High Performance Computing (HPC) Storage Challenges

Paul Krueger, Ph.D.

Cray, Inc.

Acknowledgments

- Several people helped with this presentation
  - Matt O'Keefe Alvarri
  - John Carrier Cray I/O R&D
  - Curt Fawcett Cray I/O Devices Integration & Test
  - John Levesque Cray Application Performance Specialist
  - Peter Rigsbee Cray I/O Marketing
  - Steve Scott Cray CTO
  - Scott Studham ORNL

My Personal HPC Value Metric

Maximize:

Compute-cycles

applied *usefully* to

the most *valuable*

customer problems

per *$-cost*

Optimizing for the HPC Metric

- Scalability: *Add cycle capacity*
- Efficiency: *Map computation to hardware to maximize use*
- Reliability: *Maximize availability of cycles you have*
- Developer Productivity: *Make it easy to get the most value from cycles used*
- Economy: *Decrease costs to own & operate*
- Adaptability: *Adjust hardware and software to demands*

These strategies interrelate! Optimize globally, not locally.

Scalability Challenges

Customer Demand Trends

- Last year's keynote predicted demand for:
  - 0.5 PF system in 2008 (0.5 TB / sec parallel I/O)
  - 5 PF system in 2012 (5.0 TB / sec parallel I/O)
  - Demand trend is 10X every 4 years
- Actual customer requests:
  - 1 PF system delivered in 2008
  - 10-20 PF system delivered in 2010
  - 100 PF system delivered in 2012
  - 1 EF (Exaflop) delivered in 2014
    - 1 PB / sec parallel I/O
  - Demand trend is 10X increase every 2 years

I/O Demand Example

- One solicitation (Aug 06) states the I/O requirements for a 2008 delivery:
  - 1 PF sustained computation
  - One trillion files in a single file system
  - 32K file creates per second
  - 10K metadata operations per second for checkpoint/restart
  - Attached disk access at 256 GB/sec
  - Total peak I/O bandwidth of 4.7 TB / sec.
  - 175 TB memory

- Observations
  - Customers now buy bandwidth, not capacity
  - Storage now more networked than direct
  - Peak I/O bandwidth requested is increasing, not decreasing
  - Cost of I/O as a fraction of total system cost is increasing

HPC Application Data Sets

"Data sets are expected to grow rapidly. The expected median data set within three years ranges from 200 GB to 600 GB, with the most dramatic estimate of growth in Commercial Enterprises."

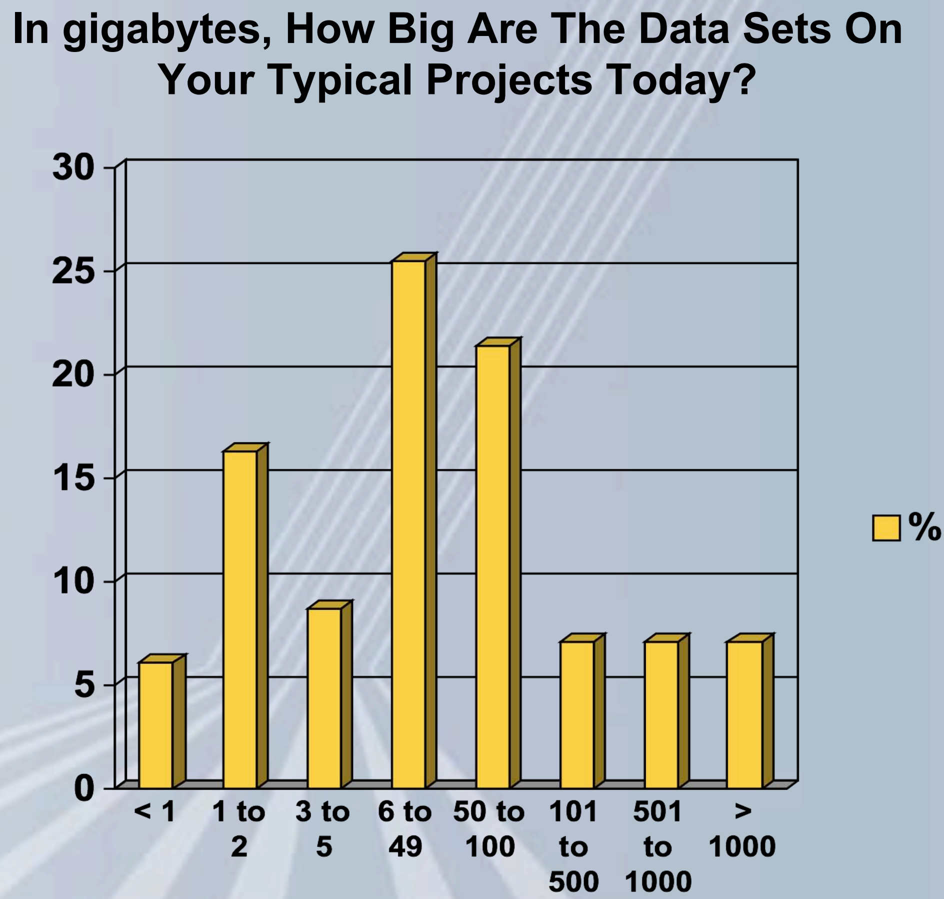
The Development of Custom Parallel Computing

