The Cray XE6 Architecture

Stefan Andersson
stefan@cray.com

CRAY THE SUPERCOMPUTER COMPANY

HLRS

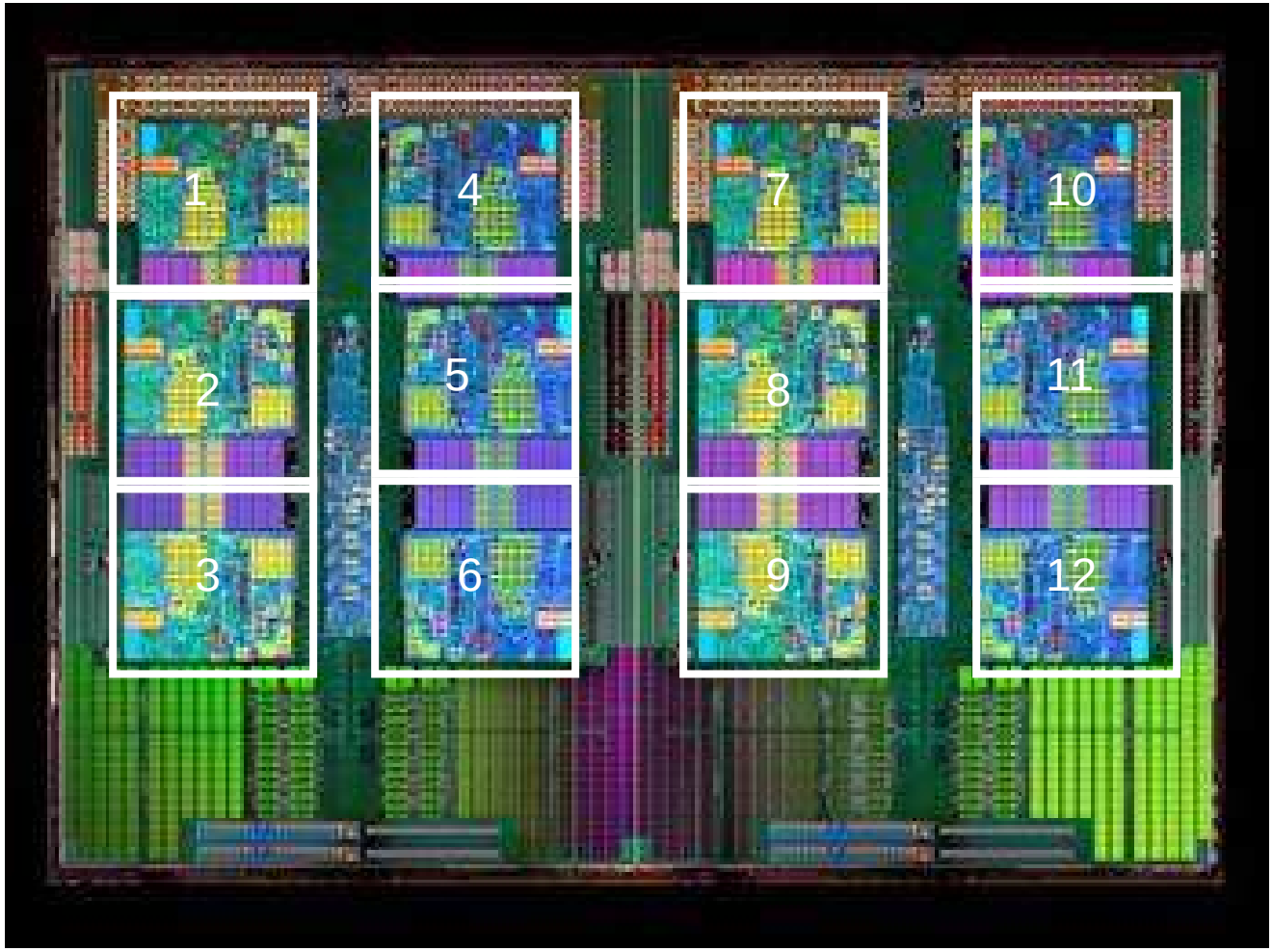
The Cray recipe for a good MPP

1. Select Best Microprocessor
2. Surround it with a balanced or “bandwidth rich” environment
3. “Scale” the System
   - Eliminate Operating System Interference (OS Jitter)
   - Design in Reliability and Resiliency
   - Provide Scalable System Management
   - Provide Scalable I/O
   - Provide Scalable Programming and Performance Tools
   - System Service Life (provide an upgrade path)

x86 64-bit Architecture Evolution

	2003	2005	2007	2008	2009	2010
	AMD Opteron™	AMD Opteron™	"Barcelona"	"Shanghai"	"Istanbul"	"Magny-Cours"
Mfg. Process	130nm SOI	90nm SOI	65nm SOI	45nm SOI	45nm SOI	45nm SOI
CPU Core	K8	K8	Greyhound	Greyhound+	Greyhound+	Greyhound+
L2/L3	1MB/0	1MB/0	512kB/2MB	512kB/6MB	512kB/6MB	512kB/12MB
Hyper Transport™ Technology	3x 1.6GT/.s	3x 1.6GT/.s	3x 2GT/s	3x 4.0GT/s	3x 4.8GT/s	4x 6.4GT/s
Memory	2x DDR1 300	2x DDR1 400	2x DDR2 667	2x DDR2 800	2x DDR2 800	4x DDR3 1333

