



# Key Issues deploying Lustre® at HP Customer sites

## Lustre User Group



April 2006

Fergal Mc Carthy  
HP SFS Senior Software Engineer  
High Performance Computing Division

Lustre, CFS, and Cluster File Systems are trademarks of Cluster File Systems, Inc in the United States.

© 2004 Hewlett-Packard Development Company, L.P.  
The information contained herein is subject to change without notice



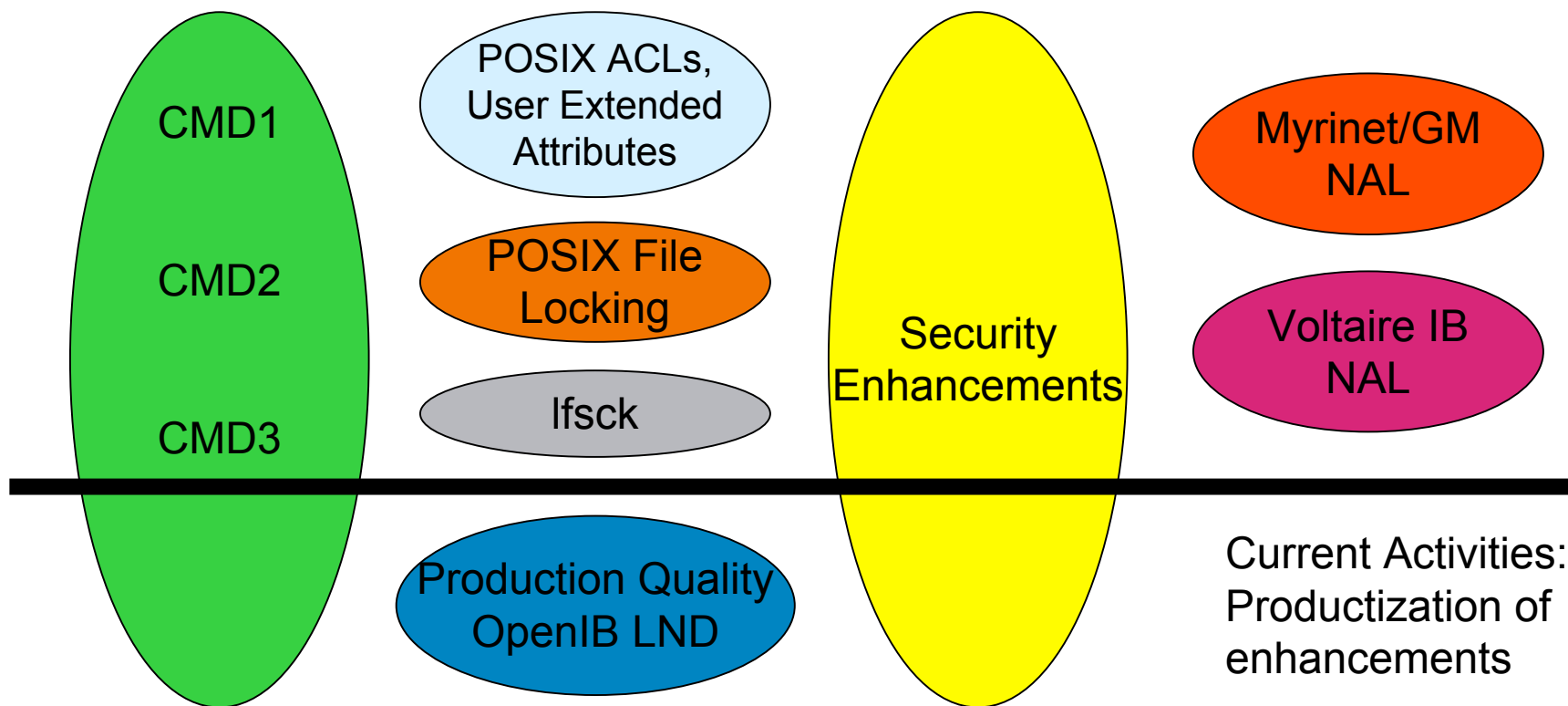
# Topics

- HP's Involvement with Lustre
- What is HP Scalable File Share?
- Deploying HP SFS/Lustre
- Lessons Learned
- Looking to the Future

