Managing Lustre on a Budget

Rick Mohr

National Institute for Computational Sciences
University of Tennessee

What to Expect

- Lustre management from the point-of-view of a “small” organization
- “Budget” refers to any limited resource (money, people, time)
- Collection of my general observations, past experiences, and current practices
  - Some advice, some options, and a whole lot of opinions
  - Try to highlight important considerations
  - Disclaimer: Opinions expressed are my own, YMMV, yadda, yadda…

My Background

- National Institute for Computational Sciences (NICS) at the University of Tennessee
  - Best known for the Kraken (Cray XT5) supercomputer
  - Also home of Nautilus, Keeneland, Beacon, and Darter
- Senior HPC systems admin and lead storage admin
  - Worked in HPC for about 15 years (7 years at NICS)
  - Worked with Lustre for 6 years
  - Deployed first site-wide Lustre file system at NICS

Outline

- General Considerations
  - Various cost factors
    - Equipment
    - Staff
    - Support
  - Topics not just applicable to small sites
- Operational Considerations
  - Topics related to deployment, monitoring, testing, etc.
  - Examples from NICS

General Considerations

Costs

- Many costs associated with running Lustre
  - Staff
  - Equipment
  - Support
- People want more for less
  - More capacity, more bandwidth
  - Commodity prices
- ~~Cutting Corners~~ → Smart Allocation
- Question is not “Is it valuable?”, but rather “Is it a value?”

Staff

- "Grow your own" vs. hire
  - Not necessarily easy to find Lustre admins (and experience isn't cheap)
  - Investing in current staff may be good long-term strategy (but plan to spend money on training)
    - Good if you can identify young talent
- Consider scale of file system and timeframe for getting up and running
- Leverage experience from current staff
  - Cluster management
  - Other file system experience

How much staff?

- That depends on…
  - Size of file system
  - Uptime/response requirements
  - Importance of the file system
- Unfortunately, no easy answer
  - Large file system → Dedicated staff
  - Small file system → Multi-purpose staff
- Day-to-day workload can vary significantly
  - Plan for the peaks, not the valleys
- Somebody's primary focus needs to be on Lustre

Lustre Support

- Self-support or vendor support?
- Could influence hardware purchase if vendor can bundle Lustre support
- Self-support:
  - More flexibility in configuration, tuning, etc.
  - Might be good if site has multiple platforms
  - Could be on your own for big problems
    - Ex – LU-5726 (file deletions caused OOM)
- Vendor support
  - More experience to tap into (sometimes)
  - Vendor requirements could be limiting factor
    - Ex – Lustre 1.8 to 2.4 upgrade

Equipment

- Most obvious and concrete cost
- Roll-your-own vs. Vendor product
  - Pros and Cons to each
  - There is somewhat of a spectrum here
- You get what you pay for…
  - Saving money on equipment could translate into more staff costs
- …but make sure you get what you pay for
  - View vendor claims with a skeptical eye
  - Make sure you understand performance numbers
  - Be prepared to verify any vendor claims

Equipment (Cont.)

- Consider existing vendor relationships
  - This could save $$
  - Bundle Lustre hardware with other purchases
- Leverage in-house knowledge to support new hardware
  - Are there any staff with experience using storage from Vendor X?
- Lustre manual provides guidance on hardware specs
  - There is an entire chapter about memory sizing, disk space requirements, RAID suggestions, etc.

Test Systems

- Mainly needed for do-it-yourself sites
- Many people don't like buying hardware that isn't used in production
  - It can look like the hardware is going to waste
- Very important to have a test system for:
  - Testing upgrade procedures
  - Reproducing bugs
  - Testing bug fixes
  - Deploying new features
- Doesn't need to be big, but should be similar technology to production system

Policies

- Policies are used to manage expectation both internally (amongst staff) and externally (with end-users)
- Examples:
  - 9x5 vs. 24x7 support
  - Purging / data retention
  - Backups
  - Quotas
- Policies can have a direct effect on purchasing and staffing decisions (and vice versa)

Operational Considerations

Deployment

- Most concerns are not Lustre specific
  - Delivery schedules
  - Facilities issues (power, cooling)
  - Installation timeline
- This is your chance to get things installed "cleanly"
  - Physically organize hardware in logical setup
  - Label all cables and hosts
  - Good cable management
- Leave sufficient time for benchmarking and testing
  - You won't get a chance like this again

Benchmarking

- Test everything for performance and functionality
  - I/O to storage
  - Server burn-in tests
  - Network performance
- Test individual components before testing composite setup
- Need to follow a systematic bottom-up approach
- Look for potential bottlenecks and gather data about baseline performance

Benchmarking (cont.)

