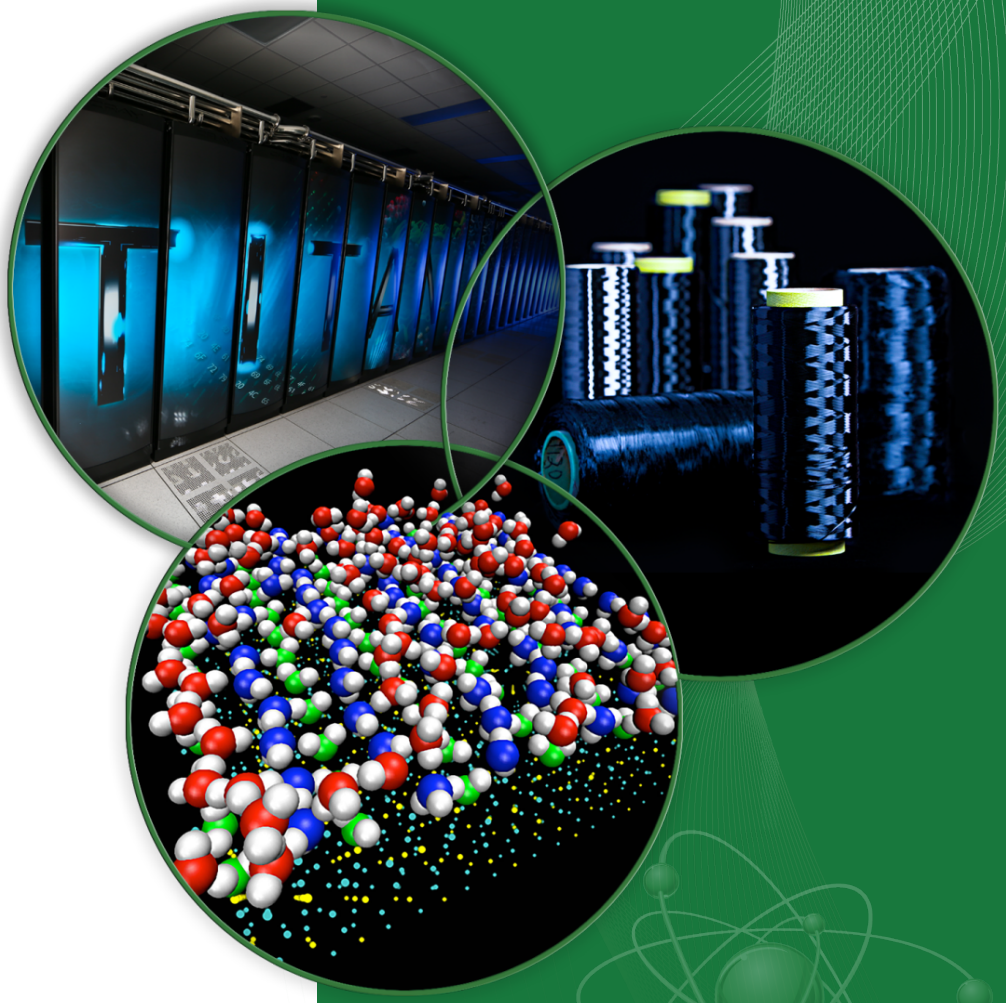
Robust Health Monitoring

- Lustre
- InfiniBand
- Storage Arrays



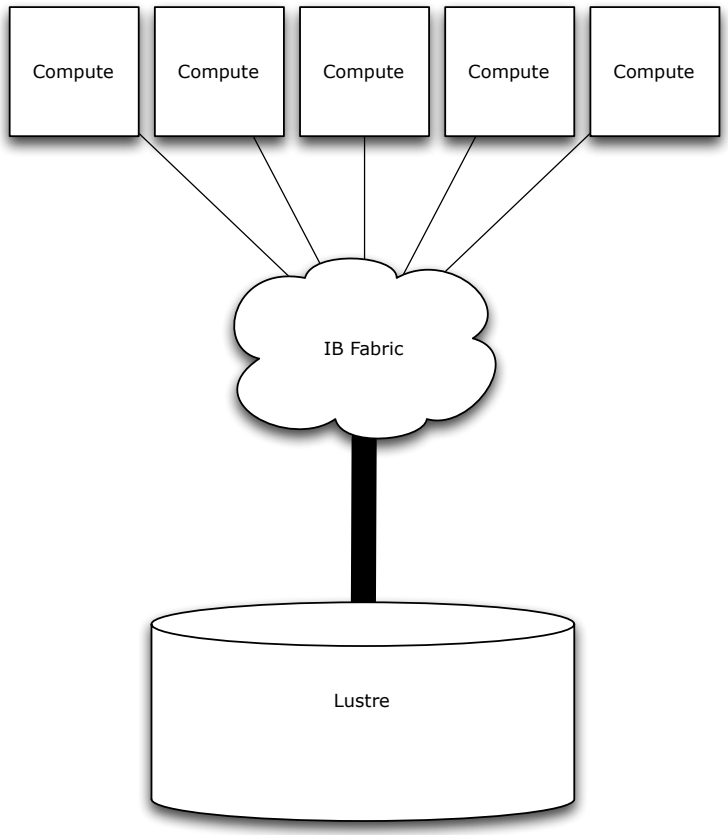
Blake Caldwell
HPC Operations
Oak Ridge Leadership Computing Facility
Oak Ridge National Laboratory

