

HPCS I/O

Lustre Users Group

28 April 2008

John Carrier
HPCS I/O Lead
Cray, Inc.
carrier@cray.com



Agenda

- DARPA HPCS
 - Cray's HPCS Platform
- HPCS I/O
 - Cray's I/O Goals
- Lustre
 - Sun's proposals for HPCS I/O

DARPA & the HPCS Program

- **DARPA** Defense Advanced Research Projects Agency
 - An R&D incubator for the US Department of Defense
 - Does not directly procure systems, nor has a specific mission or application for which it needs a system designed
- **HPCS** High Productivity Computing Systems
 - A DARPA program created to ensure that US Government agencies continue to have access to the advanced high-performance computing technologies needed to fulfill their missions
 - Includes a number of “Mission Partners” (DOE, DoD, NSF, NSA, NNSA, etc) who work with DARPA to ensure that the systems developed under the program will meet their current and future needs
 - Commercial viability of the resulting system design is a key goal
 - Online resources
 - <http://www.highproductivity.org/>
 - <http://www.darpa.mil/ipto/programs/hpcs/hpcs.asp>

DARPA HPCS Program Award

In November 2006, DARPA awarded Cray and IBM separate \$250 million development contracts under its High Productivity Computing Systems (HPCS) program.

HPCS Goals:

Provide a new generation of economically viable high productivity computing systems for the national security and industrial user community in the 2010 timeframe

- **Performance** (time-to-solution):
speed up critical applications by factors of 10 to 40
- **Programmability** (idea-to-first solution):
reduce cost and time for developing application solutions
- **Portability**:
insulate application software from system specifics
- **Robustness**:
protect applications from hardware faults and system software errors

The result will be greater productivity (not just faster machines)

