging/fixing
    - Acceleration
  - Metadata speed up
  - Adaptive timeouts
  - Lustre free space management
- LLNL development efforts
  - ZFS prototype
  - Failover implementation
  - Lustre Monitoring Tool 2 (LMT2)
- Tri-Lab PathForward efforts

At-scale testing, bug fixing and analysis

- We operate a very large test environment for use by ourselves and CFS.
  - We run around-the-clock at-scale testing of all of our releases
  - Scheduled dedicated testing by CFS benefits the entire community
- As in other areas, our scale regularly reveals bugs and performance issues that don’t show up in small-scale tests:
  - We are constantly working with CFS on issues revealed at-scale
  - LLNL’s top-10 bugs prioritized each week
  - Weekly meeting with CFS to review progress and plans



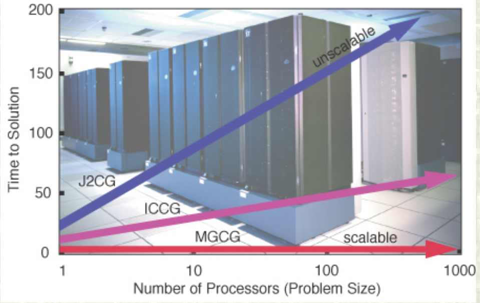
At-scale Lustre test resource



Fsck improvements

- Improvements include
  - Fixing segfault due to corrupt extent headers
  - Fixing segfault on extended attribute corruption
  - Improving e2fsck heuristics for detecting corrupted inodes
  - Shared block resolution - implement alternative to cloning
  - Coverity-detected bugs, fixes
  - …
- Speed-up milestone
  - Halve the time for fscks
  - Based on looking at only active inodes (keeping track of inode allocation high-water mark).

Metadata speedup



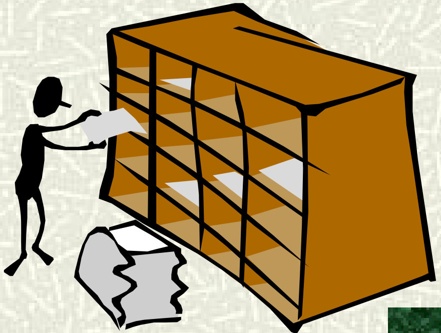
- Goal is to:
  - Cut ls –l time by 50%
  - Cut rm –r time by 75%
  - Improve performance (LRU create test) by 70%
- Achieved by client-side read ahead for MDS (for directory contents and parallel fetching of attributes)
- Dynamic sizing and automatic tuning (client-based lock timeout) of the client LRU (lock) list

Adaptive timeouts

- Static timeouts used by callers of Lustre RPCs cause difficulties in unusual-load scenarios
- CFS is modifying calls to RPCs and other Lustre components to dynamically respond to RPC delays
- Make all Lustre timeouts sensitive to recent completion times, and feedback.