March 3rd, 2015

ORNL is managed by UT-Battelle
for the US Department of Energy

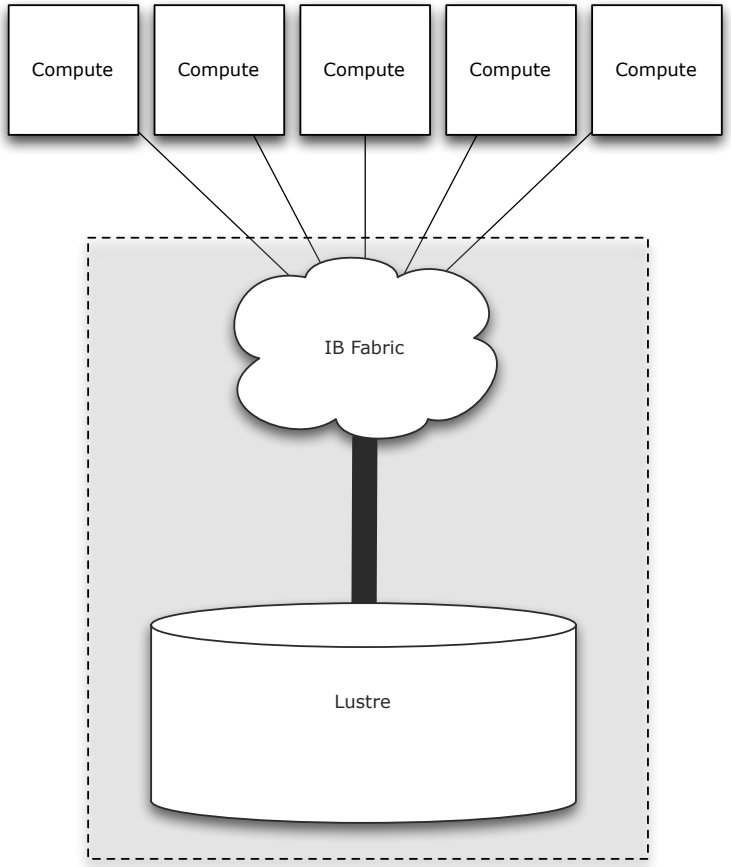
OAK RIDGE National Laboratory | OAK RIDGE LEADERSHIP COMPUTING FACILITY

Just a filesystem, right?



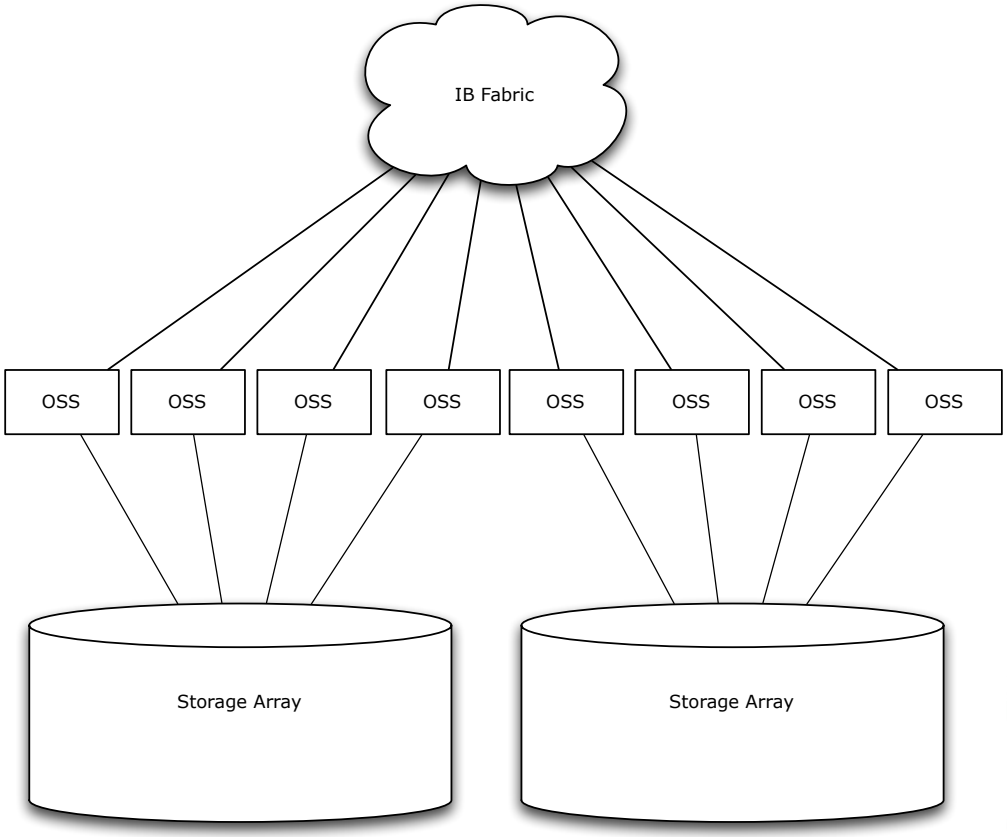
High-level view: a compute cluster connected to a parallel file system

Just a filesystem, right?