AMD Opteron™ 6000 Series Processors



12 cores
1.7-2.2Ghz
105.6Gflops

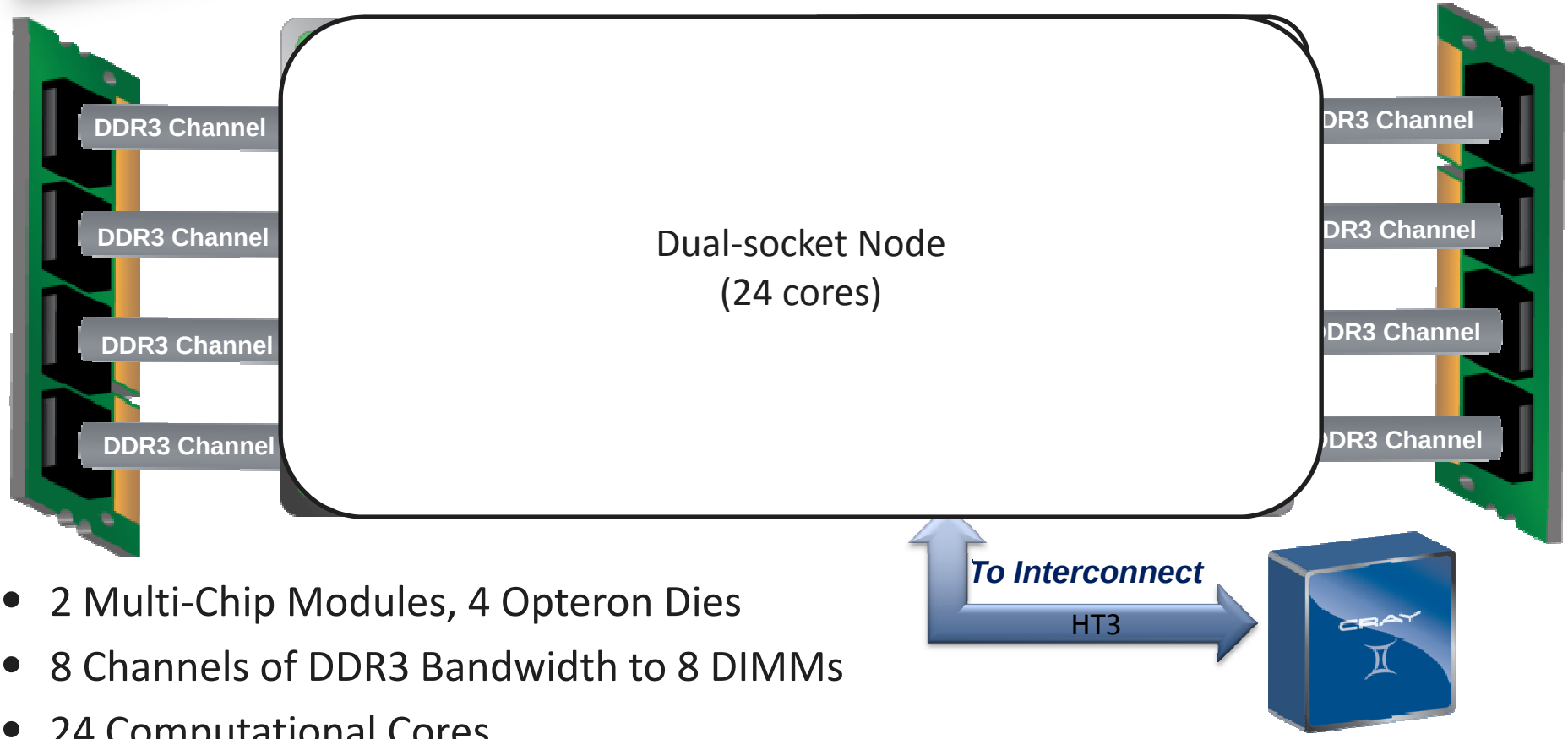
8 cores
1.8-2.4Ghz
76.8Gflops

Power (ACP)
80Watts

Stream
27.5GB/s

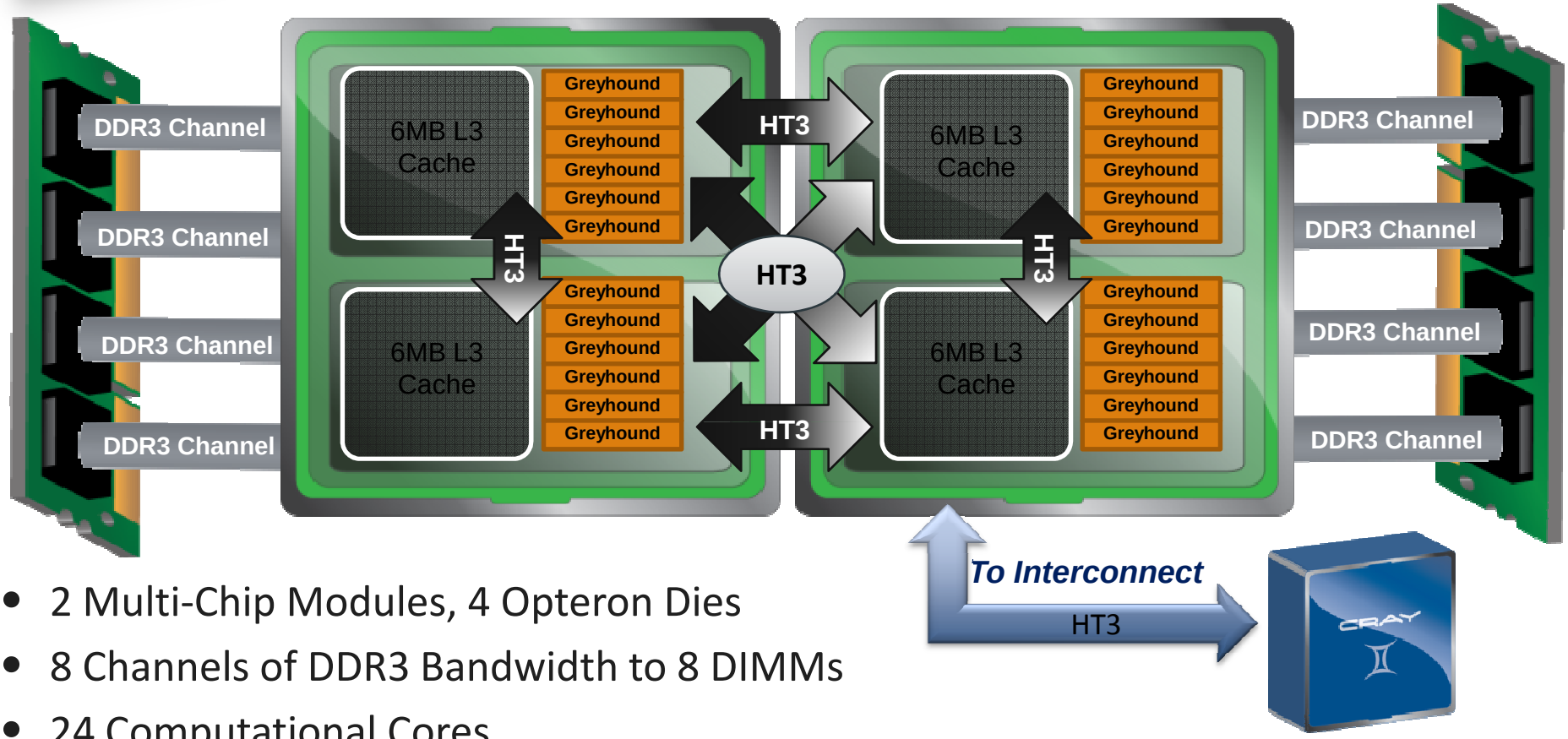
Cache
12x 64KB L1
12x 512KB L2
12MB L3

Cray XE6 Node Details: 24-core Magny-Cours



- 2 Multi-Chip Modules, 4 Opteron Dies
- 8 Channels of DDR3 Bandwidth to 8 DIMMs
- 24 Computational Cores
- Dies are fully connected with HT3
- Snoop Filter Feature Allows 4 Die SMP to scale well

Cray XE6 Node Details: 24-core Magny-Cours



- 2 Multi-Chip Modules, 4 Opteron Dies
- 8 Channels of DDR3 Bandwidth to 8 DIMMs
- 24 Computational Cores
- Dies are fully connected with HT3
- Snoop Filter Feature Allows the 4-Die-SMP to scale well

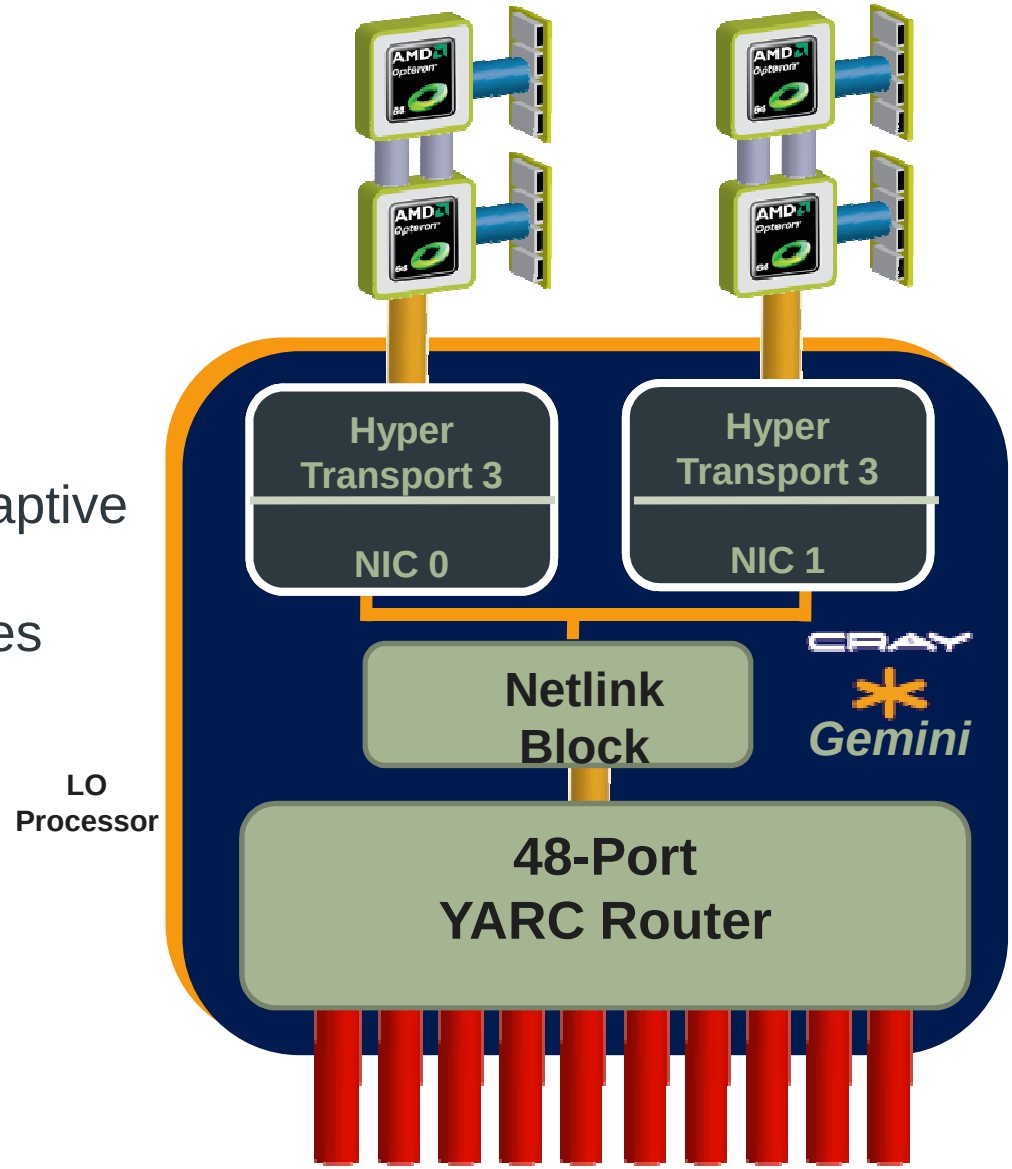
AMD Magny-Cours Data Caches (G34 MCM)

Conway et al."Cache Hierarchy and Memory Subsystem of the AMD Opteron Processor," IEEE Micro, pp. 16-29, March/April, 2010

- L1 Data Cache
  - 64KB, 64B cacheline, 2-way associative
  - Load-to-use latency: 3 clock cycles
- L2 Cache
  - 512KB, 64B cacheline , 16-way associative
  - Load-to-use latency: 12 clock cycles
  - Victim / Copy-Back from L1
  - Hits are invalidated from L2 and placed into L1
- L3 Cache, shared
  - 6 MB per die, 64B cacheline, 16-way associative
  - Victim / Copy-Back from L2
  - Hits can be removed or stay on L3 if needed by other threads