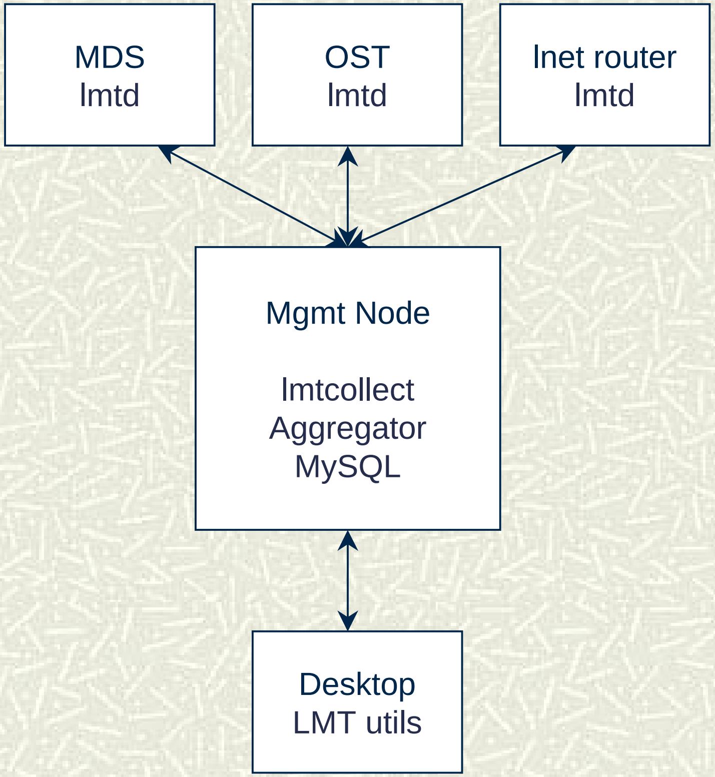
Free space management

- Automate and enhance Lustre free space management:
  - Detect full OSTs and adapt
  - Automatic space-balancing and migration
  - Administrator-initiated space balancing
  - Administrator-initiated full migration of OSTs
  - Administrator-initiated on-line defragmentation of OSTs



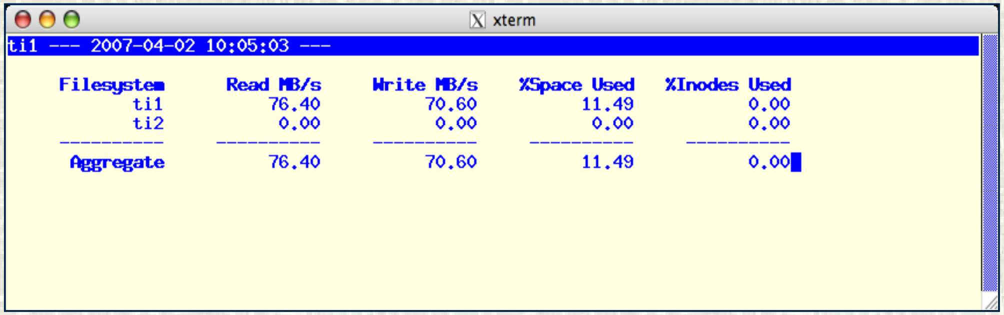
Lustre Monitoring Tools v2 – LMT2

- The 2nd generation of Lustre Monitoring Tools (LMT) uses a MySQL database backend for storing and retrieving Lustre information related to OSTs, the metadata servers, and the routers. As a result, LMT applications can analyze Lustre performance either in real-time or over specified historical periods.
- There are currently three LMT2 apps in development:
  - **lstat**: simple text display that operates like Unix “netstat” (v1.0 complete)
  - **ltop**: curses-based tool that operates like Unix “top” (v1.0 complete)
  - **jwatch** (working title) : new GUI with extensive charting capabilities (v1.0 beta)

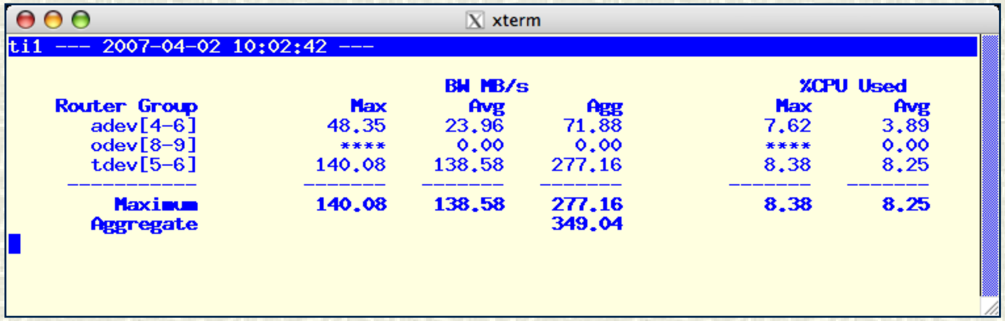


LMT2 “top” – ltop

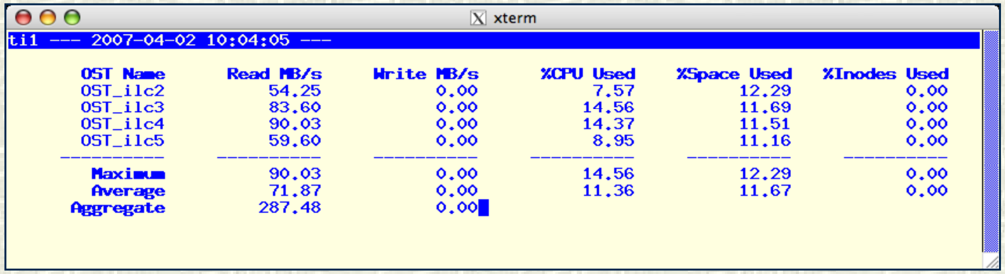
- Multiple “views” – router, router group, filesystem, OST, OSS, MDS, …
- Low overhead
- Curses-based