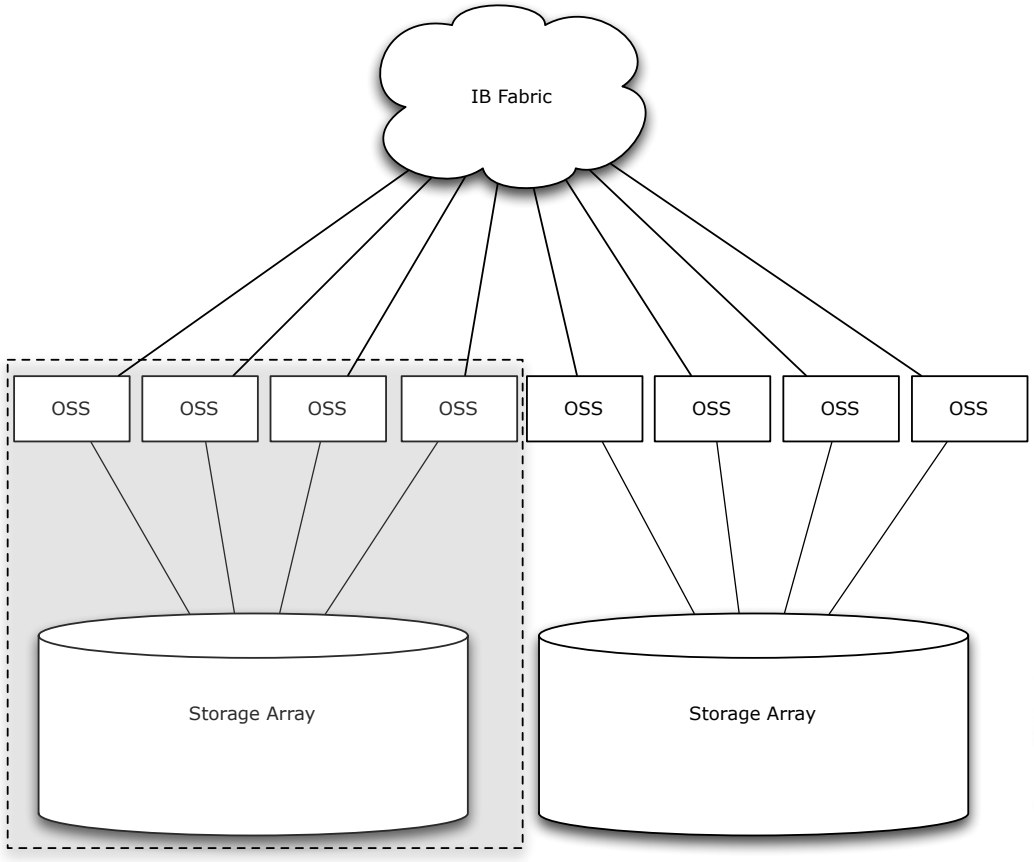
High-level view: focus in on just the filesystem infrastructure

Not too bad...



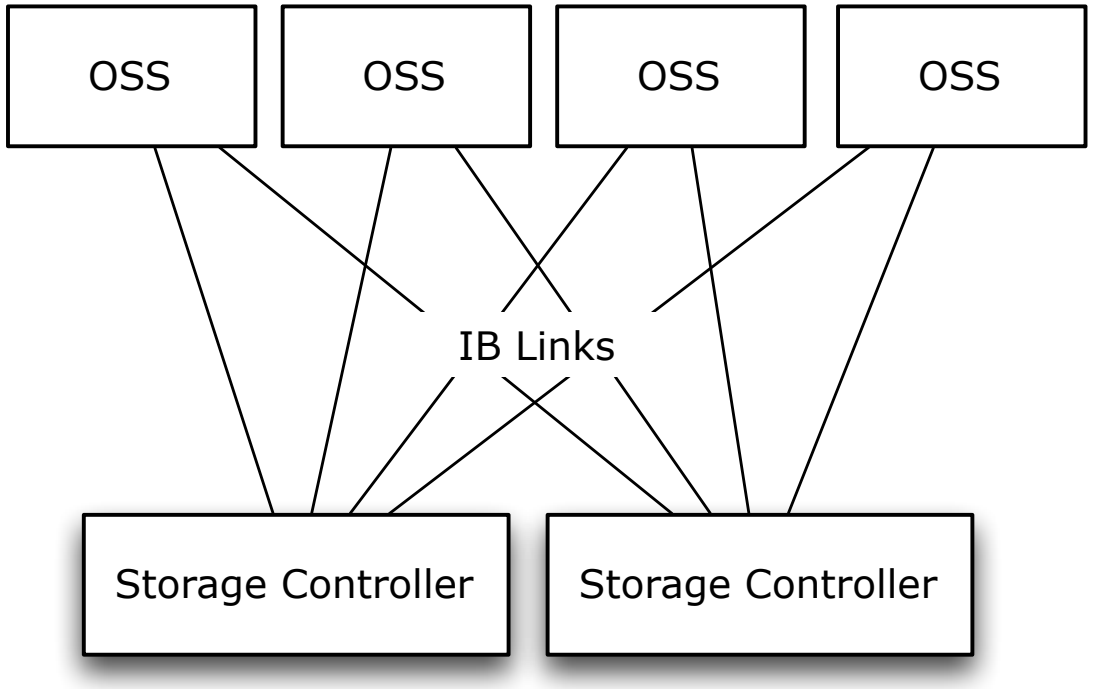
View of PFS: composed of both servers and storage arrays

Not too bad...