Cray Gemini ASIC

- Supports 2 Nodes per ASIC
- 3D Torus network
  - XT5/XT6 systems field upgradable
- Scales to over 100,000 network endpoints
  - Link Level Reliability and Adaptive Routing
  - Advanced Resiliency Features
- Advanced features
  - MPI – millions of messages / second
  - One-sided MPI
  - UPC, Coarray FORTRAN, Shmem, Global Arrays
  - Atomic memory operations

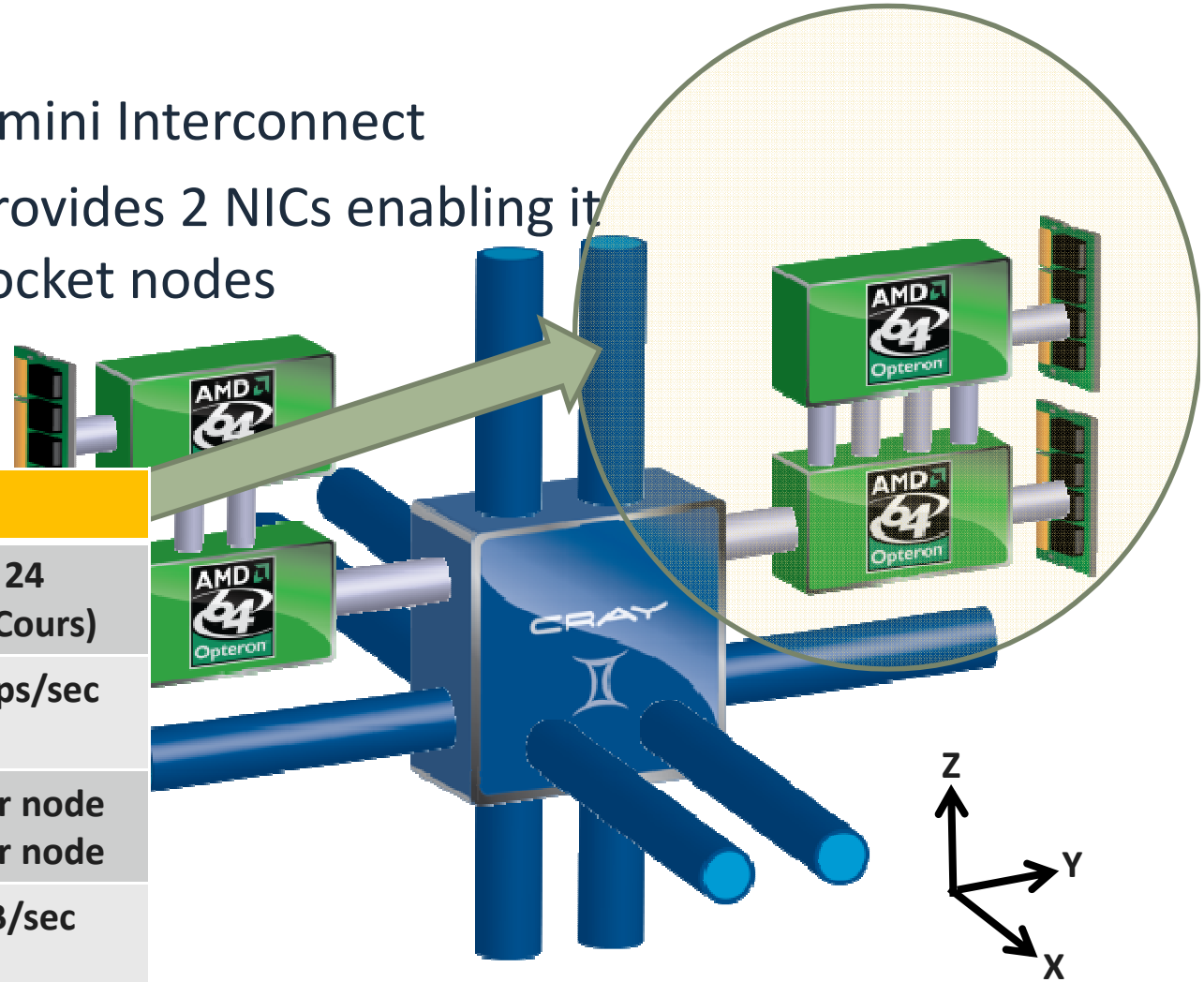


CRAY THE SUPERCOMPUTER COMPANY

HLRS

Cray XE6 Compute Node

- Built around the Gemini Interconnect
- Each Gemini ASIC provides 2 NICs enabling it to connect 2 dual-socket nodes

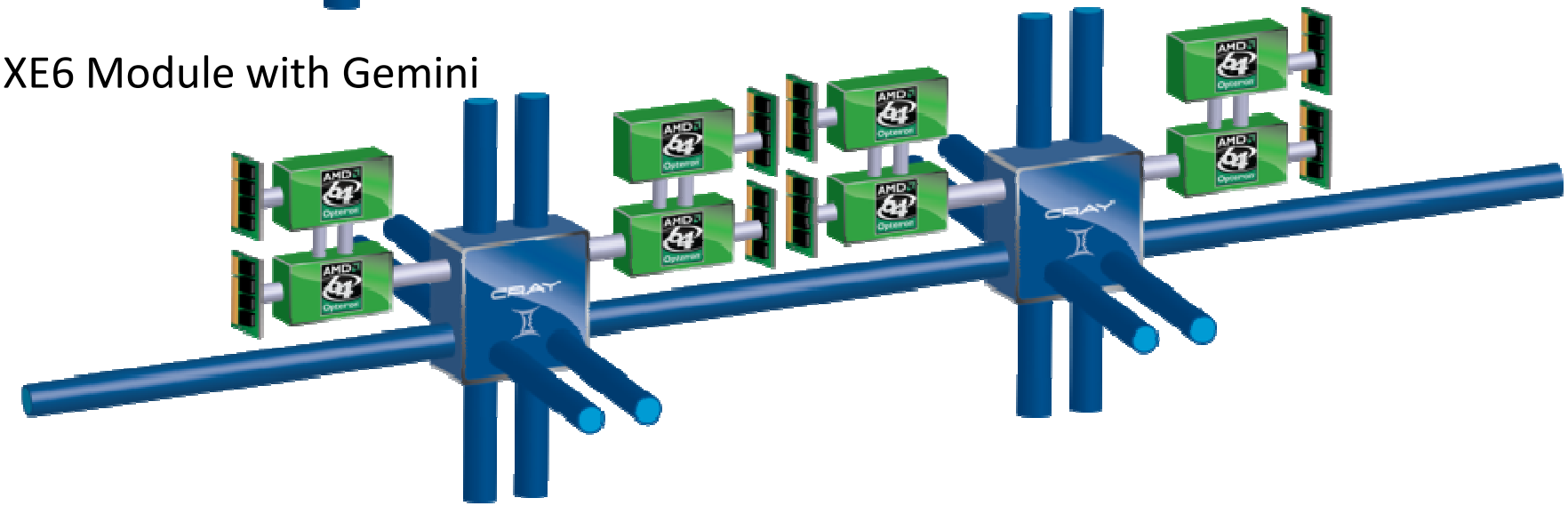


Node Characteristics	
Number of Cores	16 or 24 (Magny Cours)
Peak Performance MC-12 (2.2)	211 Gflops/sec
Memory Size	32 GB per node 64 GB per node
Memory Bandwidth (Peak)	83.5 GB/sec

Gemini vs SeaStar – Topology

XT6 Module with SeaStar

XE6 Module with Gemini



Cray Network Evolution

SeaStar (Cray XT)

- Built for scalability to 250K+ cores
- Very effective routing and low contention switch

Gemini (Cray XE)

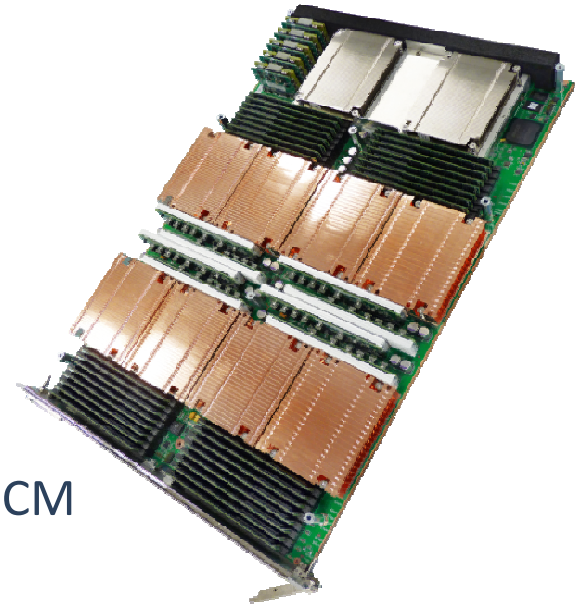
- 100x improvement in message throughput
- 3x improvement in latency
- PGAS Support, Global Address Space
- Scalability to 1M+ cores