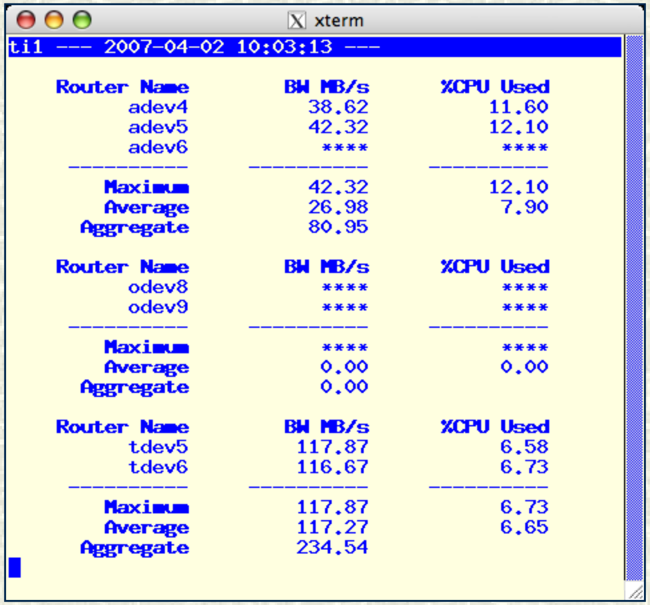
```
ti1 --- 2007-04-02 10:05:03 ---

     Filesystem    Read MB/s    Write MB/s   %Space Used  %Inodes Used
            ti1        76.40         70.60         11.49          0.00
            ti2         0.00          0.00          0.00          0.00
     ----------   ----------    ----------    ----------    ----------
      Aggregate        76.40         70.60         11.49          0.00
```



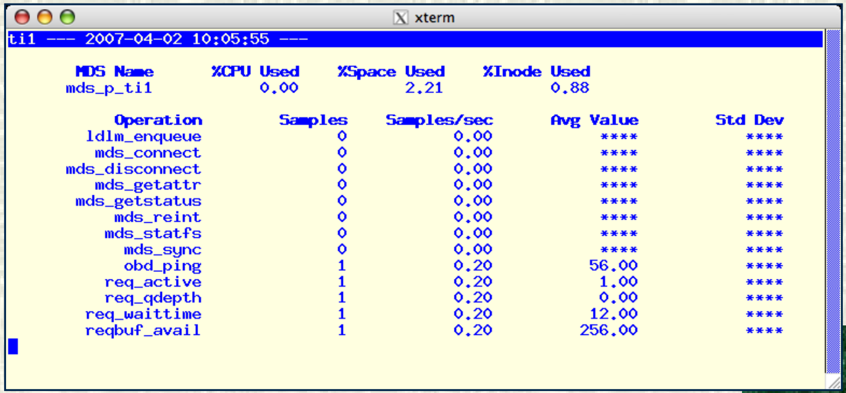
```
ti1 --- 2007-04-02 10:02:42 ---

                                      BW MB/s                        %CPU Used
    Router Group             Max       Avg       Agg             Max       Avg
       adev[4-6]           48.35     23.96     71.88            7.62      3.89
       odev[8-9]            ****      0.00      0.00            ****      0.00
       tdev[5-6]          140.08    138.58    277.16            8.38      8.25
    ------------        --------  --------  --------        --------  --------
         Maximum          140.08    138.58    277.16            8.38      8.25
       Aggregate                              349.04
```



```
ti1 --- 2007-04-02 10:04:05 ---

       OST Name    Read MB/s    Write MB/s     %CPU Used   %Space Used  %Inodes Used
       OST_ilc2        54.25          0.00          7.57         12.29          0.00
       OST_ilc3        83.60          0.00         14.56         11.69          0.00
       OST_ilc4        90.03          0.00         14.37         11.51          0.00
       OST_ilc5        59.60          0.00          8.95         11.16          0.00
     ----------   ----------    ----------    ----------    ----------    ----------
        Maximum        90.03          0.00         14.56         12.29          0.00
        Average        71.87          0.00         11.36         11.67          0.00
      Aggregate       287.48          0.00
```



```
ti1 --- 2007-04-02 10:03:13 ---

    Router Name      BW MB/s    %CPU Used
          adev4        38.62        11.60
          adev5        42.32        12.10
          adev6         ****         ****
     ----------   ----------   ----------
        Maximum        42.32        12.10
        Average        26.98         7.90
      Aggregate        80.95

    Router Name      BW MB/s    %CPU Used
          odev8         ****         ****
          odev9         ****         ****
     ----------   ----------   ----------
        Maximum         ****         ****
        Average         0.00         0.00
      Aggregate         0.00

    Router Name      BW MB/s    %CPU Used
          tdev5       117.87         6.58
          tdev6       116.67         6.73
     ----------   ----------   ----------
        Maximum       117.87         6.73
        Average       117.27         6.65
      Aggregate       234.54
```