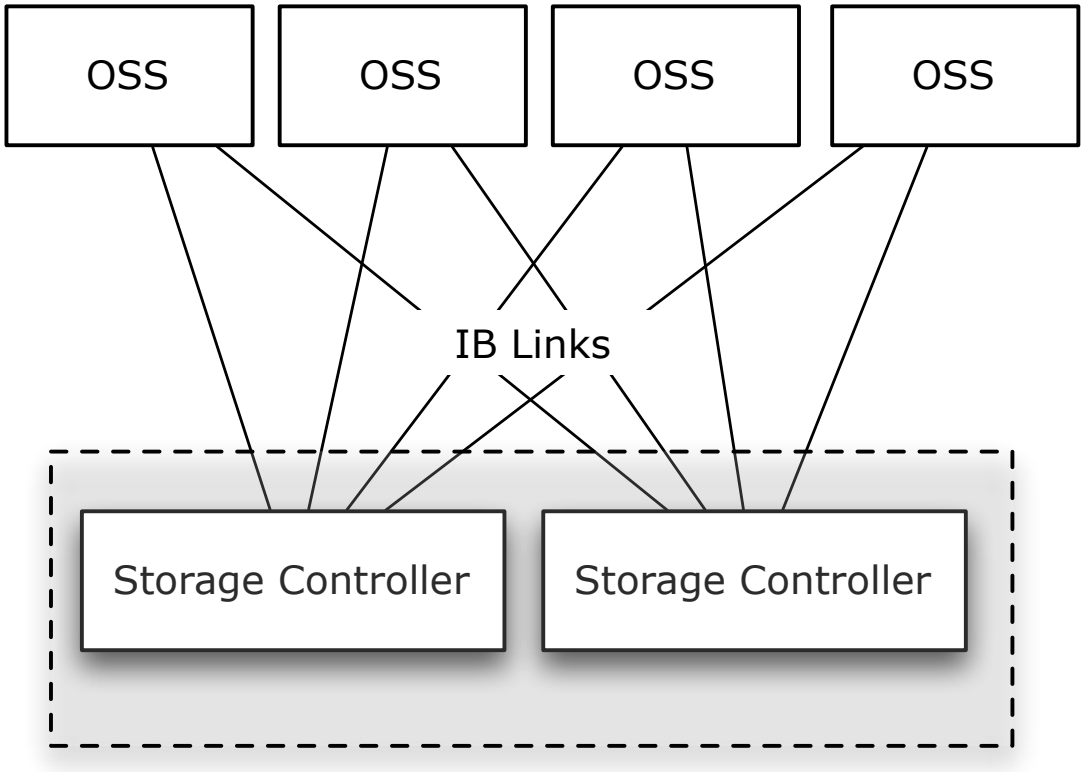
View of PFS: organized into scalable units

More than we thought…



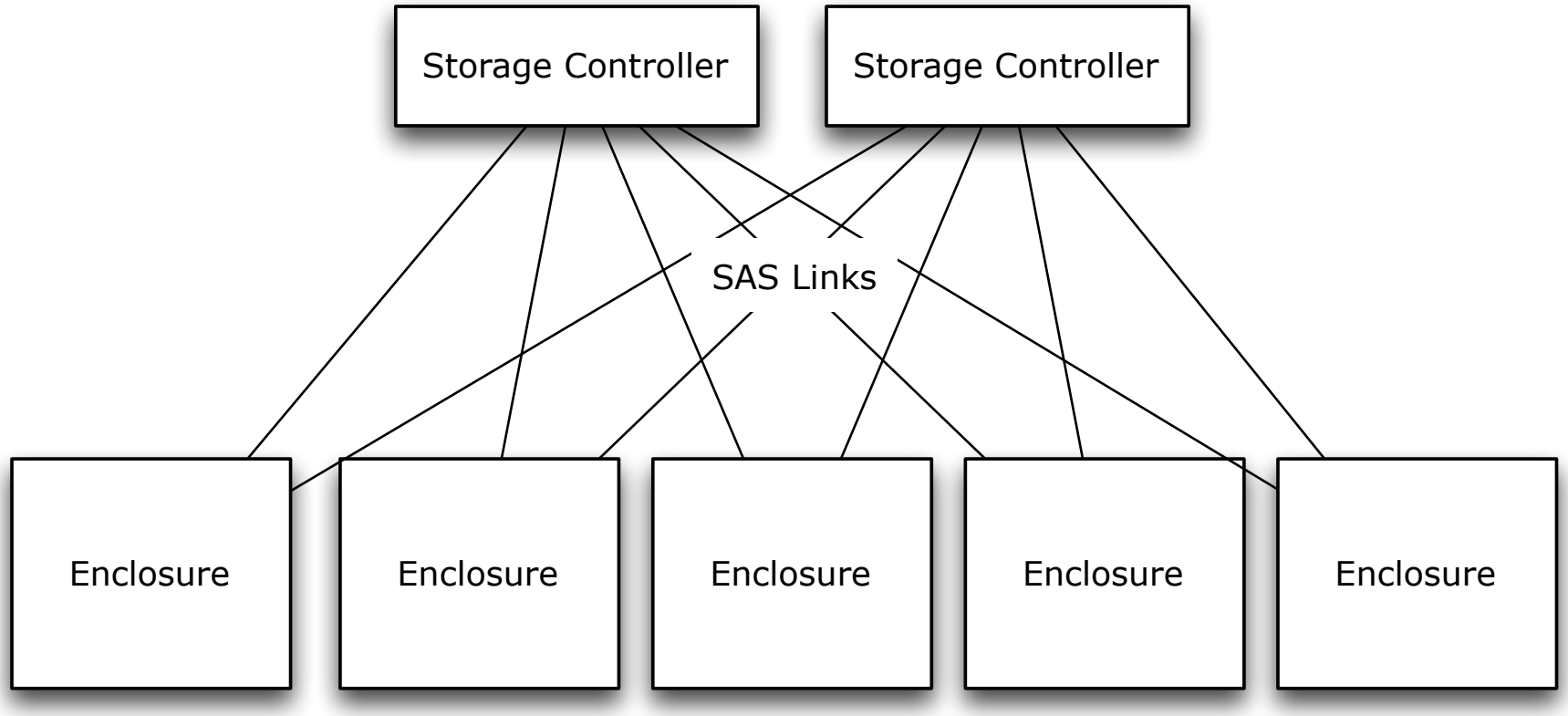
View of Scalable Unit: a mesh of IB links for throughput and high availability

More than we thought...