Aries

- DON’T ask me about it

The Cray XE6 packaging

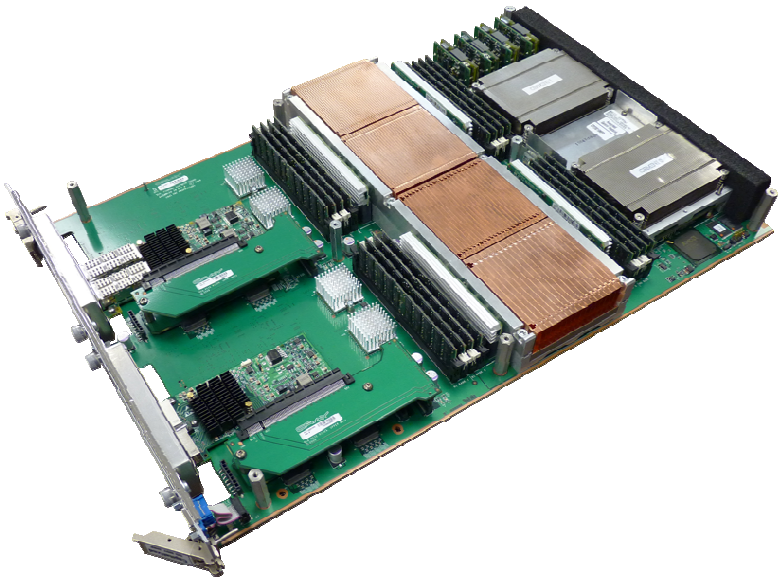
Compute nodes on the XE6



- Configuration
  - 4 compute nodes per compute blade
  - Each compute node has 2 Opteron sockets
  - Each socket hosts a 8 or 12-core Magny-Cours MCM for a total of 64 or 96 compute cores
  - 32 DDR3 Memory DIMMS + 32 DDR3 Memory channels
  - 2 Gemini ASICs
  - L0 Blade management processor
- Runs Cray Linux Environment (CLE)
  - Linux-based operating system
  - designed to run large, complex applications and scale efficiently to hundreds of thousands of processor cores

Service nodes on the XE

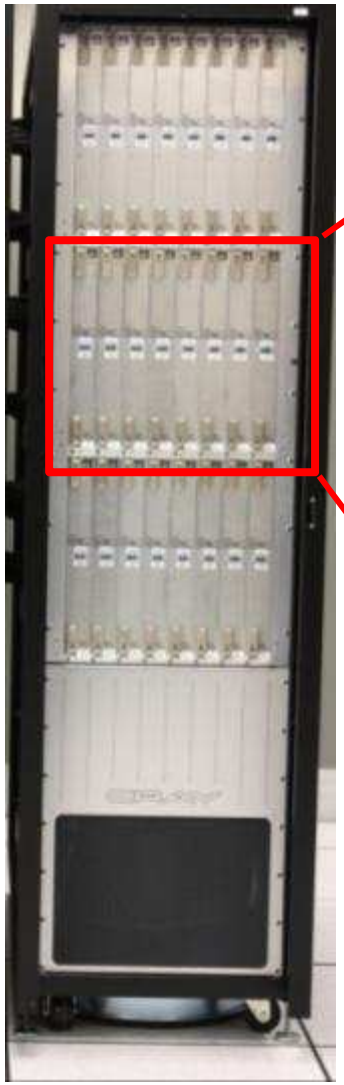
- Overview
  - Run full Linux (SuSe SLES 11)
  - 4 nodes per service blade
- Boot node
  - first XE6 node to be booted: boots all other components
- IO nodes
  - Run Lustre processes (OST, MDT)
- SDB node
  - hosts MySQL database
  - processors, allocation, accounting, PBS information
- Login nodes
  - User login and code preparation activities: compile, launch
  - Partition allocation: ALPS (Application Level Placement Scheduler)

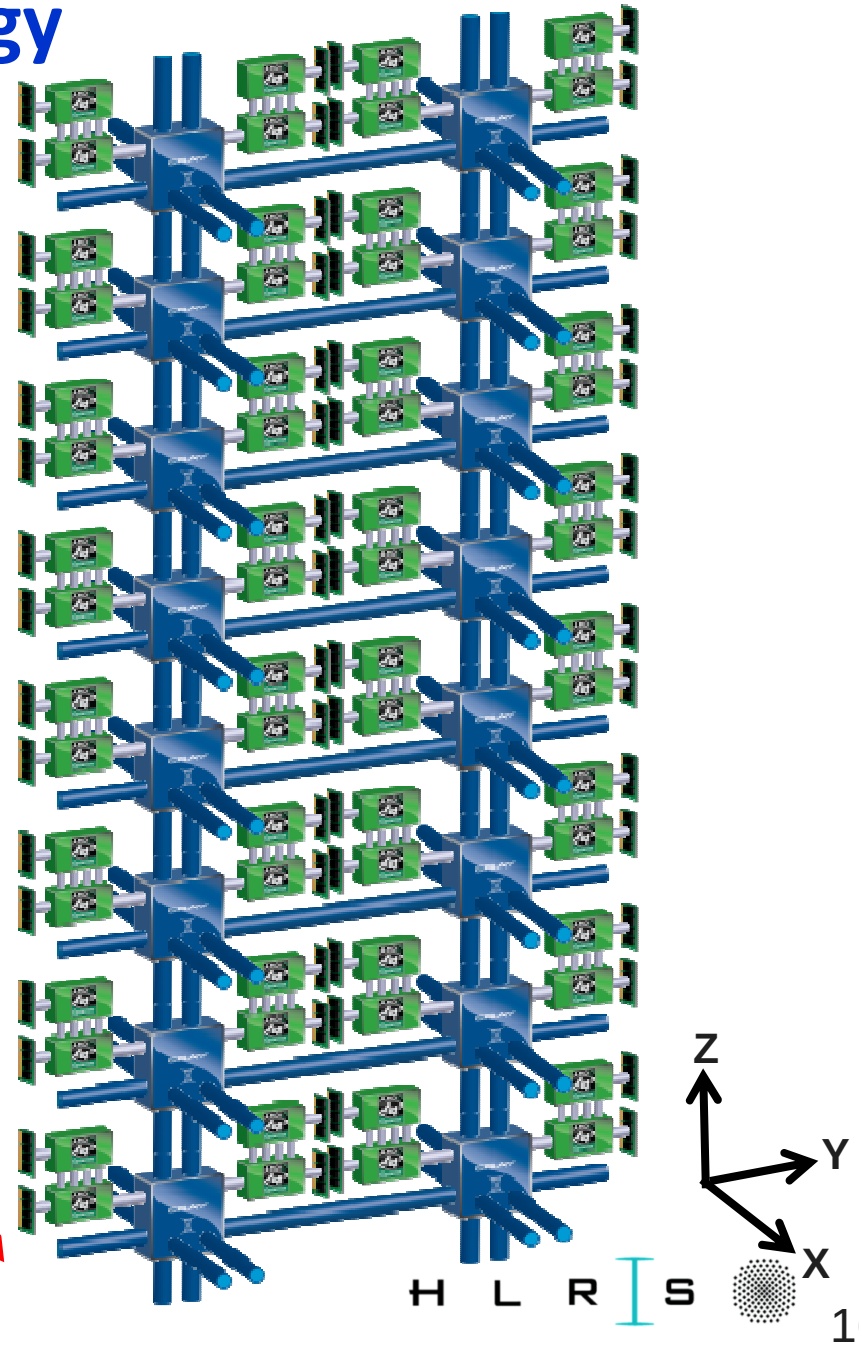


XE6 configuration details

- A XE6 cabinet contains 3 cages
- A cage contains 8 blades
- A compute blade contains
  - 8 sockets
  - 2 Gemini interconnects
  - Memory
  - L0 controller
  - VRMs
  - **No** moving parts
- One blower at the bottom

Cray XE6 Chassis Topology





XE6 Topology Tutorial

Class 0 Topology (HLRS phase0)

For a system of up to 3 cabinets, with 1 to 9 chassis, the topology is a full 3D Torus of size 3N x 4 x 8, where N is the number of chassis.

Class 1 Topology

For a system that is a single row of 4 or more and up to 16 cabinets, the topology is a full 3D Torus of size N x 12 x 8, where N is the number of cabinets.

Class 2 Topology (HLRS phase1)

For systems comprised of two rows, the topology is a full 3D torus of size N x 12 x 16, where N is the number of cabinets in a row (total 2 x N cabinets). This class covers configurations from 16 (N=8) to 48 (N=24) cabinets.