Prepared by: Simon Management Group

September 2006

HPC Application Data Set Sizes

Developers work with both small and large data sets – 6.1% of respondents cited less than 1 GB, while 7.1% of respondents cited greater than 1,000 GB (Figure 2). Commercial Enterprises, In comparison to Educational Institutions and Government Labs, reported the smallest median data sets today. However, Commercial Enterprises also anticipated the largest growth, indicating that they expect the median data three years from now to be 600 GB, compared to expectations of 200 GB (Educational Institutions) and 500 GB (Government Labs).



"The Development of Custom Parallel Computing"
Simon Management Group
September 2006

The Looming HPC I/O Crisis

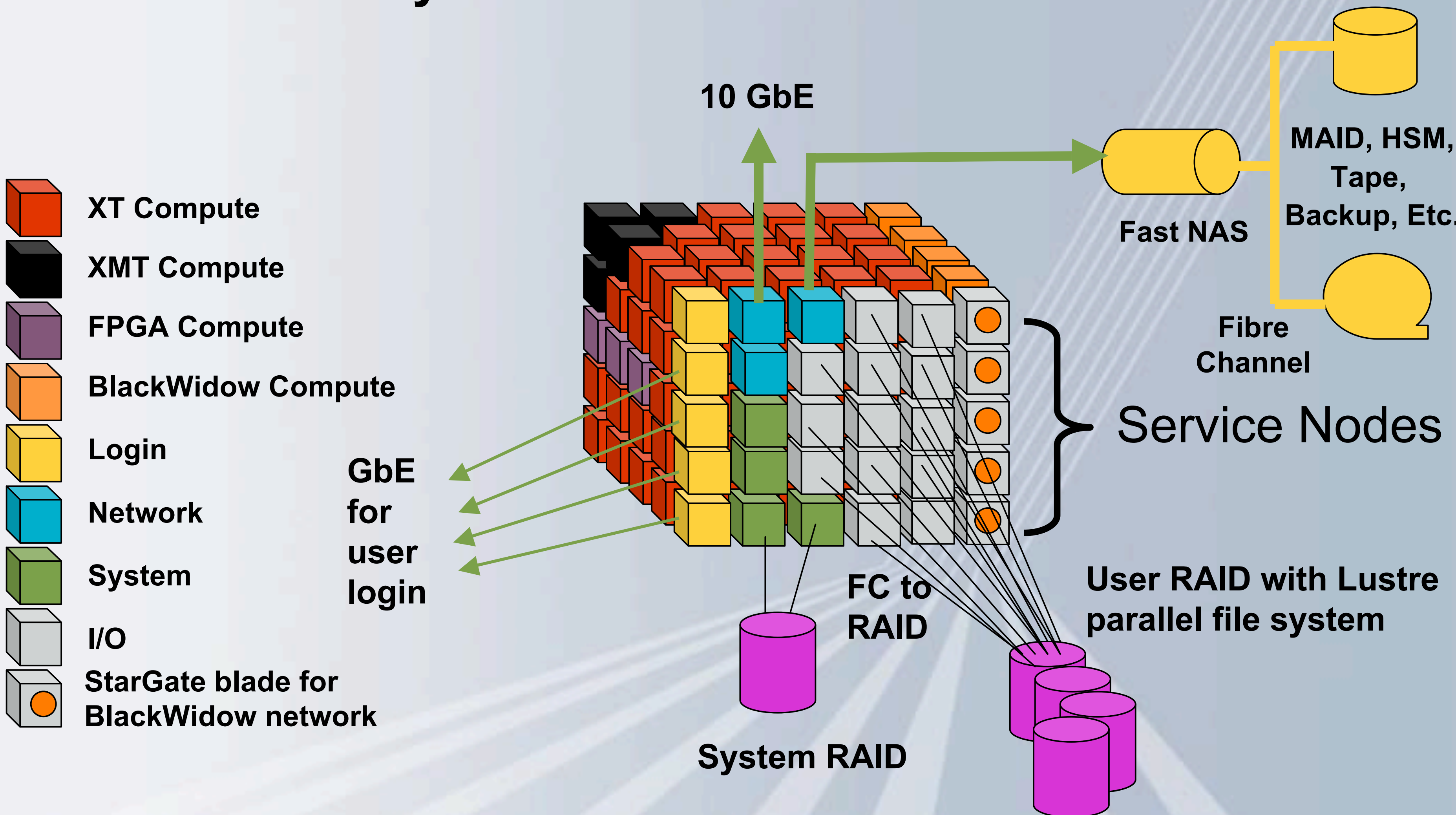
- Historical increase in drive bandwidth is 30% per year
- Application data set sizes growing at 44% per year
- HPC Flops demand growing at 216% per year
- Best current file systems targeted to ~50 GB/sec
- A hypothetical Exaflop system in 2014:
  - Needs ~500X the number of drives needed for a 300TF system in 2007 to maintain comparable bandwidth
  - Needs 20,000 file systems of 50 GB/sec each (or a 1 PB/sec FS)