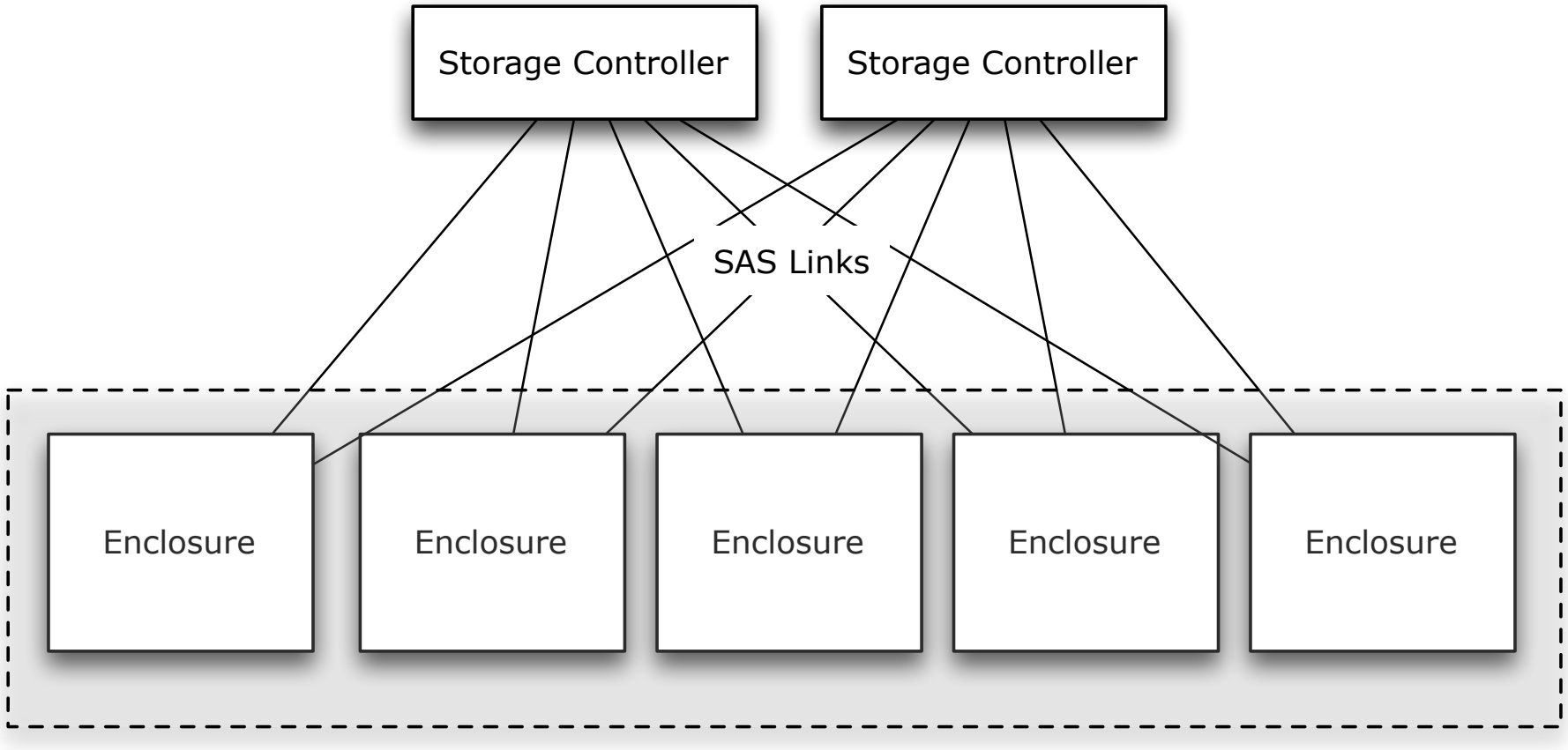
View of Scalable Unit: the storage array has individually monitor-able components

Another layer?