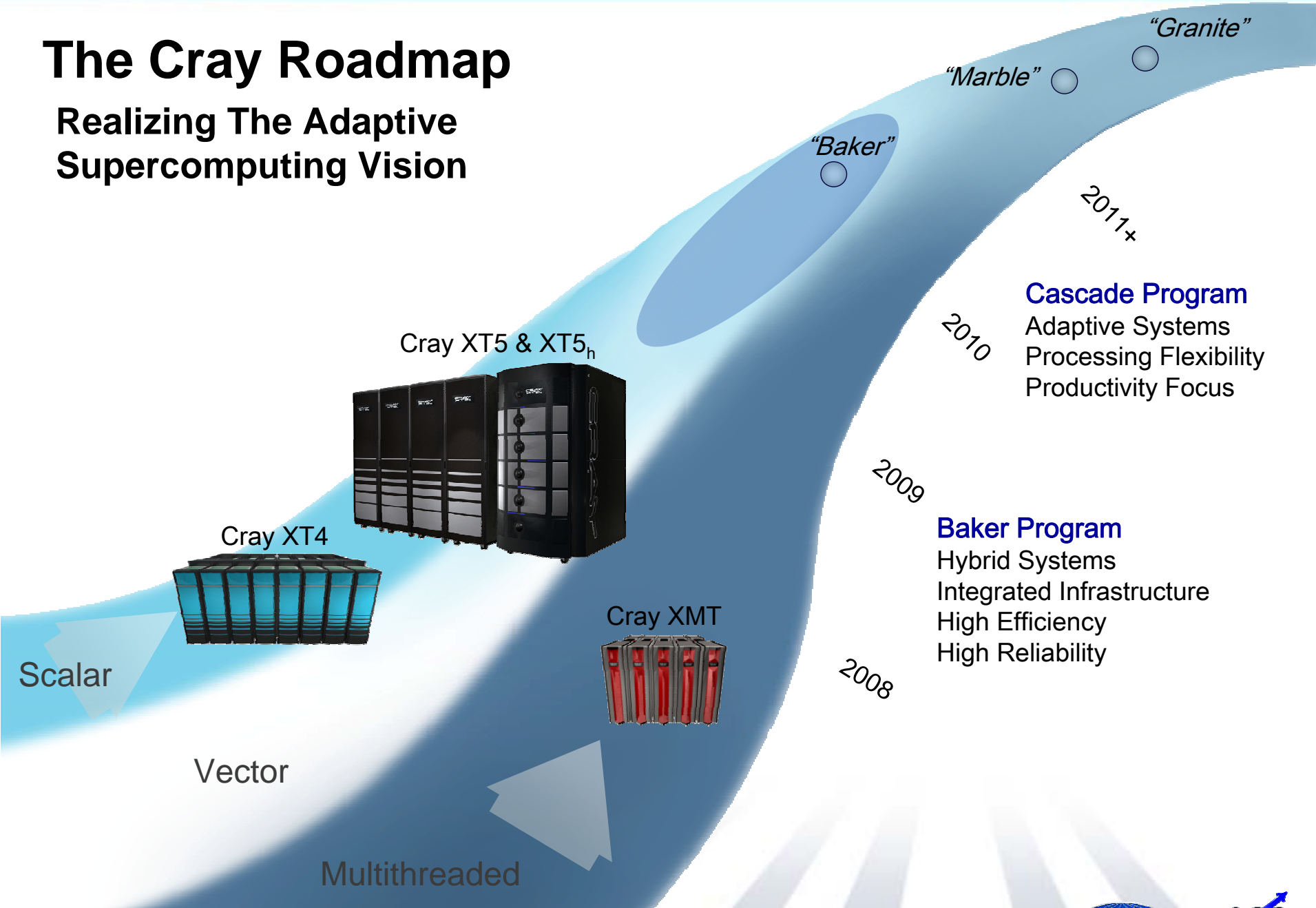
Cray's HPCS Approach

- Ease the development of parallel codes
 - Legacy programming models: MPI, OpenMP, pthreads
 - PGAS programming models: SHMEM, UPC, CAF and Global Arrays
 - Global View programming model: Chapel, GMA
- Provide programming tools to ease debugging and tuning
 - “Expert system” performance tuning tools
 - Data-centric debugging view (relative debugging)
- Design an **adaptive, configurable** machine that can match the attributes of a wide variety of applications:
 - Fast serial performance
 - SIMD data level parallelism (vectorizable)
 - Fine grained MIMD parallelism (threadable)
 - Regular and sparse bandwidth of varying intensities
 - ⇒ Increases performance
 - ⇒ Significantly eases programming
 - ⇒ Makes the machine much more broadly applicable

Adapt the system to the application – not the application to the system

The Cray Roadmap

Realizing The Adaptive Supercomputing Vision



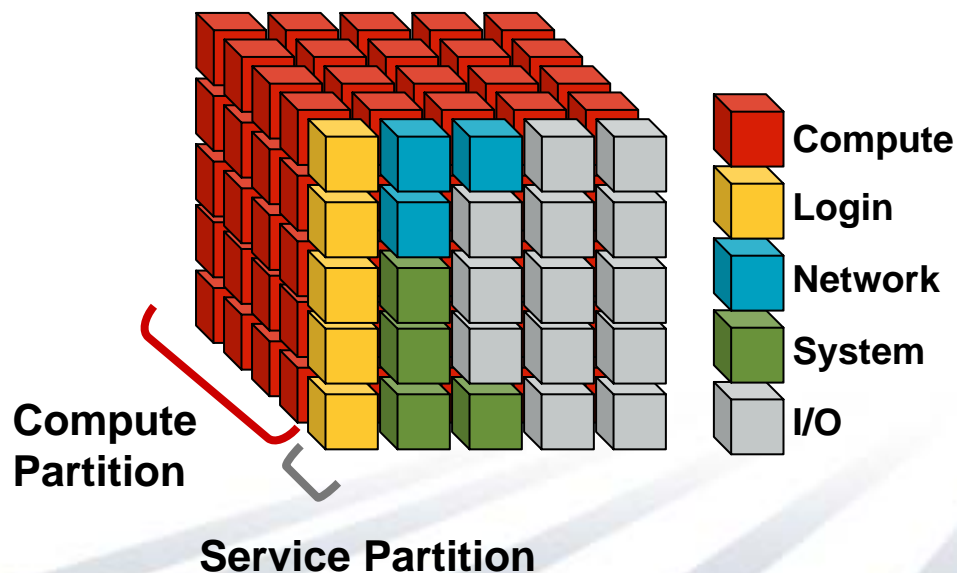
Cray System Architecture

- **Baker & Cascade** build on the existing XT system architecture
 - Massively parallel scalar processing
 - Low latency, low overhead message passing