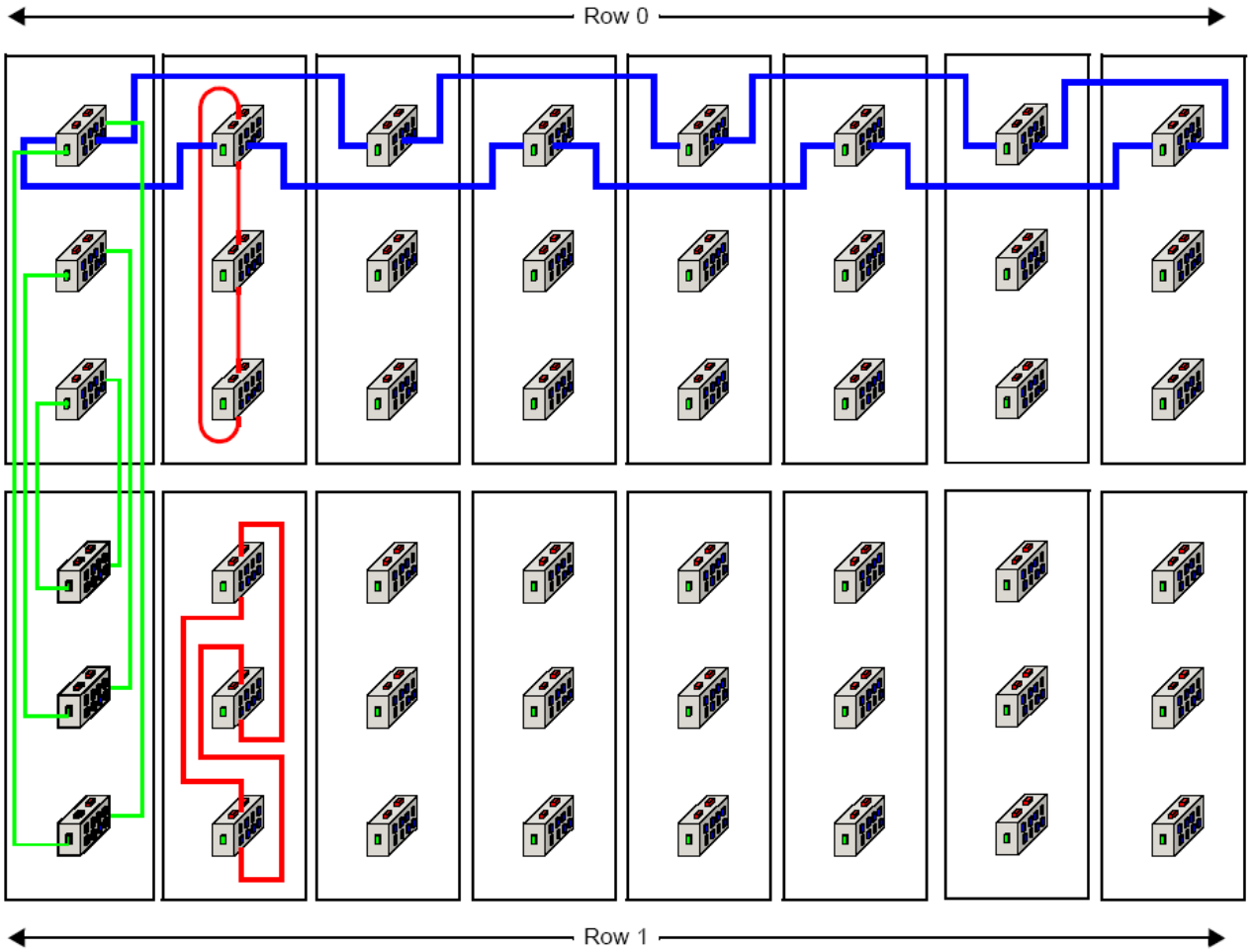
Class 3 Topology (ORNL JaguarPF)

For larger, multi-row systems, with an even number of rows, the topology is a full 3D torus of size N x (4 x number of rows) x 24. This class covers configurations from 48 (4 rows, 12 cabinets per row) to 576 (12 rows, 48 cabinets per row) cabinets.

HLRS Phase0 system

- Cabinets: 1 cabinet
- Topology: 3 x 4 x 8
- Compute
  - 21 compute blades: 84 nodes, 1344 cores @ 2.0 GHz
  - Memory 32 GB/node, 2 GB/core
  - Peak/node: 128 Gflop/s
  - Peak total: 10.75 Tflop/s
- Service
  - 3 service blades: 12 nodes, hex-core @ 2.2 GHz
  - Memory 16 GB/node

Class 2 example : 16 Cabinets, 8 x 12 x 16



X

Y

Z

16 Cabinets
2 Rows of 8
(8 x 12 x 16)

Class 3: ORNL JaguarPF

Peak Performance	**2.33 Petaflops**
System Memory	300 Terabytes
Disk Space	10.7 Petabytes
Interconnect	3D Torus 25x32x24
Processor Cores	224,256

Nodes description: xtprocadmin -A

```
   NID    (HEX)    NODENAME     TYPE ARCH        OS CORES AVAILMEM   PAGESZ CLOCKMHZ
     0      0x0  c0-0c0s0n0  service   xt (service)     2     8000     4096     2600
     3      0x3  c0-0c0s0n3  service   xt (service)     2     8000     4096     2600
     4      0x4  c0-0c0s1n0  service   xt (service)     2     8000     4096     2600
     7      0x7  c0-0c0s1n3  service   xt (service)     2     8000     4096     2600
     8      0x8  c0-0c0s2n0  service   xt (service)     2     8000     4096     2600
    11      0xb  c0-0c0s2n3  service   xt (service)     2     8000     4096     2600
    12      0xc  c0-0c0s3n0  service   xt (service)     2     8000     4096     2600
    15      0xf  c0-0c0s3n3  service   xt (service)     2     8000     4096     2600…
    16     0x10  c0-0c0s4n0  compute   xt       CNL     8    16000     4096     2400
    17     0x11  c0-0c0s4n1  compute   xt       CNL     8    16000     4096     2400
    18     0x12  c0-0c0s4n2  compute   xt       CNL     8    16000     4096     2400
    19     0x13  c0-0c0s4n3  compute   xt       CNL     8    16000     4096     2400
……

  2520    0x9d8  c9-1c2s6n0  compute   xt       CNL     8    16000     4096     2400
  2521    0x9d9  c9-1c2s6n1  compute   xt       CNL     8    16000     4096     2400
  2522    0x9da  c9-1c2s6n2  compute   xt       CNL     8    16000     4096     2400
  2523    0x9db  c9-1c2s6n3  compute   xt       CNL     8    16000     4096     2400
  2524    0x9dc  c9-1c2s7n0  compute   xt       CNL     8    16000     4096     2400
  2525    0x9dd  c9-1c2s7n1  compute   xt       CNL     8    16000     4096     2400
  2526    0x9de  c9-1c2s7n2  compute   xt       CNL     8    16000     4096     2400
  2527    0x9df  c9-1c2s7n3  compute   xt       CNL     8    16000     4096     2400
```

xtnodestat

cabinet

cage

```
     C0-0     C0-1     C1-0     C1-1     C2-0     C2-1     C3-0     C3-1     C4-0     C4-1
  n3 aaaaaaaa aaaaaaaa SaaaaSaa aaaSaaaS aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n2 aaaaaaaa aaaaaaaa  aaaa aa aaa aaa  aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n1 aaaaaaaa aaaaaaaa  aaaa aa aaa aaa  aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
c2n0 aaaaaaaa aaaaaaaa SaaaaSaa aaaSaaaS aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n3 aaaaaaaa SaaaSaaa SaaaSaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n2 aaaaaaaa  aaa aaa  aaa aaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n1 aaaaaaaa  aaa aaa  aaa aaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
c1n0 aaaaaaaa SaaaSaaa SaaaSaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n3 SSSSaaaa aaSaaaSa SSSaaaaa SSaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n2     aaaa aa aaa a    aaaaa   aaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n1     aaaa aa aaa a    aaaaa   aaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
c0n0 SSSSaaaa aaSaaaSa SSSaaaaa SSaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567

     C5-0     C5-1     C6-0     C6-1     C7-0     C7-1     C8-0     C8-1     C9-0     C9-1
  n3 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n2 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n1 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
c2n0 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n3 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n2 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n1 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
c1n0 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n3 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n2 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
  n1 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
c0n0 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567
```

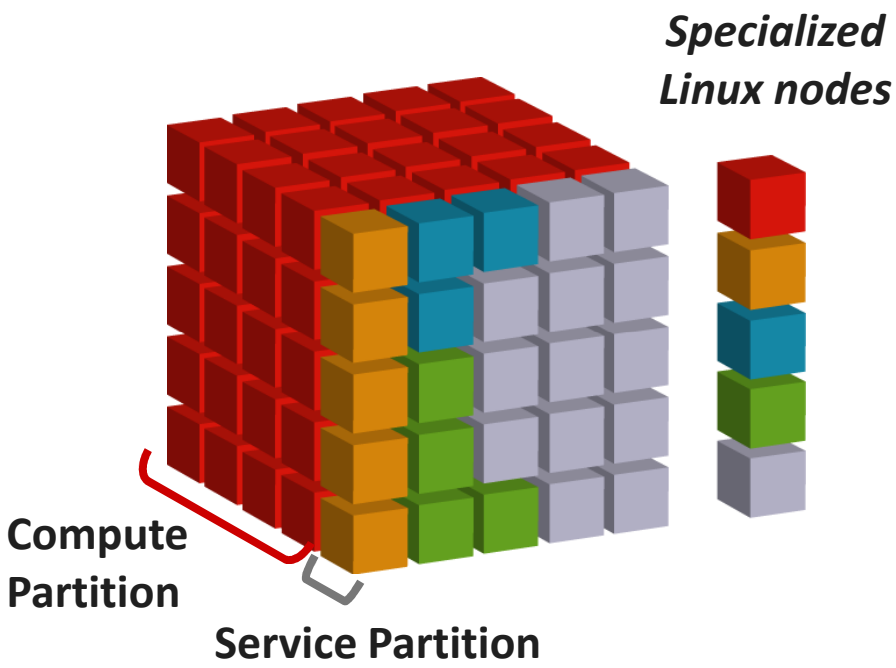
Blades / slots

Scalable Software Architecture

CRAY
THE SUPERCOMPUTER COMPANY

HLRS

Scalable Software Architecture: CLE



Microkernel on Compute nodes, full featured Linux on Service nodes.