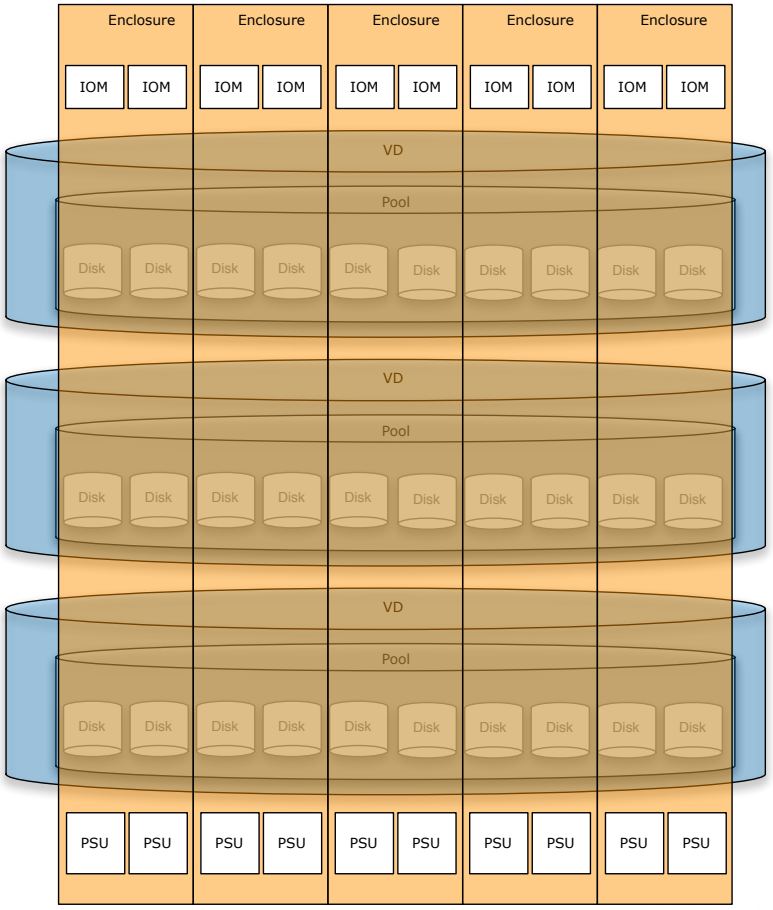
View of Storage Array: redundant SAS links connect disk enclosures

Another layer?



View of Storage Array: more complexity still within enclosure units

We need some help!



View of Storage Enclosures: contains pools, VDs, IO modules, power supplies, and hundreds of disks, all of which can fail (**so we monitor them**)

This talk will cover…

- A monitoring infrastructure for Lustre
- Tools used for monitoring layers
  - DDN SFA check (block)
  - IB health check (network)
  - Custom scripts (Lustre)
- How common kernel LustreError log messages correlate with monitored events

Monitoring Infrastructure

- Nagios: for ***alerting***
- Splunk: for ***information*** to be investigated
  - Send out snippets of syslog to filesystem admins
  - Interesting DDN SFA logs
    - SCSI sense errors (predict PD failure)
    - RAID parity mismatches
  - Rebooted OSS/MDS
  - Read-only LUN
  - Memory errors
- LMT/others: for ***performance*** monitoring

Nagios Infrastructure

- All Lustre hosts and DDN controllers are *hosts*
- *Service* checks: (e.g. sfa_check, ib_health_check) 13,000 in OLCF!!
  - Check commands:
    - check_snmp_sfa_health.sh
    - check_snmp_extend.pl
- Hosts run scripts via snmp extend (snmpd.conf)

```
extend monitor_ib_health /opt/bin/monitor_ib_health.sh
```

1st layer: backend storage arrays

- Hardware failure events:
  - Disk failures
  - Enclosure power supplies
  - Inter-controller links
- Assess the impact on:
  - Redundancy
  - Performance
  - Cache status

SFA Check

- Periodic execution of API commands on all DDN arrays
  - Asynchronous from Nagios polling
- Python multiprocessing library
  - Manages a pool of workers (one per SFA couplet)
  - Times out stuck workers and propagates error to SNMP
- Modular design
  - All component checks perform doHealthCheck() for overall component status (OK, NON_CRITICAL, CRITICAL)
  - Additional component-specific checks (e.g. ICLChannelCheck)
    - InfinibandPortState == ACTIVE
    - InfinibandCurrentWidth == 4
    - ErrorStatisticCounts[SymbolErrorCounter] < 20
- Checks configuration of pools (caching, parity/integrity checks)

SFA Check (2)

Classes that return health status (OK, NON_CRITICAL, CRITICAL):

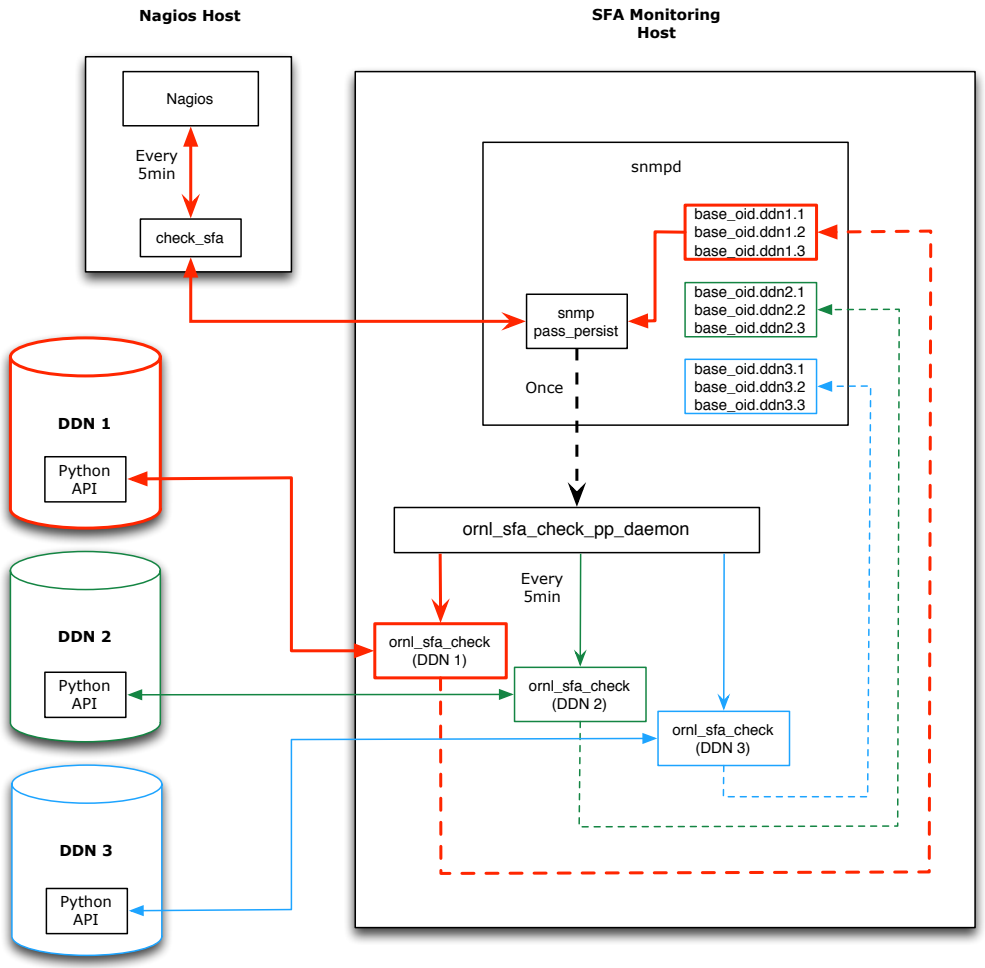
- Physical disks (PD)
- Virtual disk (VD) objects (made up of PDs)
- Pools (made up of VDs)
- Inter-controller links (ICL)
- Power supplies
- Host channels (HCAs)
- Internal configuration disks
- SAS expanders
- SAS expander processor (SEP)
- UPS units (external)
- IO Controllers (IOC)
- Fans
- RAID processors
- Voltage sensors
- Temperature sensors