- Benchmarking examples
  - Storage tests:
    - Single device I/O on single host
    - Multi-device I/O on single host
    - Multi-device I/O on multiple hosts
  - Network tests:
    - One-to-one, many-to-one, and many-to-many network performance tests
    - Lustre's LNet selftest tool is very useful here
  - File system tests:
    - Single host, single OST
    - Single host, multiple OSTs
    - Multiple hosts, multiple OSTs (Hero Run)
- Useful tools: xdd, IOR, lnet_selftest, perftest

Monitoring

- Absolutely essential
- Many tools for monitoring
  - Nagios
  - Ganglia
  - Collectl
  - Telegraf
  - LMT
- Best to integrate with whatever tools are used for the rest of your site
- Lots of metrics that can be monitored
  - Can even integrate job stats from batch system

Monitoring (Cont.)

- Can get lots of mileage from a few metrics
  - Are the Lustre servers up/down?
  - Are all the OSTs mounted?
  - /proc/fs/lustre/health_check
  - Server load
    - Can indicate I/O bottleneck
  - Server memory usage
    - Can identify lots of locks
  - Network interface errors and rates
  - Total file system usage and individual OST usage
  - OST usage spread
    - This can help identify poorly striped files

Monitoring (cont.)

- Users are a good monitoring tool
  - Users may see symptoms before problem is apparent
  - Users can trigger edge cases and expose weak points in the file system
  - Users aren't afraid to tell you something is wrong
- Users are a terrible monitoring tool
  - User information can be ambiguous
    - Lustre is slow…
  - User information can be inaccurate
    - This used to work before…
- Good or bad, it's best not to ignore them

Troubleshooting

- What to do when “Lustre is broken”
  - Try to find a reproducer
  - Does the problem occur on multiple clients or just one?
  - Does the problem occur in batch jobs? Interactive commands?
  - When was the problem first noticed?
- Try to determine if problem is on client or server
- Log at system logs and run basic tests
  - lctl ping
  - Other benchmark tests

System Logs

- Aggregate logs from servers in one location
  - Makes it easier to correlate events and detect patterns
- Lustre log messages can be hard to “decode”
  - Not uncommon to see LustreError messages for things that aren’t really errors
  - When things break, the number of error messages can make it hard to weed out the root cause
- Try to detect bigger patterns in the log messages
  - Learning how to do this just takes some experience

Logs: What Should I Look For?

- **Client evictions**
  - `Haven't heard from client XX in YY seconds. I think it's dead, and I am evicting it`
- **Multiple errors about/from the same node**
  - All servers complaining about single client
  - All clients complaining about a single server
- **Errors with "rc -30"**
  - `medusa-OST0053: unable to precreate: rc = -30`
  - Indicator that block device is not accessible
- **Timeouts**
- **Anything out of the ordinary**
  - Large numbers of message may be indicator

RAS

- Be realistic with uptime/support expectations
  - Don't overpromise
- Hardware selection
  - RAID, dual power supplies, etc.
  - Uniform hardware can be a big help
    - Test file system can be source of spare parts
  - Simple hardware configuration
    - Blades vs. stand-alone servers
    - Diskless servers with NFS root
- Take advantage of Lustre failover feature
  - Doesn't need to be automated failover
  - Aggressive monitoring with simple, well-documented procedures may be just as good

RAS (cont.)

- Lustre has built-in recovery mechanism
  - If server goes down, client I/O should pause until server is back up
  - Not 100% fool-proof, but in my experience, the success rate is very high
  - Allows you to perform some maintenance tasks while file system is online
    - Ex – Reboot a server, rolling upgrades
  - May consider pausing the batch system (just in case)
  - For prolonged work, it is probably better to just schedule a downtime if possible