Service PEs specialize by function

Software Architecture eliminates OS "Jitter"

Software Architecture enables reproducible run times

CLE3, An Adaptive Linux OS designed specifically for HPC

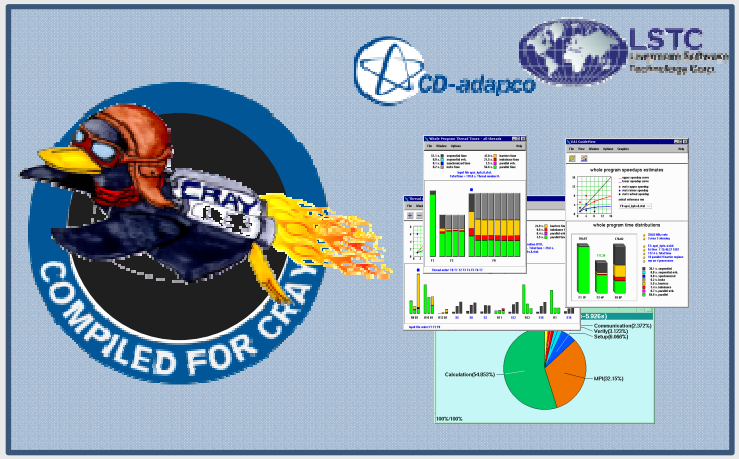
CRAY LINUX ENVIRONMENT CLE3

ESM – *Extreme Scalability Mode*

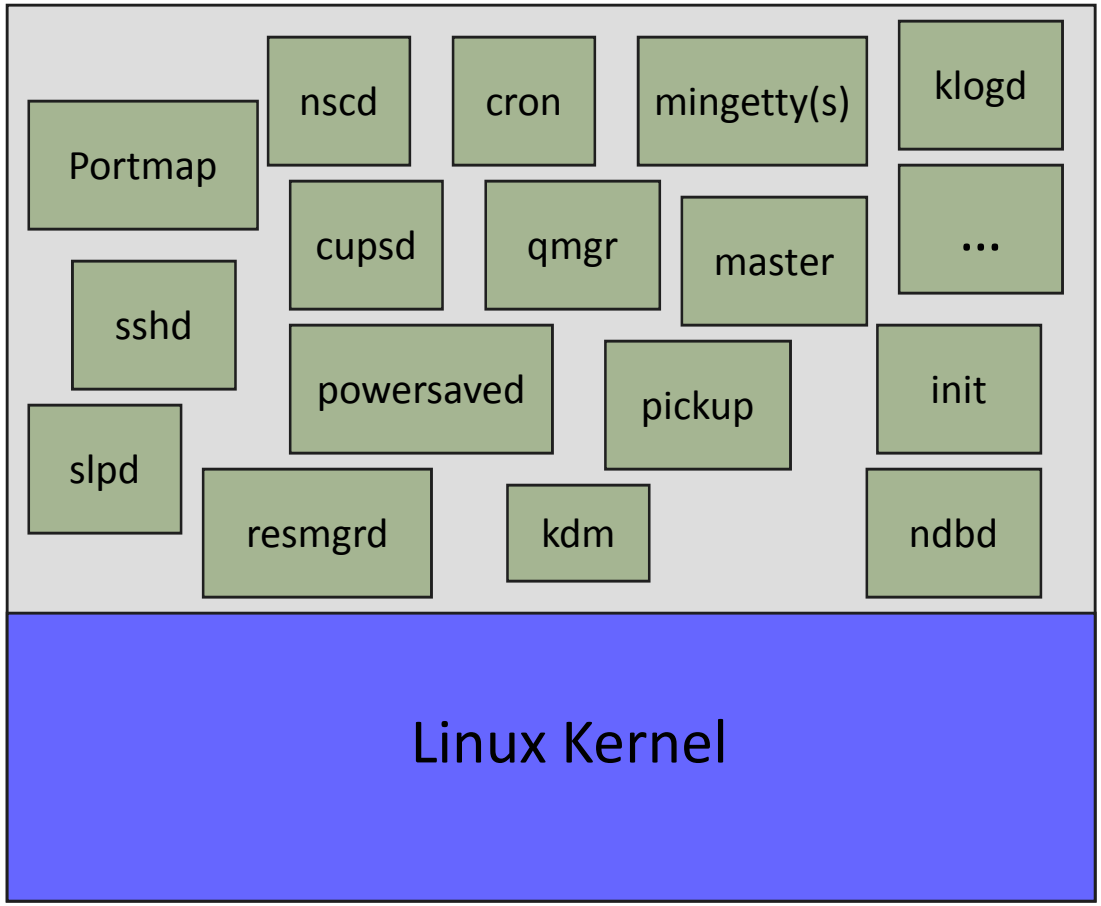
- No compromise *scalability*
- Low-Noise Kernel for scalability
- Native Comm. & Optimized MPI
- Application-specific performance tuning and scaling

CCM –*Cluster Compatibility Mode*

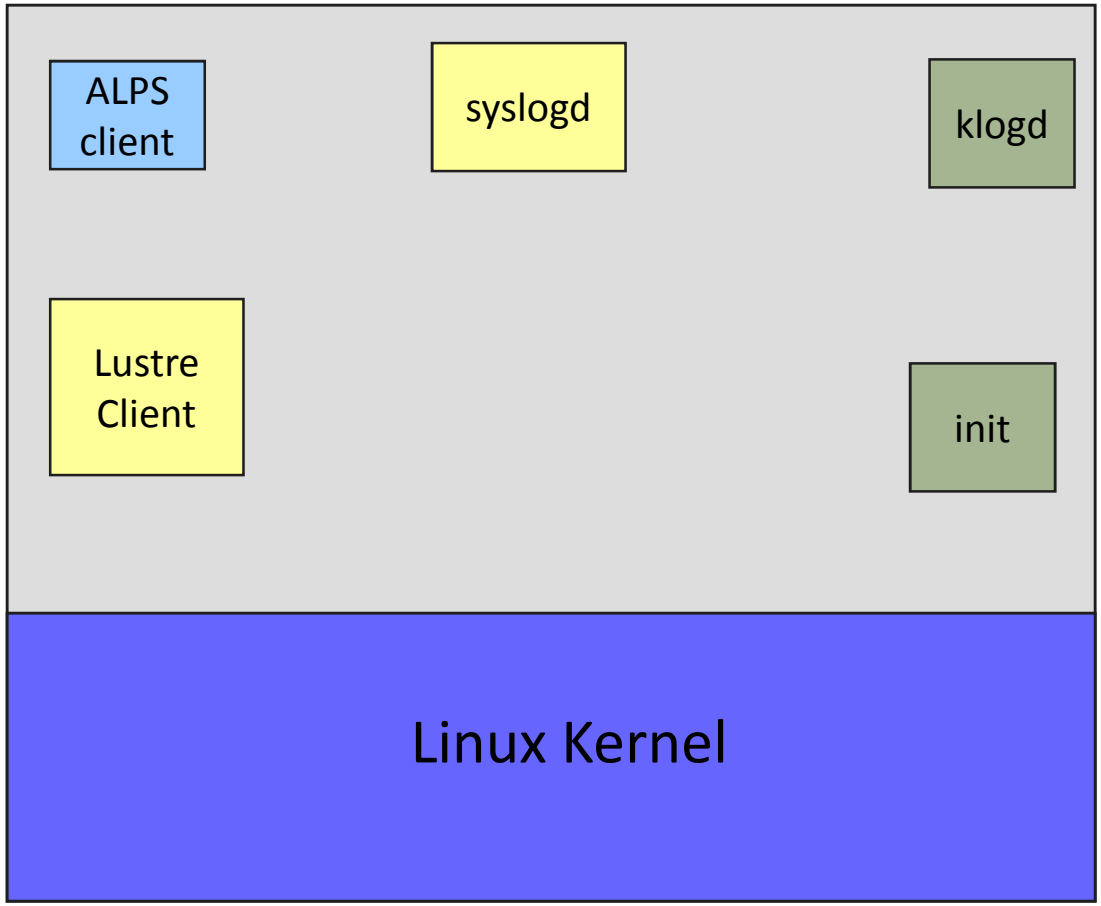
- No compromise *compatibility*
- Fully standard x86/Linux
- Standardized Communication Layer
- Out-of-the-box ISV Installation
- ISV applications simply install and run