Exploring the Idea of a Storage Cache

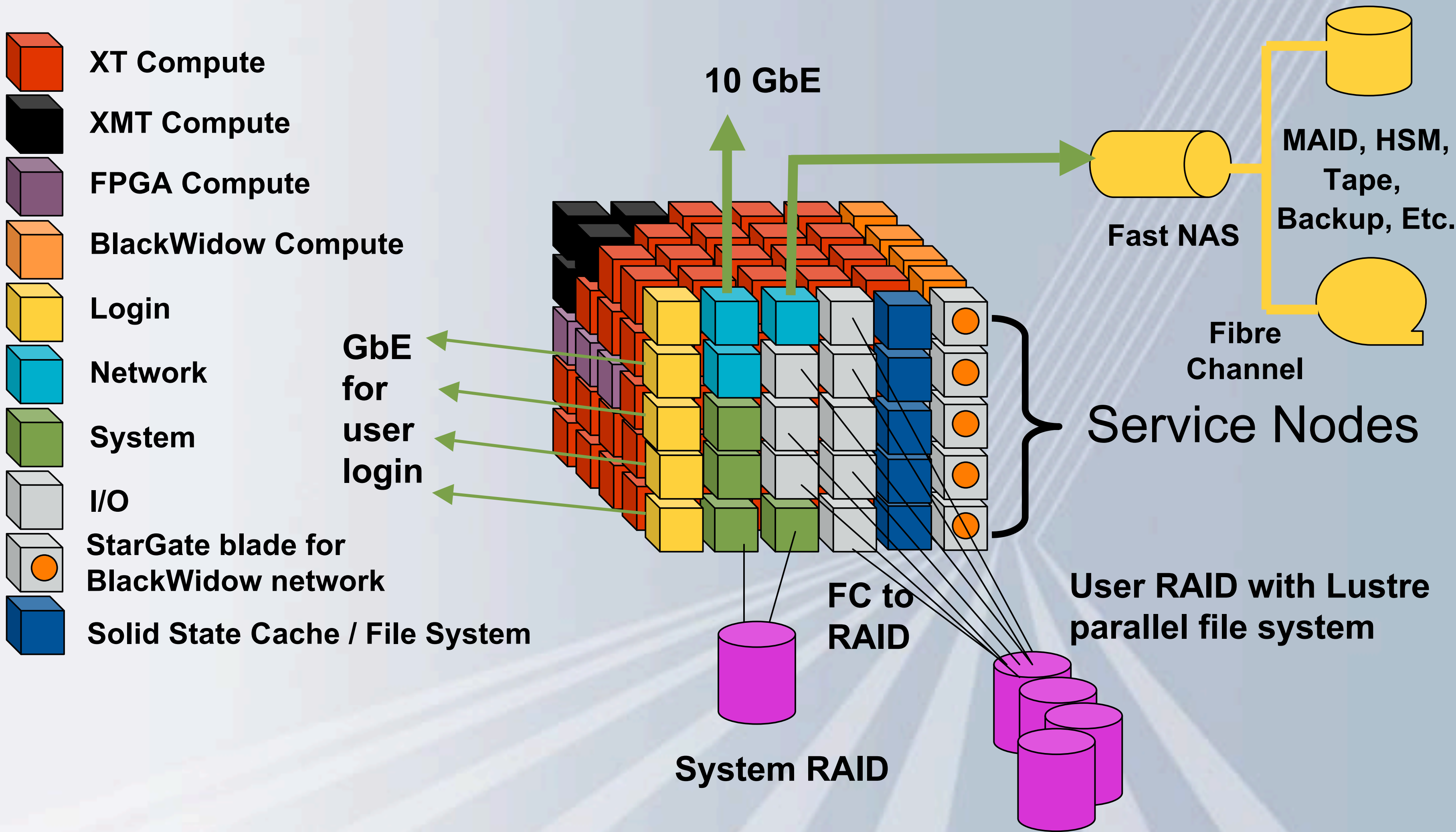
- Requirements:
  - Sufficient size to accommodate application data sets
  - Sufficient speed and bandwidth to minimize application stalls
  - Exploit application reference patterns to pre-load and empty cache
  - Data movement to/from cache must be coordinated with application execution, but scheduled asynchronously
  - Cache must be parallel, distributed, shared and globally coherent
  - Just another tier of very fast storage?
- Object-Based storage techniques will have a role to play …
  - Objects in cache as well as on physical media
  - Object migration to/from cache
  - Pre-staging objects in best location for next application to access it
  - Partial objects
  - Object replication or distributed objects

Integrated Multi-Architecture Systems

Cray Rainier Infrastructure



Hypothetical Ranier Enhancement



Challenges at Large Scales

Challenge #1: Performance! Increase bandwidth and operation rates of hardware and software at least as fast as capacity. Keep latency on the order of current locally-attached I/O.

Challenge #2: Keep administrative complexity and costs constant for all scales.

Challenge #3: Eliminate the need to manually move data from system to system while maintaining the performance characteristics of locally-attached storage.

Challenge #4: Make effective use of the highest-bandwidth commercial channels that are available.

Challenges at Large Scales (cont.)

Challenge #5: Every type of operation must be viable at massive scales (Meta-data operations, security authentication, credential generation, file/object accesses, attribute query/set operations).