poolCheck() example

- Top-level generic check: doHealthCheck()
- Specific checks:
  - State: degraded, no redundancy, critical
  - Rebuild state
  - Ownership by controller
  - Bad block count (*)

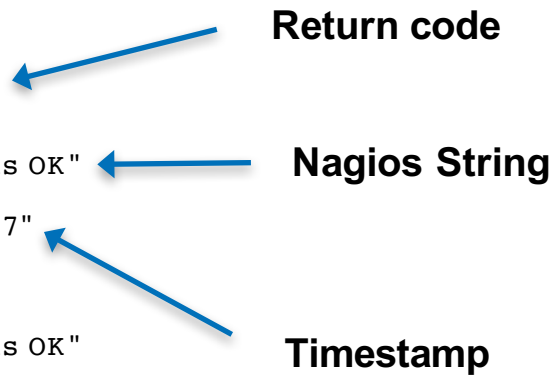
(*) only in CLI extended mode (-x)

SFA check design



SNMP OID structure

```
# snmpwalk -v2c –c public monitor_host .1.3.6.1.4.1.341.49.1
SNMPv2-SMI::enterprises.341.49.1.12.97...98.49.1 = INTEGER: 0
SNMPv2-SMI::enterprises.341.49.1.12.97...98.49.2 = STRING: "All Checks OK"
SNMPv2-SMI::enterprises.341.49.1.12.97...98.49.3 = STRING: "1424574807"
SNMPv2-SMI::enterprises.341.49.1.12.97...98.50.1 = INTEGER: 0
SNMPv2-SMI::enterprises.341.49.1.12.97...98.50.2 = STRING: "All Checks OK"
SNMPv2-SMI::enterprises.341.49.1.12.97...98.50.3 = STRING: "1424574807"
SNMPv2-SMI::enterprises.341.49.1.12.97...99.49.1 = INTEGER: 0
SNMPv2-SMI::enterprises.341.49.1.12.97...99.49.2 = STRING: "All Checks OK"
SNMPv2-SMI::enterprises.341.49.1.12.97...99.49.3 = STRING: "1424574807”
```



SFA Check Output

```
[fpc@or-mgmt01 ~]$ /opt/lustre/bin/ornl_sfa_check.py or-ddn-a1 or-ddn-b1 or-ddn-c1 or-ddn-d1
POOL: 2 Checks WARNING - Index 53 DEGRADED
POWER SUPPLY: CRITICAL
All Checks OK
All Checks OK
[fpc@or-mgmt01 ~]$ /opt/lustre/bin/ornl_sfa_check.py -x or-ddn-a1 or-ddn-b1 or-ddn-c1 or-ddn-d1
or_ddn_a Check Summary:
------------------------
Messages from check POOL
Messages from check VIRTUAL DISK
VIRTUAL DISK: 2 Checks WARNING
VIRTUAL DISK Health: OK   Index: 37; Child Health: NON_CRITICAL
VIRTUAL DISK Health: OK   Index: 53; Child Health: NON_CRITICAL

POOL: 2 Checks WARNING
POOL Health: NON_CRITICAL  Index: 37
POOL Health: NON_CRITICAL  Index: 53

or_ddn_b Check Summary:
------------------------
All Checks OK
```

SFA Check Output (2)

```
or-ddn-c Check Summary:
------------------------
Messages from check POOL
Messages from check VIRTUAL DISK
VIRTUAL DISK: 1 Check WARNING
VIRTUAL DISK Index: 4; BadBlocks: 1

POOL: 1 Check WARNING
Pool Index: 4; BadBlocks: 1


or-ddn-d Check Summary:
------------------------
Messages from check POWER SUPPLY
Messages from check CONTROLLER
CONTROLLER: 2 Checks WARNING
CONTROLLER Health: OK  Name: A Index: 0; Child Health: CRITICAL
CONTROLLER Health: OK  Name: B Index: 1; Child Health: CRITICAL

POWER SUPPLY: 1 Check CRITICAL
POWER SUPPLY Health: CRITICAL  EnclosureIndex: 10 Location: PSU 2    Index: 2
```