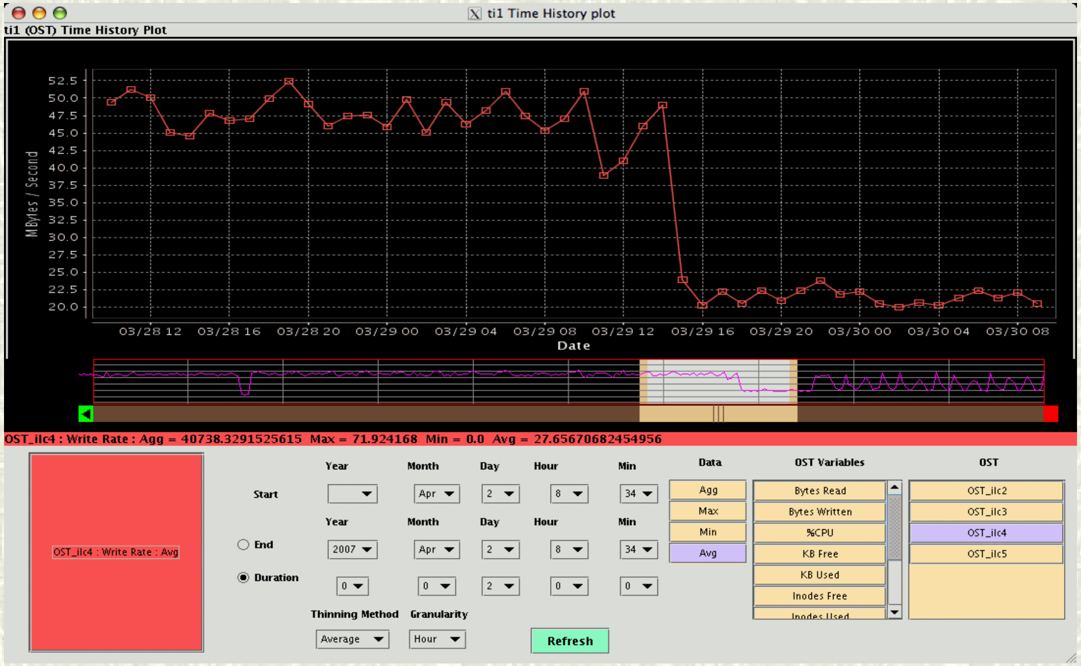


```
ti1 --- 2007-04-02 10:05:55 ---

        MDS Name     %CPU Used    %Space Used    %Inode Used
       mds_p_ti1          0.00           2.21           0.88

         Operation        Samples    Samples/sec       Avg Value           Std Dev
      ldlm_enqueue              0           0.00            ****              ****
       mds_connect              0           0.00            ****              ****
    mds_disconnect              0           0.00            ****              ****
       mds_getattr              0           0.00            ****              ****
     mds_getstatus              0           0.00            ****              ****
         mds_reint              0           0.00            ****              ****
        mds_statfs              0           0.00            ****              ****
          mds_sync              0           0.00            ****              ****
          obd_ping              1           0.20           56.00              ****
        req_active              1           0.20            1.00              ****
        req_qdepth              1           0.20            0.00              ****
      req_waittime              1           0.20           12.00              ****
      reqbuf_avail              1           0.20          256.00              ****
```