# Hendrix

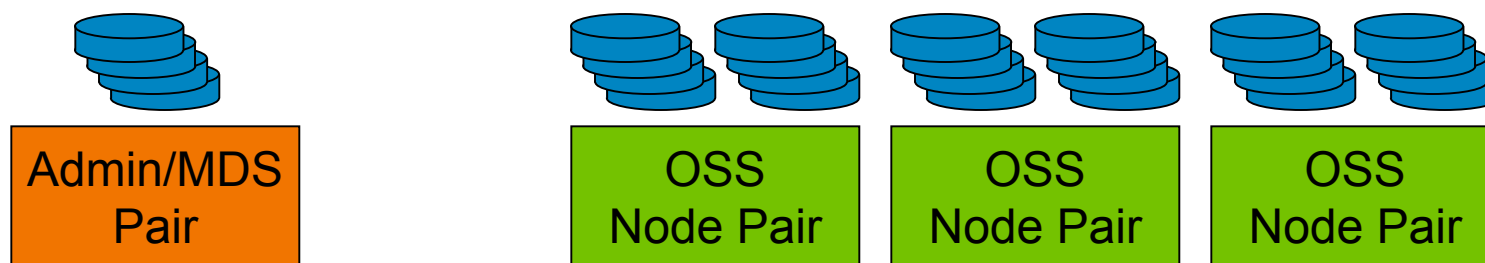
- Advanced Development Program
- Collaboration between CFS, Tri-Labs, HP and Intel
- Focused on accelerated developing of Lustre functionality

Previous Activities: Development/Testing of Lustre Enhancements



# HP Scalable File Share (SFS)

- High Availability Lustre Server Appliance
  - Install Admin/MDS node pair only
  - Integrated Centralised Management tool
  - Simplified interfaces to low level Lustre activities
  - Health Check facility



- Management services
  - Configuration DB
  - LDAP DB
  - HTTPD
  - PXE/tftp boot services
- Multiple FS MDS devices
- Serve OSTs in minimal setups
- OSS node pairs serve OSTs
  - Network boot from Admin/MDS pair
  - Up to 8 2TB OST devices per pair
  - Up to 32 pairs /64 OSS nodes
  - Up to 512TB capacity per SFS
- Add extra OSS pairs to extend existing capacity

# HP Scalable File Share (SFS) (cntd.)



- High Availability within node pairs
  - Automatic failover on:
    - Server node death/crash/LBUG
    - Storage or interconnect connectivity loss
  - Automatic failback when failed nodes restart
- Interconnects
  - Choice of one high performance interconnect (Elan4, Voltaire IB, Myrinet/GM)
    - Optional simultaneous GigE/TCP
  - Or Dual-GigE/TCP

# HP Scalable File Share (SFS) (cntd.)



- Easy to update:
  - Stop file systems and shutdown OSS nodes
  - Re-install Admin/MDS pair with updated kit
    - Automatically detects and uses existing configuration
  - Boot OSS nodes and start file serving again
- Monitoring
  - Configurable automated alert mechanism
  - NAGIOS integration
  - Performance data collection (collectl) and visualisation via HTTPD



# Deploying HP SFS/Lustre

- HP SFS is deployed at many customer sites including a number of systems with approx 1K nodes
- Along the way we have encountered and overcome some interesting issues:
  - Failover/Failback Robustness
  - Client only build
  - Iconf DB (LDAP & XML) Optimisation
  - LDLM Robustness
  - Zeroconfig
  - Performance



# Failover/Failback Robustness

## Software Failover Cleanup of Lustre Devices

- Initial work with CFS 3 years ago
  - Provide software teardown/failback mechanism
  - Permit rebalance of OST serving after failed OSS restart
- Stability problems encountered in Lustre 1.2 stream
  - MDS and OSS node crashes
- Worked closely with CFS to address in Lustre 1.4 stream
  - Needed for HA automatic failover/failback integration
  - Client operations should not fail
    - May take longer to complete