Other block-level OSS checks

- Multipath
  - 2 paths to each LUN
- SRP daemon
  - 2 processes (for each IB port)
- Block tuning
  - Udev rules correctly set for each block **AND** dm device
    - max_sectors_kb = max_hw_sectors_kb
    - scheduler = “noop”
    - nr_requests, read_ahead_kb

2nd layer: network interconnect (IB)

What we're looking for:

- Faulty host IB links to:
  - Storage arrays
  - Top of rack switches
- Fabric health (switch ports and inter-switch links)
  - Error counters, degraded/failed links
  - IB topology/forwarding routing

2nd layer: network interconnect (IB)

What are we looking for:

- Faulty host IB links to:
  - Storage arrays
  - Top of rack switches
- Fabric health (switch ports and inter-switch links)
  - Error counters, degraded/failed links
  - IB topology/SM routing

TODO

IB Health Check

- HCA and local link health
  - Local errors check (HCA port)
  - Remote errors check (switch port)
- PCI width/speed of each HCA
  - Identify failed hardware or firmware issues
  - Appropriate slot placement
- Port in up/active state
- Link speed/width matches capability
- SM lid is set

IB Health Check Output

Host	Service	Status	Last Check	Duration	Attempt	Status Information
ninja17	IB_HEALTH	CRITICAL	08:55:47	3d 10h 12m 35s	10/10 #2	mlx4_0-ib0:CRITICAL - Link rate: 40Gbps is inappropriate for FDR IB ::

select all (hosts) - unselect all - all problems - all with downtime

Host	Service	Status	Last Check	Duration	Attempt	Status Information
atlas-spare03	IB_HEALTH	CRITICAL	08:58:21	4d 21h 36m 7s	10/10 #20	mlx4_1-ib0:CRITICAL - Link rate: 20Gbps is inappropriate for FDR IB - IB HCA errors for "atlas-spare03 mlx4_1" GUID 0x2c90300185511 port 1: [SymbolErrorCounter == 5583] ::

select all (hosts) - unselect all - all problems - all with downtime

Fabric Monitoring

- Issues to resolve:
  - Scaling health checks to 2000 ports
  - Discover new trends, not thresholds crossed
  - Storage and retrieval
  - Selective presentation of information
    - Alerting interface
    - Performance monitoring interface

Nagios scripts for IB Fabric and SFA checks

- https://github.com/bacaldwell/scalable-monitoring
  - DDN SFA checks
    - sfa_check
  - Monitor IB fabric
    - monitor_ib_health

3rd layer: Lustre monitoring

- /proc/fs/lustre/health_check
- /proc/mounts
- /proc/fs/lustre/devices
  - osd, osp, mgc, mds, mdt in UP state
- /proc/sys/lnet/stats
  - queued LNET messages
- /proc/sys/lnet/peers
  - connection state and queued messages to other servers
- lfs check servers
- llstat

Lustre monitoring tools

- Capturing monitoring metrics in database for analysis
  - Robinhood (changelogs)
  - LMT (collectl)
- Scripts to collect using llstat, plot with gnuplot or matplotlib
  - See “Monitoring the Lustre file system to maintain optimal performance,” by Gabriele Paciucci, LAD 2013

How monitoring helps with these kernel messages

- Read-only OST
  - Trigger lustre_health, Splunk alerts

```
Oct 25 14:35:09 oss1b8 kernel: [  726.179459] LDISKFS-fs (dm-25):
Remounting filesystem read-only
Oct 25 14:35:09 oss1b8 kernel: [  726.445822] Remounting filesystem
read-only
```

More Lustre kernel messages

- Bulk I/O failure and LNET timeouts
  - Check IB_health for errors
  - Probably useful syslog messages
    - After determining root cause, can a Splunk alert be written?

```
Dec 8 12:11:52 atlas-oss3b4 kernel: [1032388.030434] Lustre: atlas2-OST009b:
Bulk IO read error with 3b57a9ed-bec6-9b0d-7da8-04d696e1a7f2 (at
10.36.202.138@o2ib), client will retry: rc -110
```

More Lustre kernel messages

- OST unreachable
  - Health checks for OSS on which OST resides
  - Next check IB fabric (are messages getting lost?)

```
Dec 8 12:11:52 dtn04.ccs.ornl.gov kernel: Lustre: atlas2-
OST009b-osc-ffff880c392f9400: Connection to service atlas2-
OST009b via nid 10.1.0.145@o2ib was lost; in progress
operations using this service will wait for recovery to
complete.
```

More Lustre kernel messages

- Client eviction/reconnection cycle
  - This was caused by MTU mismatch on IB fabric
  - Lessoned learned: ensure lower layers are healthy

```
Mar  3 11:14:09 atlas-oss4e1.ccs.ornl.gov kernel:
[2428963.716071] Lustre: atlas2-OST0068: Client c618c441-fb87-
27e9-21ae-6c345ddc40c8 (at 10.1.0.151@o2ib) reconnecting

Mar  3 11:14:09 atlas-oss4e1.ccs.ornl.gov kernel:
[2428963.740253] Lustre: Skipped 6 previous similar messages
```

Conclusion

- OLCF monitoring best practices
- Tools used at each layer
  - DDN SFA check (block)
  - IB health check (network)
  - Custom scripts (Lustre)
- Applicability to common filesystem problems

Thank You

Blake Caldwell blakec@ornl.gov

Monitoring and visualization tools:

https://github.com/bacaldwell