and add

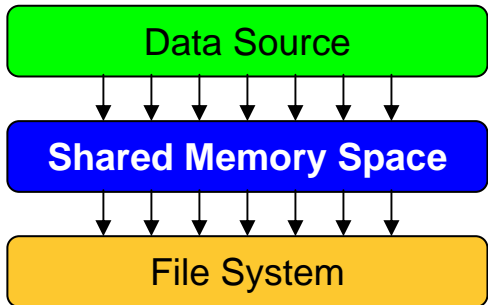
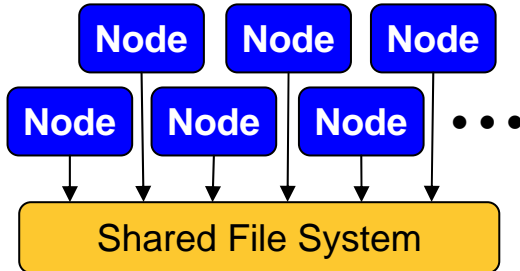
- Globally addressable memory with unified addressing architecture
- Heterogeneous processing across and within node types

- **Compute nodes** support application processes
 - Limited functions delegated to the compute nodes to allow user processes priority
- **Service & I/O (SIO) nodes** provide all services for the system
 - Division of labor – distributing the system services on SIO Nodes in the Service group
 - Built on distributed hardware architecture
 - Supports distributed application profile
 - Provides the opportunity to evolve services and how the services are distributed



Agenda

- DARPA HPCS
 - Cray's HPCS Platform
- HPCS I/O
 - Cray's I/O Goals
- Lustre
 - Sun's proposals for HPCS I/O

Capture Environment	Parallel Environment
<p>System dedicated to moving external data to a large shared memory space where an application will analyze the data streams, create files and then write data to disk</p>  <pre> graph TD DS[Data Source] --> SMS[Shared Memory Space] SMS --> FS[File System] </pre> <p>Diagram illustrating the Capture Environment architecture. A green box labeled "Data Source" has six arrows pointing down to a blue box labeled "Shared Memory Space". The "Shared Memory Space" box has six arrows pointing down to an orange box labeled "File System".</p> <p>Requirements</p> <ul style="list-style-type: none"> ■ Streaming I/O at 30 GB/sec ■ 32K file creates per second 	<p>System with many client nodes connected to a single shared file system or name space. Both the data and metadata operations are important for efficient use.</p>  <pre> graph TD N1[Node] --> SFS[Shared File System] N2[Node] --> SFS N3[Node] --> SFS N4[Node] --> SFS N5[Node] --> SFS </pre> <p>Diagram illustrating the Parallel Environment architecture. Five blue boxes labeled "Node" are arranged in two rows (three on top, two on bottom). Each node has an arrow pointing down to a single orange box labeled "Shared File System". Ellipses (...) follow the bottom row of nodes, indicating more nodes.</p> <p>Requirements</p> <ul style="list-style-type: none"> ■ 30K nodes ■ One trillion files in a single file system ■ 10K metadata operations per second

HPCS I/O Scenarios

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 **Capture Environment**
5. Checkpoint/restart with large I/O requests **Parallel Environment**
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

**Scaling performance, rather than absolute throughput,
is important to all scenarios!**

Mission Partner Requirements for 2010/2011

Focus	Feature	Description
Capacity	Files per directory (max)	10 billion
	File system size (max)	100 PB
	File size range	0 B to 1 PB
	Request size range	1B to multi GB
Performance	<code>fsck</code> recovery time ^[1]	100 hrs for 10 ¹² files
	File create IOPs (max)	40K
	Metadata Ops	Equal to "ls -l" from 10 users
Reliability	Storage resiliency	T10 DIF or equivalent
	File system uptime	99.99%
	Bandwidth availability during rebuild	99.97%

^[1] Because of the 99.99% uptime requirement, the file system cannot be unavailable while performing the file system check.