# Failover/Failback Robustness (cntd.)

## High Availability and Client Recovery

- Originally FAILED\_IMPORT upcall used
  - Userspace could query config DB for updated server
  - Lead to significant LDAP DB load at large scale
- Failover friendly DEFAULT recovery added in Lustre 1.4.2
  - Round robins connecting to servers until device is found again

## Safer /proc access

- Accessing client or server /proc areas during device creation or teardown could cause LBUGs/crashes
- HP Patches provided to CFS to reduce instability
  - Facilitates more versatile automated monitoring



# Client Only Build

- Grew from HP desire to support 2.4 Lustre clients without need to modify Ext3
  - Clients only need subset of Lustre Server functionality
  - Customers don't like having to patch client kernels to enable Lustre
  - Also don't like having to apply more patches than absolutely necessary
- CFS improved upon HP's prototype mechanism
  - Server only Lustre build also possible
- Significant effort still required to maintain patches for supported client kernels

# Iconf DB (LDAP & XML) Optimisations



## HP SFS relies heavily on LDAP

- Optimize Iconf DB access mechanisms
  - Cache query results
    - Noticed some queries repeated many times
    - Accelerates LDAP & XML access
  - Pull complete LDAP DB snapshot and query locally
    - Minimise network traffic/LDAP server load
  - Cache LDAP DB snapshot on-disk for client mounts
    - Remove need for unmount to access LDAP DB
      - Uses python object pickling



# LDLM Robustness

## Scalability and Robustness Improvements

- Large client populations (256 – 1K clients) showed LDLM scaling issues
  - Frequently due to stale client references
    - Clients that crashed, or were rebooted/reset without unmount
  - Ping Evictor greatly helped in this:
    - Improved failover/recovery restart times
    - Improved lock recovery times after multi-node failure
  - Upcoming Nested Lock Elimination for I/O code paths
    - Failover of unrelated OSTs could cause client evictions
    - Should help reduce lock contention for multi-stripe N→1 writers

# LDLM Robustness (cntd.)

## Deadlocks and Client Evictions

- Lead to unexplained I/O errors or stalls
- HP has been working very closely with CFS to
  - Identify reproducers for problem scenarios
  - Develop and test fixes
- Issues being addressed
  - Deadlocks/evictions due to MDS and OST device failover
  - Deadlocks/evictions due to racing general operations
    - LTP racer test - <http://kbugs.org/racer.tar.gz>

# Zeroconfig

- Failover MDS specification
  - Lustre 1.4.6 can specify multiple MDS LNET nids
    - Round robins through list until it config found
- Re-mount optimisation
  - Re-mount showed exponentially increasing slowdown
    - Unless obdclass unloaded
  - Problem identified with UUID management
  - Working with CFS to develop patch



# Performance

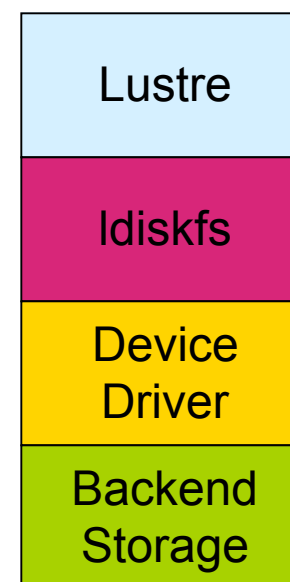
## HP Enhancements for 2.6 Kernels

- Functionality that we had been using in 2.4 kernels
- General
  - TCP Zero Copy Support
- Lustre Server
  - Extent only (non-mballoc) operation
    - Ported 2.4 style extent block allocator
  - Async Journaling Support
- Lustre Client
  - Direct I/O Support