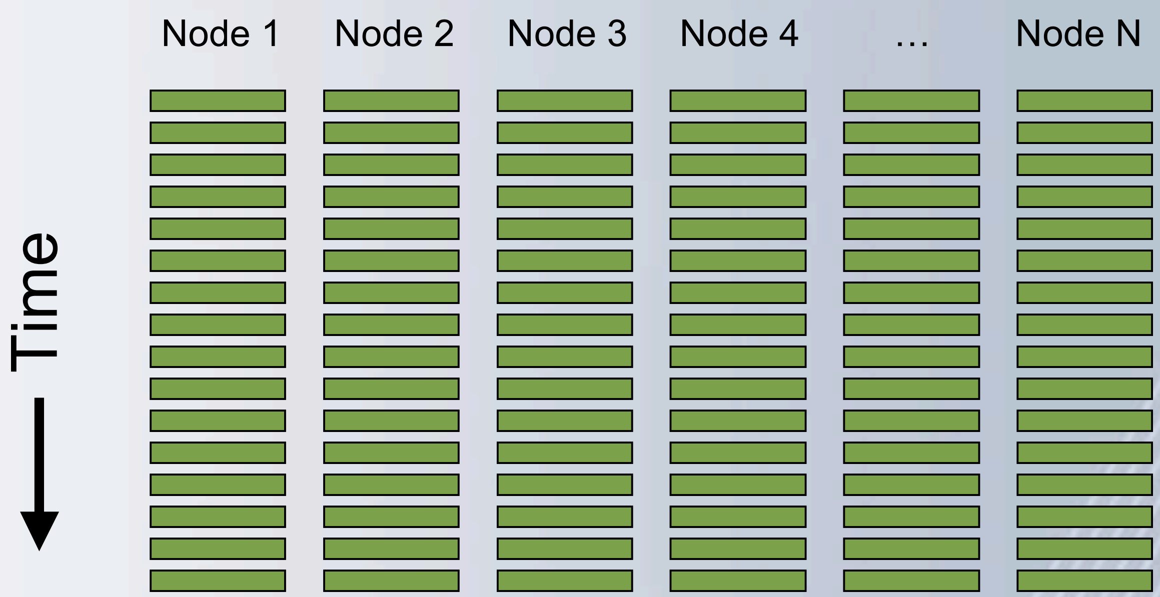
Challenge #6: Test/validate other than running at scale.

Challenge #7: Instrument storage and system software to gather all data necessary to solve any problem that occurs.