The LMT2 GUI

Start with xwatch-lustre functionality, then add:

- New views (OSS, Filesystem, Router Group, …)
- Plotting capability (historical trends, heart-beat, …)
- Customization features
- Full-system health “at a glance”
- Client display



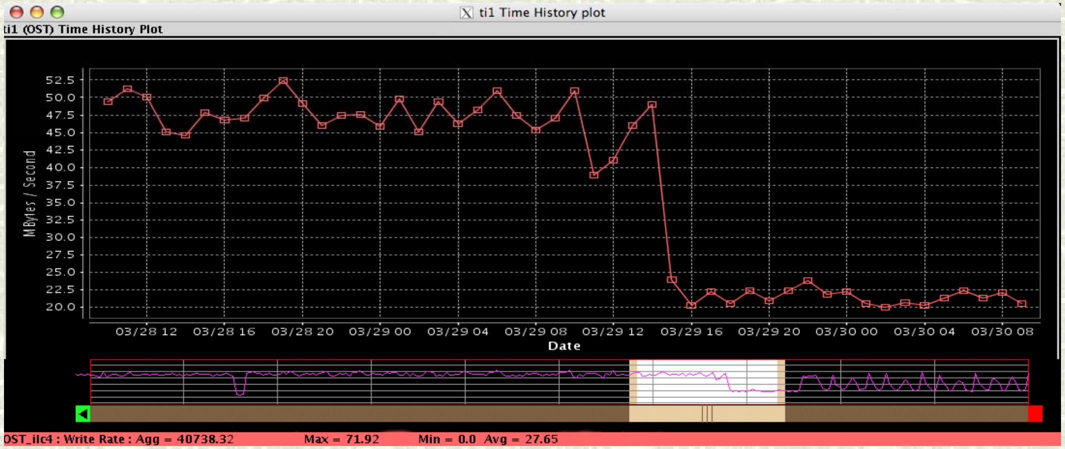
OST_ilc4 : Write Rate : Agg = 40738.3291525615 Max = 71.924168 Min = 0.0 Avg = 27.65670682454956

OST_ilc4 : Write Rate : Avg

	Year	Month	Day	Hour	Min
Start		Apr	2	8	34
○ End	2007	Apr	2	8	34
● Duration	0	0	2	0	0

Thinning Method: Average
Granularity: Hour

New graphical chart control in development.



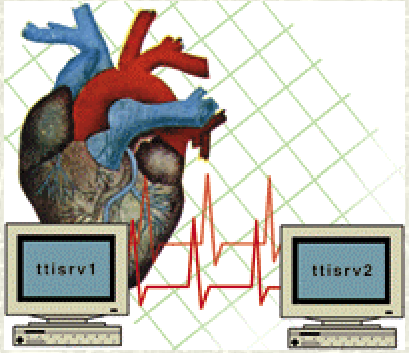
OST_ilc4 : Write Rate : Agg = 40738.32 Max = 71.92 Min = 0.0 Avg = 27.65