# Lessons Learned

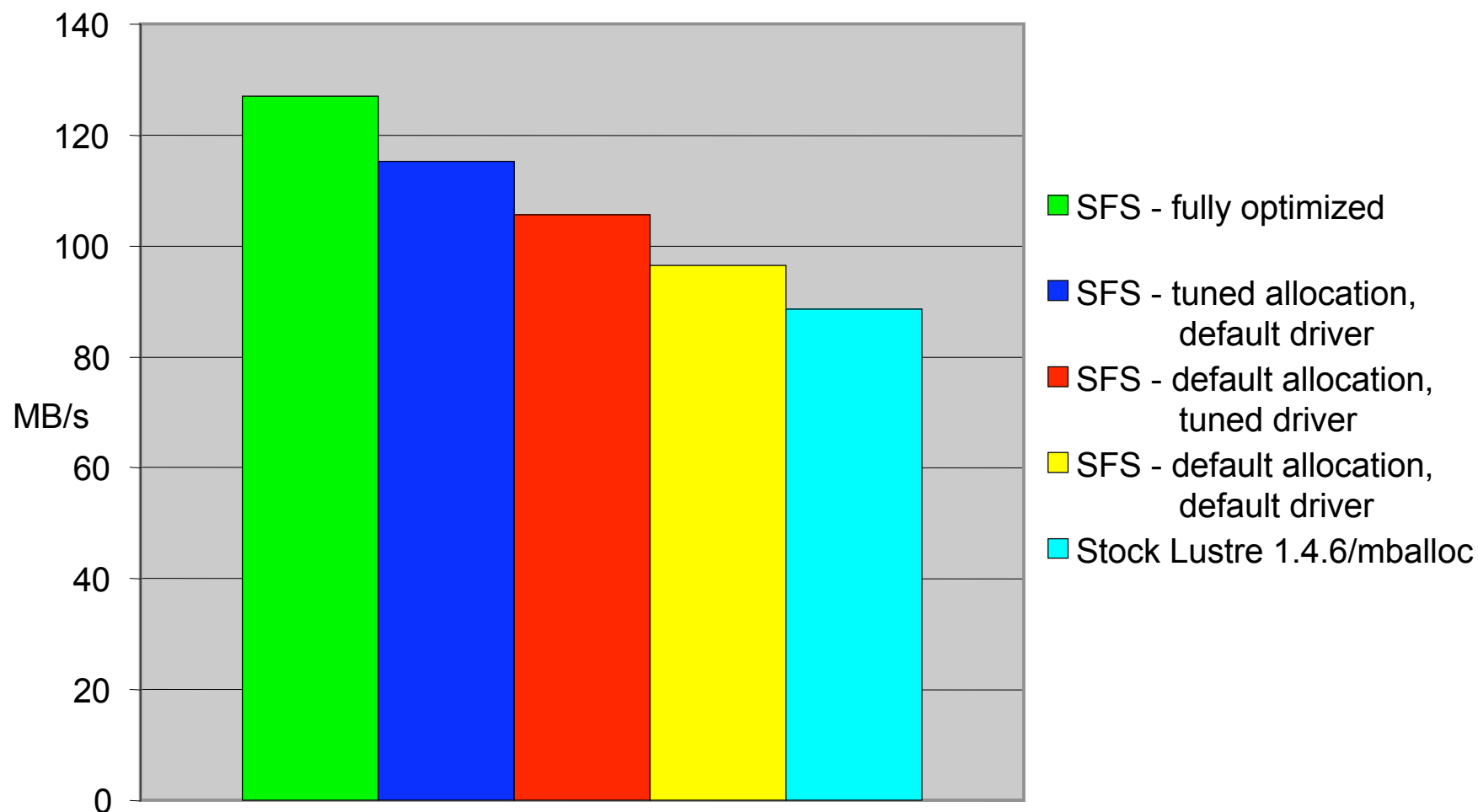
## Getting the best I/O rates

- Need to optimise the following
  - Delivering data to the server
  - Allocating blocks to write data to
  - Device driver transfers
  - Configuring backend storage
- Taking into account storage device characteristics





# Lessons Learned (cntd.)



# Looking to the Future

- Patchless Client Support
- 2.6 NFS Re-export
  - Multiple NFS Re-export
- Version interoperability
- Performance Optimisations
  - N → 1 and Small file I/O
  - Meta-data, e.g. ls -l and rm
- Security



## For further information

- HP Scalable File Share Product Information
  - <http://www.hp.com/techservers/products/sfs.html>
- HP Collaboration and Competency Network
  - <http://www.hp.com/techservers/hpccn>



i n v e n t