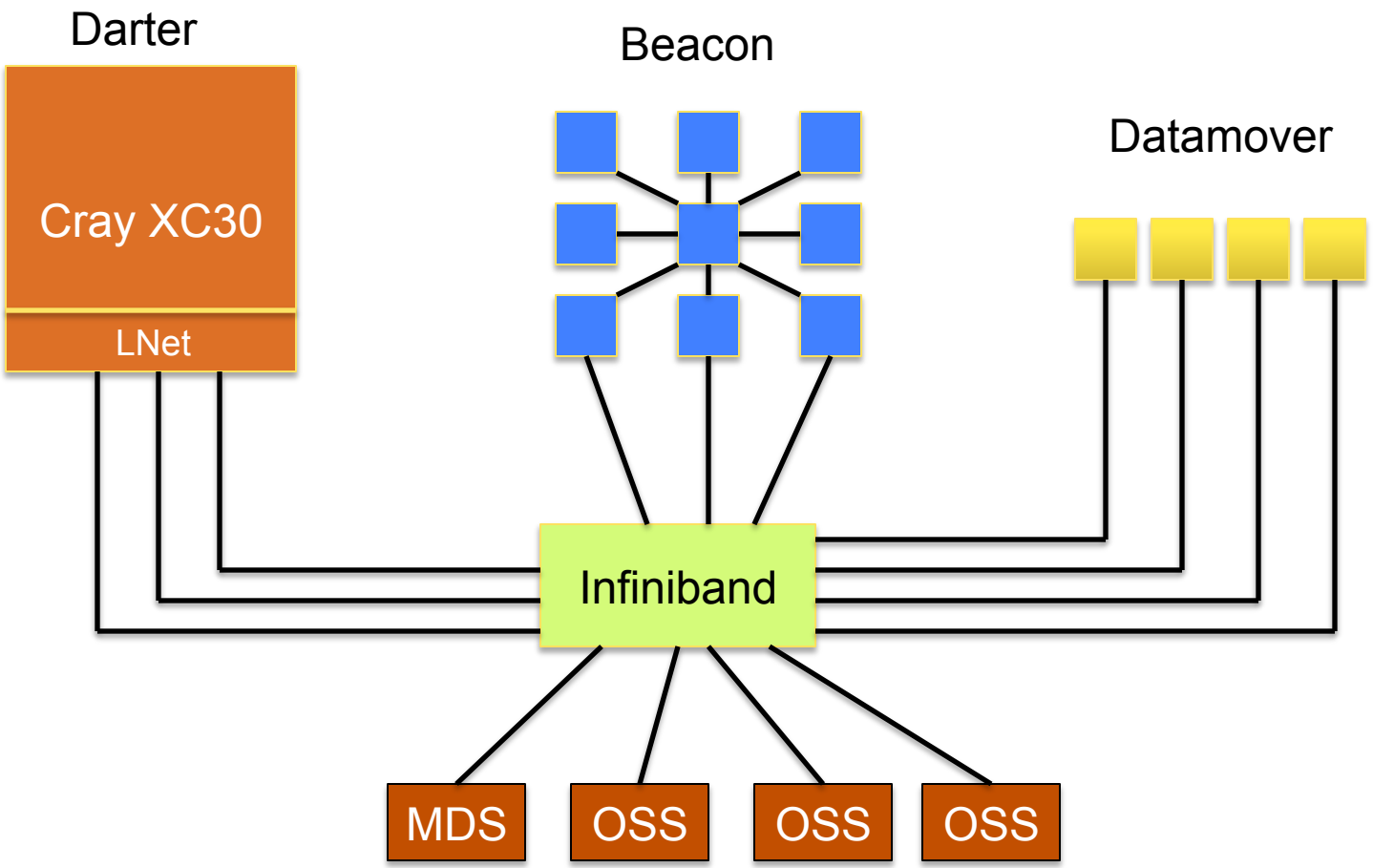
Upgrades

- First question: “Why?”
  - If it ain’t broke, don’t fix it
- Do your homework
  - Gather information about kernels/OS on clients
  - Look at compatibility matrix
    - Make sure new version is supported on clients and servers
  - Look at manual for upgrade notes
  - Do you need to enable new features?
  - Can you do a staged upgrade?
- Test, test, test
- Create a detailed upgrade plan
  - …and an escape plan

Reduce, Reuse, Recycle

- Uniform hardware
- Use existing infrastructure
  - DNS, NTP, monitoring, etc.
  - Ticket system, wiki, etc.
- Leverage existing staff knowledge
  - Cluster management
  - Hardware expertise
- Consider site-wide Lustre file system
- Add on to existing file system