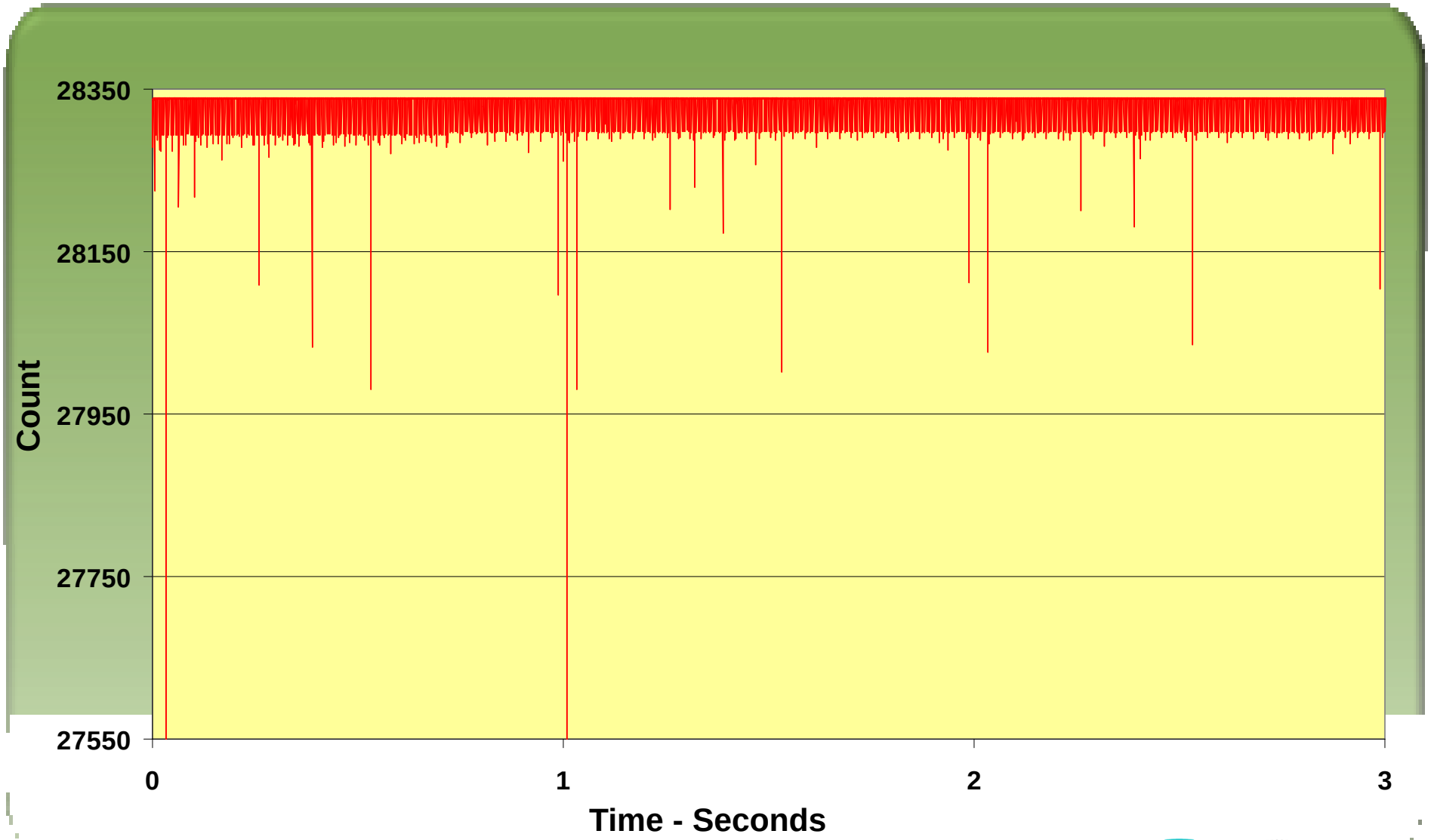
Trimming OS – *Standard Linux Server*



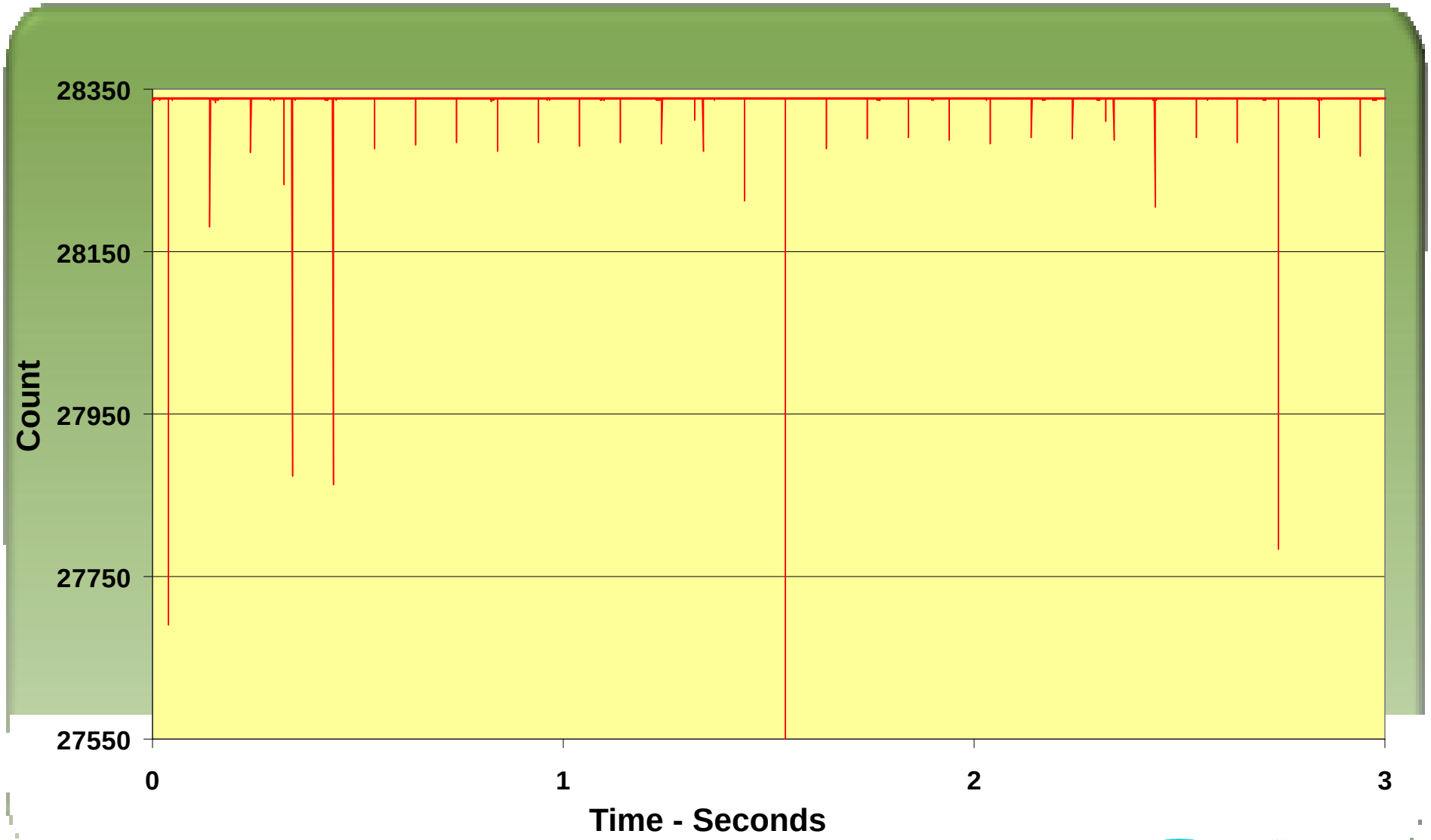
Linux on a Diet – *CNL*



FTQ Plot of Stock SuSE (most daemons removed)



FTQ plot of CNL



DSL : Dynamic shared libraries

- Benefit: root file system environment available to applications
- Shared root from SIO nodes will be available on compute nodes
- Standard libraries / tools will be in the standard places
- Able to deliver customer-provided root file system to compute nodes
- Programming environment supports static and dynamic linking
- Performance impact negligible, due to scalable implementation

Cluster Compatibility Mode: Overview

- Provides the runtime environment on compute nodes expected by ISV applications
- Associated with specific batch queues
- Dynamically allocates and configures compute nodes at job start
  - Nodes are not permanently dedicated to CCM
  - Any compute node can be used
  - Allocated like any other batch job (on demand)