LMT2 Plans

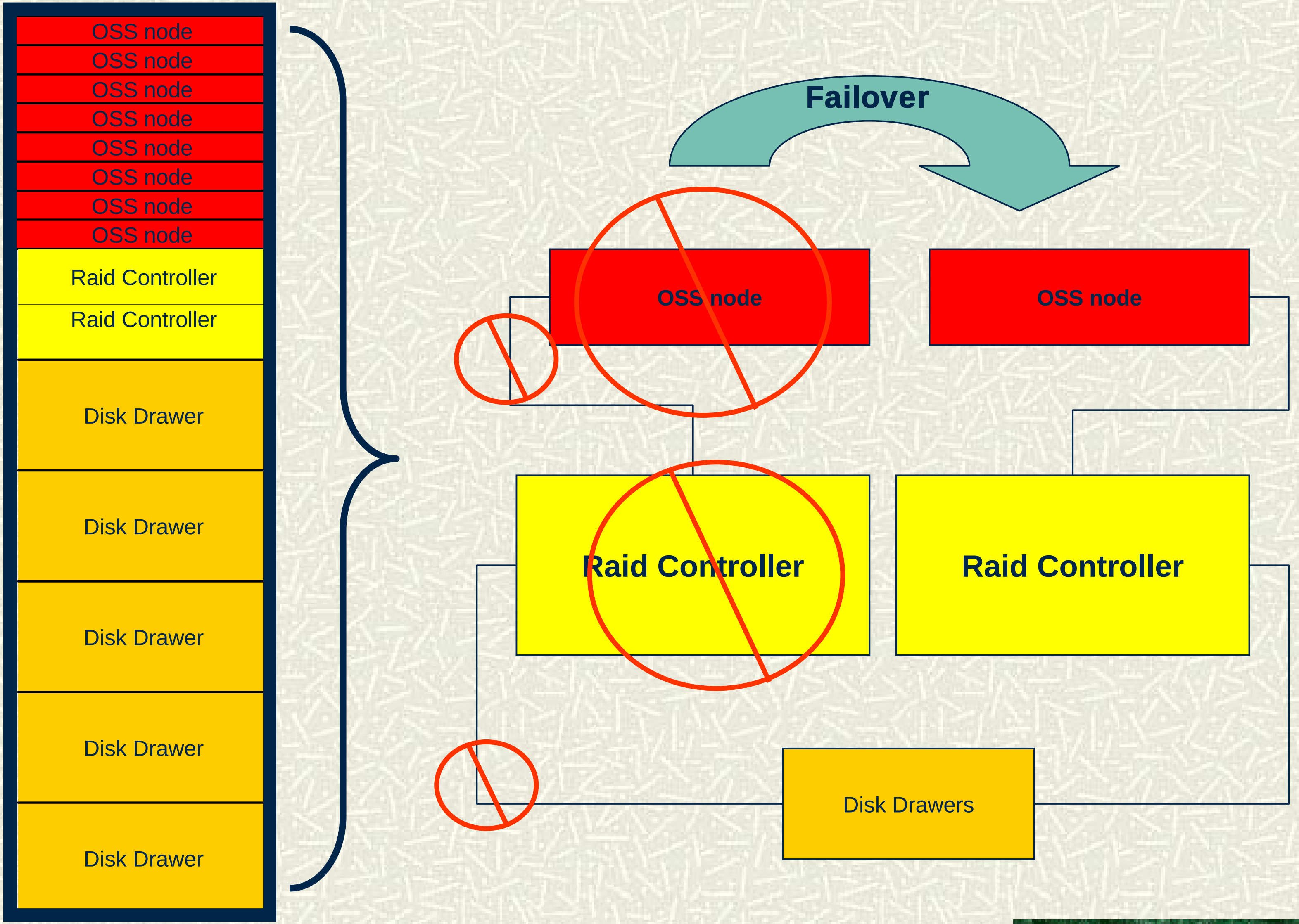
- LMT 2.0 release [internal]
- LMT 2.0 release [external]
- Extend database access class
- Add more views to GUI and ltop
- Extend new GUI to support historical and trending plots.
- Release version LMT 2.1
- Collect OSS-specific data
- Add views for OSS-specific data in LMT utilities
- Extend new xwatch-lustre to include a global health view of Lustre
- Release version LMT 2.2
- Add support for viewing client data
- Release version LMT 2.3

Failover implementation

- Linux-ha based
- Initial implementation currently undergoing test
- Priority on fencing and prevention of data loss requirements
- Based upon Release 2 of Linux-HA software (active development, testing, fixing)



Failover



OSS node
OSS node
OSS node
OSS node
OSS node
OSS node
OSS node
OSS node
Raid Controller
Raid Controller
Disk Drawer
Disk Drawer
Disk Drawer
Disk Drawer
Disk Drawer