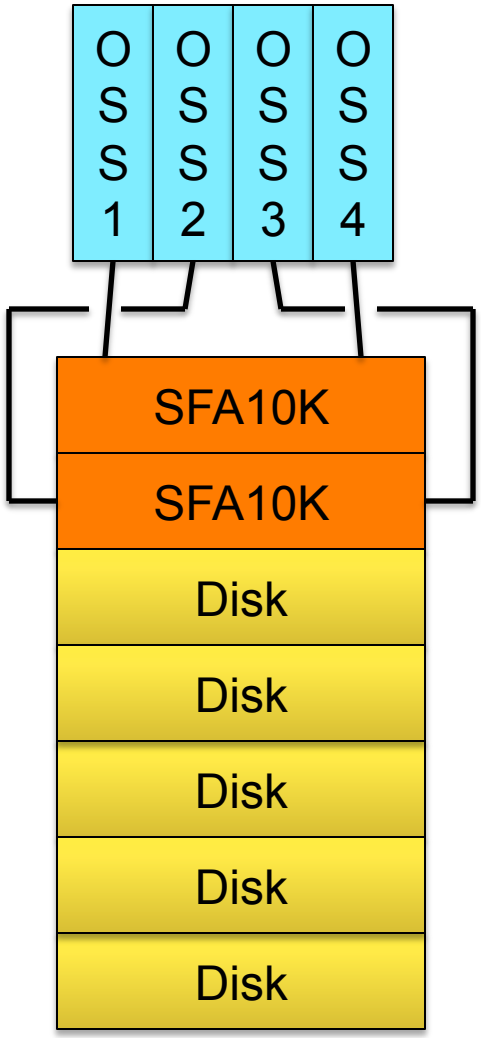
NICS: Medusa File System



Medusa: Networking

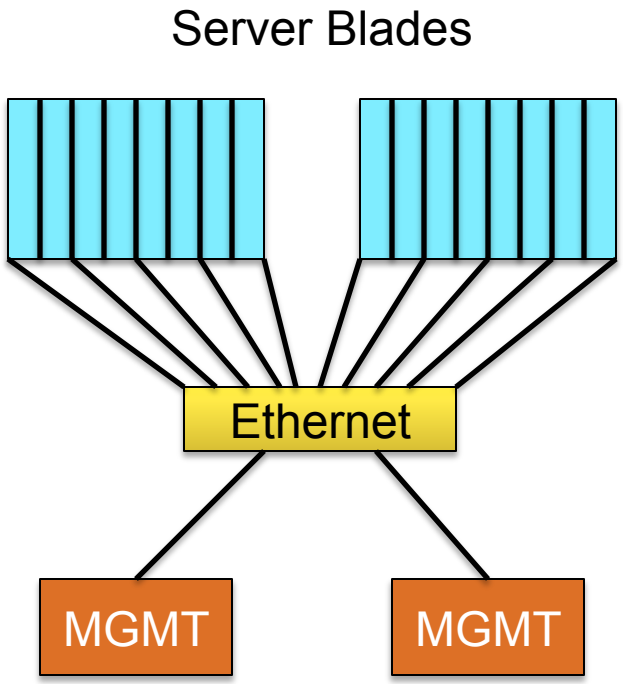
- Compute resources (with few exceptions) do not connect directly to core Infiniband switch
  - Each resource has a set of Lnet routers
- Two IPoIB subnets are reserved for Lustre:
  - 172.16.0.0/16
    - This is the “core” Infiniband subnet
    - Anything connected to core IB switch uses this address space (Lustre servers, LNet routers, etc.)
    - Addresses logically divided by system (e.g. – Data transfer nodes use 172.16.30.*)
  - 172.17.0.0/16
    - Divided into subnets of varying sizes
    - Clusters with internal IB fabric get assigned one of these subnets

Medusa: Storage Building Block



- 4 OSS nodes per SFA10K couplet
- 300 disks → 30x 8+2 RAID6 OSTs
- Nodes in a failover pair are connected to different controllers
- Nodes only see their OSTs and the OSTs for failover partner
- OSTs have consecutive index numbers
  - Easy to map OST index to couplet/ server/LUN

Medusa: Management



Management Services

- DHCP
- TFTP
- NFS
- NTP
- Conman
- Powerman
- Syslog
- Rsync

- **Two blade chassis with 10 blades each**
- **Failover pairs are split between chassis**
- **Blades are diskless**
  - **Use NFS root file system**
- **Two management nodes**
  - **One acts as cold spare**
- **All sys admin work is done from management nodes**

K.I.S.S.

- When working with limited resources, always strive for simplicity and uniformity
  - Applies to everything from hardware to software to policies
  - The simpler it is, the easier it will be to train others
- Be realistic when determining requirements, features, and priorities
- Documentation is your friend
  - Keeps you from having to rediscover procedures each time
  - But this is easier said than done

Lustre Resources

- Web site

  http://lustre.org

- Documentation

  http://lustre.org/documentation

- Mailing Lists

  http://lustre.org/mailing-lists

- Conferences
  - Lustre User Group (LUG)
  - Lustre Administrator and Developer Workshop (LAD)
  - International Workshop on the Lustre Ecosystem

Questions?