Challenge #8: Keep recovery times acceptable at very large scales.

Efficiency Challenges

Ideal Parallel Performance



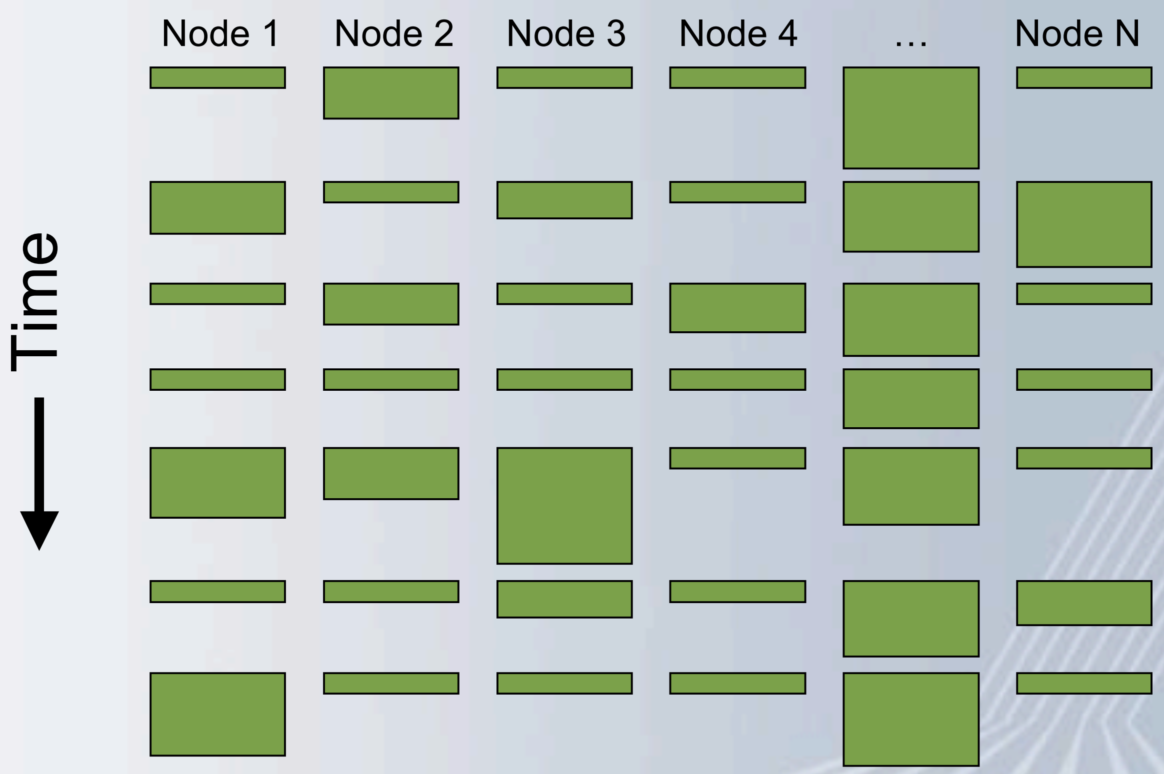
Perfect distribution of the load across all N processors.