Failover

OSS node

OSS node

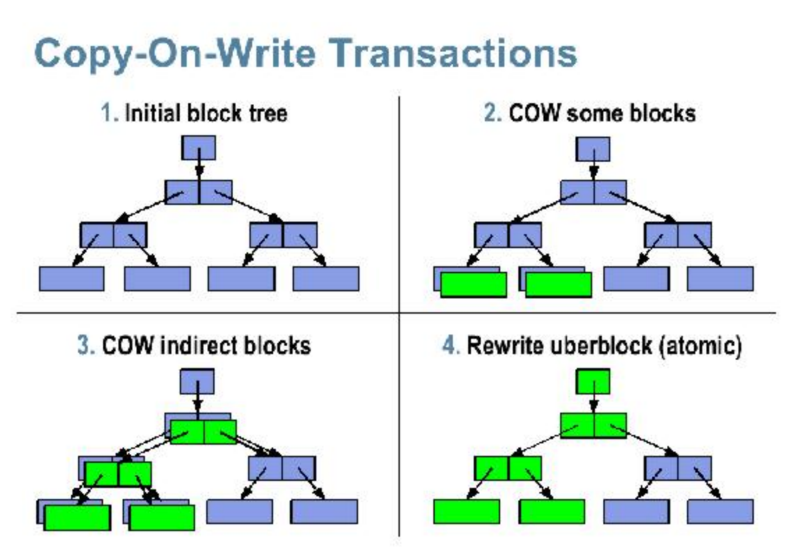
Raid Controller

Raid Controller

Disk Drawers

ZFS prototype

- LLNL is launching a prototyping effort to investigate the viability of running OSTs atop Sun's ZFS file system.
- Our prototyping effort only goes as far as porting a portion of ZFS into the Linux kernel
- Our goal is to learn the viability of the partial port and let the results guide any future work



Lustre/ZFS motivation

EXT3 Problems

- Max OST FS Size of 16-32TB
- Offline fsck recovery time
- Data corruption goes unnoticed
- Crashes, corruption, fsck challenges and complexity

ZFS

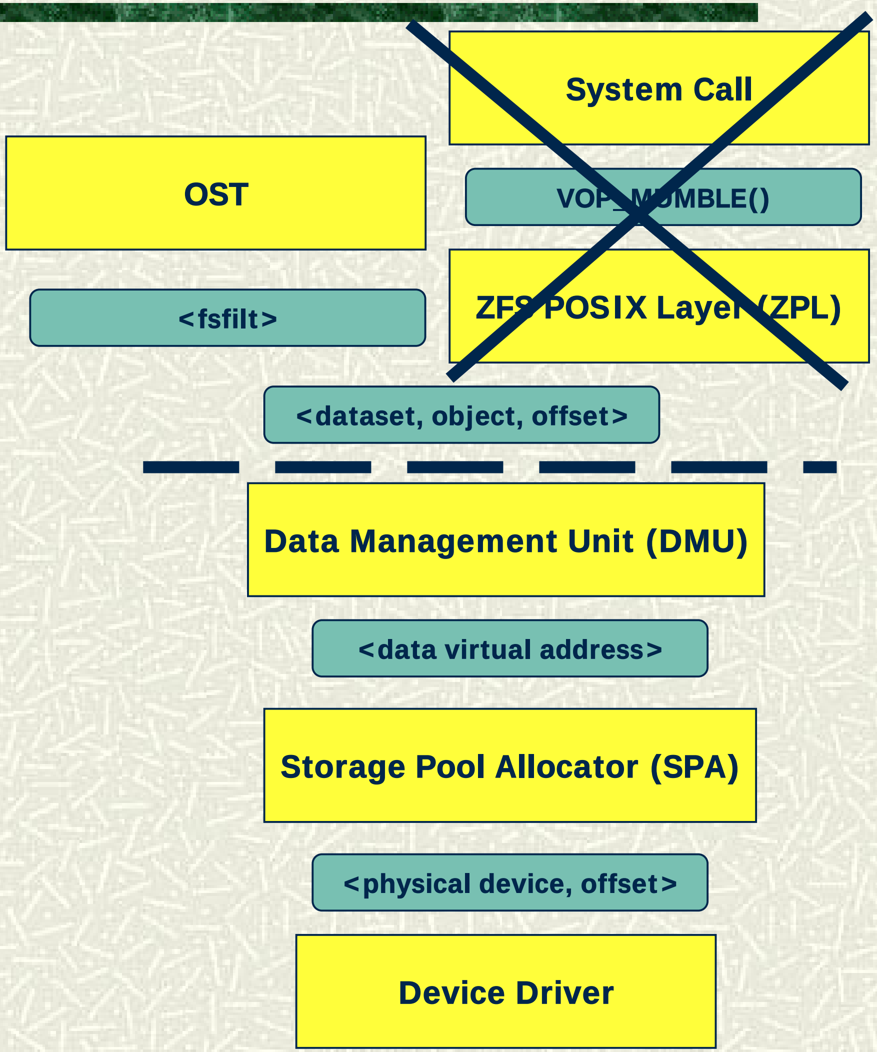
- Max OST FS size unlimited by file system
- Consistency checking is online
- Every block is checksummed (metadata and data)