Mission Partner Suggestions for 2010/2011

Focus	Suggestion
Data Access	Place metadata on a separate volume in order to improve bandwidth.
	Use file versioning as an alternative to full-system backups
	Support O_DIRECT in order to maximize performance of large transfers
	Support extensions to the POSIX I/O API for high end computing ^[2]
File Usage	Long file names and 0-length file lengths for applications that use file metadata as data records
	Support for > 1K files per process and up to 100K open shared files per process

^[2] HPC I/O extensions to the POSIX API proposed by the High End Computing Extensions Working Group (HECEWG) at the OpenGroup : <http://www.opengroup.org/platform/hecewg/>

Cray's HPCS I/O Goals (1)

Requirement

Metadata Performance

- 40,000 file creates per second from a single client
- 10,000 metadata operations per second in aggregate (equal to "ls -lR" from 10 users in separate directory trees)

I/O Performance

- 30 GB/s full-duplex streaming I/O bandwidth from a single client
- 240 GB/s in aggregate^[3] for both file per process and single shared file access

Capacity

- 1 trillion (10^{12}) files per file system:
 - 10 billion files per directory
 - 100 PB maximum file system size
 - 0 to 1 PB file size range
 - 1B to 1 GB I/O request size range
- > 30K client nodes

^[3] expected global disk bandwidth for the Baker System at Oak Ridge National Labs

Cray's HPCS I/O Goals (2)

Requirement

Scalability

- Demonstrate scalable performance using the HPCS I/O Scenarios

Reliability

- End-to-end resiliency equivalent to T10 DIF or better
- Uptime of 99.99%
- No impact of rebuilds on file system performance

Client Accessibility

- O_DIRECT to maximize performance of large transfers
- Open 100,000 shared files per process
- Long file names and 0-length file lengths for applications that use file metadata as data records
- POSIX I/O API extensions proposed at the OpenGroup

Agenda

- DARPA HPCS
 - Cray's HPCS Platform
- HPCS I/O
 - Cray's I/O Goals
- Lustre
 - Sun's proposals for HPCS I/O

HPCS I/O & Lustre

- Cray partnered with Cluster File Systems to use Lustre to meet the I/O requirements in Cray's HPCS proposal to DARPA
 - Founded on the close working relationship Cray had with CFS for our existing XT platforms
 - HPCS extended the relationship for development of advanced features
- Sun has enthusiastically embraced the Cray relationship
 - Cray continues to work closely with the Lustre team at Sun
 - HPCS I/O features already impacting the Lustre roadmap

Lustre Proposal for HPCS I/O

Challenge	Proposal
One Trillion Files per File System	<ul style="list-style-type: none">• Increase Scale of Physical File System• File System Hardening• Clustered Metadata
Capture Node File Creation & Data Rates	<ul style="list-style-type: none">• Metadata Write-back Cache• Multi-threaded, Fully-reentrant Client• File I/O Aggregation

Lustre Server Features for HPCS I/O

Feature	Description
File System Hardening	<ul style="list-style-type: none"> • Use ZFS as the physical file system <ul style="list-style-type: none"> ◦ 128 bit data structures enable ZFS to scale to extreme file system sizes ◦ checksums and metadata mirroring enable ZFS to detect and correct damaged records and remove need to rebuild the file system
File System Performance	<ul style="list-style-type: none"> • Add parity de-clustering and distributed sparing for improved software RAID performance during rebuilds • LNET/LND channel bonding to improve router parallel throughput
Clustered Metadata Servers	<ul style="list-style-type: none"> • Increase the number of servers managing the name space • Scale out the rate of metadata operations • Add active-active failover

Lustre Client Features for HPCS I/O

Feature	Description
Metadata Write-back Cache	<ul style="list-style-type: none"> • Add a fully reentrant, metadata write-back cache to the client for all file operations (create, open, close, read, write, etc) over a mix of small and large files • Enable a single client to generate 30 GB/s and 32K file creates per second
File I/O Aggregation	<ul style="list-style-type: none"> • Improve efficiency of small I/O's and other client file operations by combining I/Os into fewer RPCs to the backend OSTs
File System Hardening	<ul style="list-style-type: none"> • Improve data integrity through end-to-end checksumming from the client to the backend storage

Conclusions

- DARPA is funding development of Cray's High Productivity Computing System
- Cray selected Lustre as its HPCS file system and is working with Sun to meet the challenges created by DARPA's HPCS I/O Requirements and Goals
 - Parallel I/O Challenge 32K clients sharing a single file system containing one trillion files
 - Capture I/O Challenge Process 30 GB/s of streaming data while creating 30K files per second and streaming data out to disk
- Many features developed under this program will appear in Sun's Lustre roadmap

Thank You!