Only small amounts of wait time for synchronization.

No node waits for any other.

Minimizes time required for the computation.

Parallel Performance with Unbalanced Load



Widely varying loads on N processors.

Nodes wait at each time-step for node with biggest load.

Time required for computation increases dramatically.

Jitter

- Jitter is similar to having unbalanced loads
- Caused by extra unplanned work for a node and/or unexpected delays in executing the planned work
- Delays node’s arrival at global synchronization points
- Other nodes wait; doing nothing
- Typically occurs at a small subset of nodes in any time-step
- Often random in nature
- Cannot be “fixed” by the application developer
- Examples:
  - Routine OS service Interrupts
  - Error Retries or Other Fault Handling
  - Unplanned Variability of I/O or Internal Data Transfers

Jitter from Storage & Parallel File Systems

- Synchronization of Parallel I/O
  - Any load imbalance introduces a form of “I/O Jitter”
  - I/O completes only as fast as the slowest drive
  - Max I/O time goes up dramatically
- Any asynchronous operation that results in interrupts on compute resources can cause jitter
- Error Recovery also introduces jitter
  - Wider stripes across bigger arrays increase probability of this
  - Less expensive SATA drives also increase probability
  - RAID recovery times measured in seconds to minutes!

Storage Access Jitter

If one storage access out of every 10,000 takes significantly longer than the rest, it can have an extremely negative impact on overall application performance. Within a couple of years it will be 1 out of every 100,000.

Challenge #9: File/object systems and I/O devices must maintain a nearly uniform quality of service within an application and create no arbitrary delays.

Challenge #10: File/object systems and I/O devices must tolerate arbitrary delays and continue normally.

Reliability Challenges

HPC Reliability Issues

- Application Survivability
  - Critically important to HPC
  - Not enough just to keep the “system” up
- Fault Tolerance
  - Mandatory to have hardware and software resiliency strategies for all storage accesses
  - Tolerate long delays caused by recovery of other subsystems
  - Performance degradation is a form of fault
- Fault Prevention
  - Failures must be identified and corrected the first time they occur
  - Issues of scale make fault replication challenging
  - Fault data collection must be a priority for OBS software

Reliability Challenges

Challenge #11: Provide sufficient profiling and performance monitoring to identify root causes after the first occurrence of a fault

Challenge #12: Institute a resiliency strategy for each typical hardware or software fault. Assure applications survive as many types of fault as possible:

- Application compute components
- Meta-data service components
- Storage components
- Communication channels

Developer Productivity Challenges

HPC Developer Productivity Issues

- HPC Development
  - Requires significant expertise
  - Difficult optimization process
  - Unfamiliar specialized programming models
  - Programming will become even more complicated
- HPC Development Environments
  - Commercial environments currently lack features for:
    - HPC scale
    - Multiple levels of parallelism
    - High degree of communication & synchronization
  - DOE Common Component Architecture
    - http://www.cca-forum.org
    - Scientific Interface Definition Language (SIDL)
    - Babel

HPC Developer Productivity Challenge

Storage access should not be just an afterthought in the HPC development process.