Other ZFS benefits

- Copy-on-write may result in more streaming I/O
- More redundancy options (RAIDZ2, metadata "ditto blocks",…)
- Administrative flexibility
- JBOD & other hdwr options

Lustre/ZFS Integration Strategy

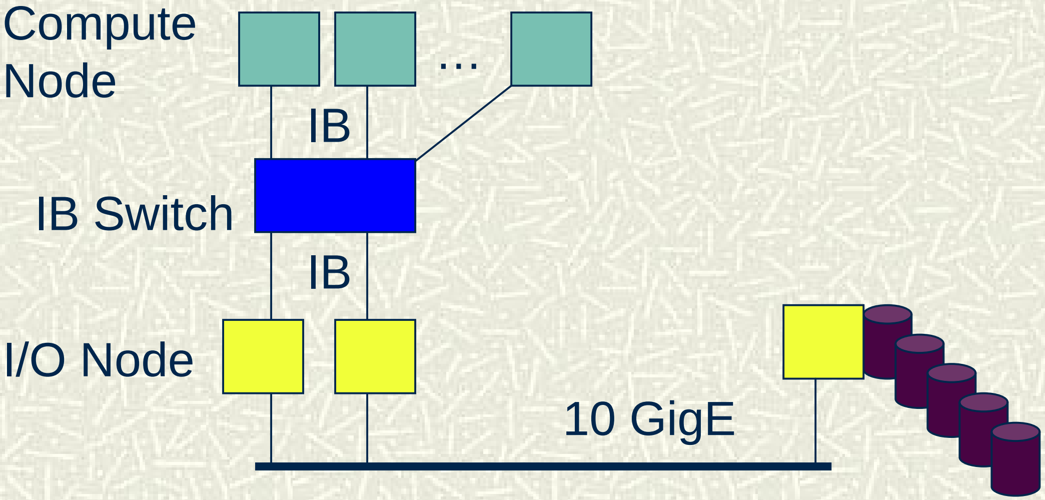
- Replace EXT3 on OSTs with ZFS
- Port ZFS Data Management Unit (DMU) and Storage Pool Allocator (SPA) only
- Requires fsfilt to DMU integration



System Call
OST
VOP_MUMBLE()
<fsfilt>
ZFS POSIX Layer (ZPL)
<dataset, object, offset>
Data Management Unit (DMU)
<data virtual address>
Storage Pool Allocator (SPA)
<physical device, offset>
Device Driver

Tri-Lab PathForward Efforts

- Tri-Lab (LANL, SNL, LLNL)/HP/CFS efforts
  - Infiniband
    - Compute nodes speak only IB
    - I/O nodes translate to IP for 10GigE
    - Lustre storage exists on 10GigE LAN
  - Clustered MDS
  - Security



Conclusion

- The LLNL/CFS relationship is active and varied:
  - At-scale testing, bug fixing, performance issues
  - fsck improvements
  - Metadata speed up
  - Adaptive timeouts
  - Lustre free space management
- LLNL is pursuing a number of development efforts
  - ZFS prototype
  - Lustre Monitoring Tool 2 (LMT2)
  - Failover implementation
- Tri-Labs, HP and CFS are working other areas

The LCE is working and benefiting the entire Lustre community