Challenge #13: Integrate OBS concepts into DOE CCA architecture (or any other environment that seems to show promise).

Economy Challenges

Primary Storage Costs

"Current storage controller technology allows for the architecture and deployment of a 1 TB/s system. This system would have a total of 540 storage controller couplets; 4320 host-side connections; 10,800 disk-side connections; 5400 16-slot storage enclosures; and 79,920 disks. Such a storage system would deliver 1 TB/s of bandwidth and cost approximately $90M. A 10TB/s system in 2010 would have a total of 5400 storage controller couplets, 43200 host-side connections, 108,000 disk-side connections, 54,000 16-slot storage enclosures, and 799,200 disks. Such a storage system would deliver 10 TB/s of bandwidth and cost approximately $900M. This cost would include disks and enclosures with associated overhead and storage controllers and associated overhead."

R. Scott Studham, ORNL

U.S. Department of Energy SCIDAC proposal 04/16/06

Secondary Costs

- Power Consumption
- Cooling
- Backup
- Hierarchical Storage Management (HSM)
- Administrative Costs
- Data migration Costs

Economy Challenges

Challenge #14: Manage storage so as to minimize power consumption

Challenge #15: Minimize administrative and operational costs for arbitrary storage classes

Challenge #16: Improve interoperability which lowers customer costs

Adaptability Challenges

What Is Adaptive Supercomputing?

An ADAPTIVE system is one that adjusts its form and behavior to meet varying user and application requirements.

- It is *relatively* easy to optimize and be efficient for one type of application coded with one programming model and running at one scale on one type of hardware and accessing data in one form from one medium.
- HPC environments must:
  - Run applications at many different scales (time and space).
  - Allow developers their choice of programming model
  - Permit combinations of heterogeneous computing resources
  - Permit alternative patterns of access to data of different forms
  - Permit access to alternative storage classes

DARPA HPCS File I/O Scenarios

"The following twelve (12) Scenarios have been reviewed and agreed upon by the HPCS Mission Partners:

1. Single stream with large data blocks operating in half duplex mode
2. Single stream with large data blocks operating in full duplex mode
3. Multiple streams with large data blocks operating in full duplex mode
4. Extreme file creation rates
5. Checkpoint/restart with large I/O requests
6. Checkpoint/restart with small I/O requests
7. Checkpoint/Restart Large File Count Per Directory large I/Os
8. Checkpoint/Restart Large File Count Per Directory small I/Os
9. Walking through directory trees
10. Parallel walking through directory trees
11. Random stat() system call to files in the file system one process
12. Random stat() system call to files in the file system multiple processes

Scenarios 3, 4 and 5 are considered the most important given the state of current technology and the degree of difficulty."

Storage Classes (one customer's take)

- Mission Critical
- Main Operational
  - Shared
  - Dedicated
- Bulk
- Archival
  - Tape
  - MAID

Adaptability Challenges

Challenge #17: Perform well within a large range of scales

- Scale meta-data services
- Scale file/object system services
- Buffering or other services as appropriate

Challenge #18: Perform well for varying application I/O needs

- Highly distributed, massively parallel small transactions
- Concentrated high-bandwidth large-block transfers
- Streamed
- Bursts of distributed large-block accesses
- Bursts of meta-data activity

Challenge #19: Adapt to arbitrary hierarchies and classes of storage components

Summary

- HPC will be very interesting for the next decade
- Technology inflection points for chip technology, processors, memory, storage all at the same time
- HPC storage access will be in crisis without some new technology to bridge the bandwidth gap
- Object-Based Storage may be able to help …

Questions?