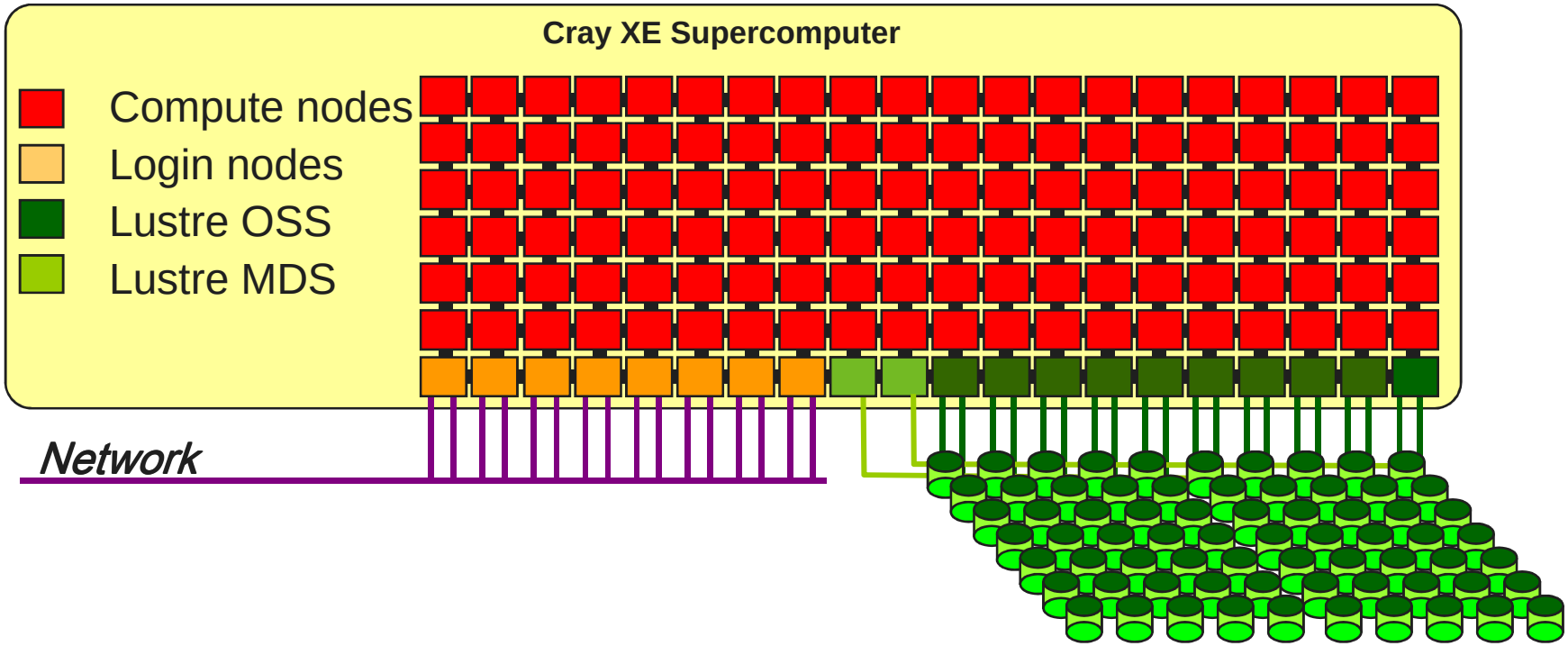
    Third party MPI runs over TCP/IP over HSN
- Supports standard services: ssh, rsh, nscd, ldap
- Complete root file system on the compute nodes
  - built on top of the Dynamic Shared Libraries (DSL) environment

Under CCM, everything the application can "see" is identical to a standard Linux cluster: Linux OS, x86 processor, and MPI

Cray XE I/O architecture

- All I/O is offloaded to service nodes
- Lustre
  - High performance parallel I/O file system
  - Direct data transfer between compute nodes and files
- DVS
  - Virtualization service
  - Allows compute nodes to access NFS mounted on service node
  - Applications must execute on file systems mounted on compute nodes
- No local disks
- /tmp is a MEMORY file system, on each login node

The Storage Environment

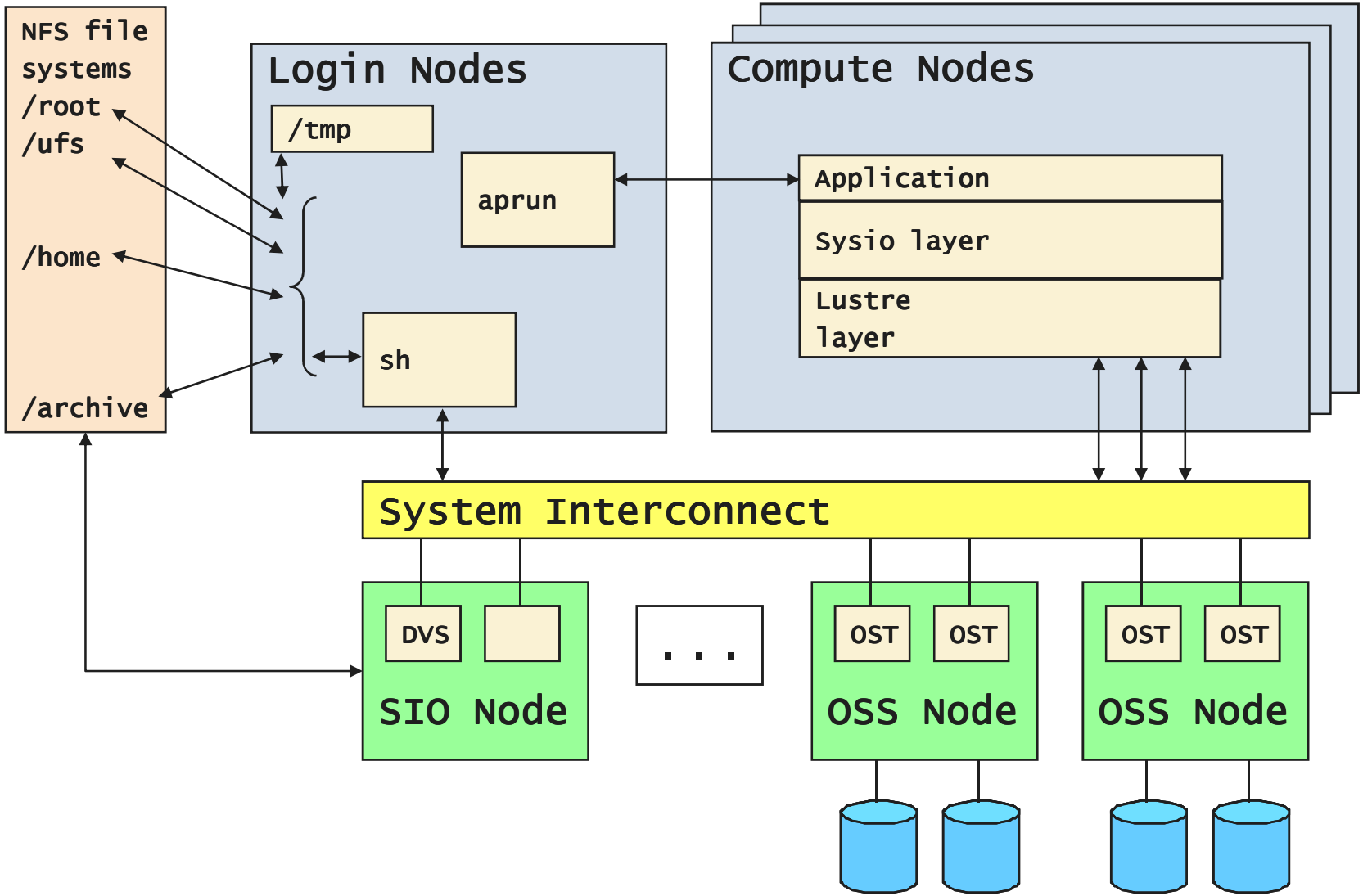


Lustre

high performance

parallel filesystem

Cray XE I/O architecture



Lustre

- A scalable cluster file system for Linux
  - Developed by Cluster File Systems, Inc.
  - Name derives from “Linux Cluster”
  - The Lustre file system consists of software subsystems, storage, and an associated network
- **MDS** – metadata server
  - Handles information about files and directories
- **OSS** – Object Storage Server
  - The hardware entity
  - The server node
  - Support multiple OSTs
- **OST** – Object Storage Target
  - The software entity
  - This is the software interface to the backend volume

Gemini Software

- Cray MPI uses MPICH2 distribution from Argonne
  - CH3 device Nemesis: multi-method device with a highly optimized shared memory sub-method
- MPI device for Gemini based on
  - User level Gemini Network Interface (uGNI)
  - Distributed Memory Applications (DMAPP) library
- FMA (Fast Memory Access)
  - In general used for small transfers
  - FMA transfers are lower latency
- BTE (Block Transfer Engine)
  - BTE transfers take longer to start but can transfer large amount of data without CPU involvement

ALPS

- ALPS (Application Level Placement Scheduler)
  - Handles the execution of applications on compute nodes
  - aprun is ALPS application launcher
  - The algorithm used by ALPS to allocate compute nodes for the applications is configurable at ALPS startup
- Compute Node allocation
  - Nodes are allocated in configurable topology aware sequence, according to 3D torus dimensions
  - Current setup will allocate 2x2x2 ‚cubes‘

Short list of commands

- xtprocadmin
  - List the node configuration
- xtnodestat
  - List the applications and the node partitions
- apstat
  - Status of applications
- cnselect
  - Provides a list of nodes querying the sdb database

```
hpcander@xe601:~> cnselect -L clockmhz
2000
hpcander@xe601:~> cnselect -L availmem
32000
hpcander@xe601:~> cnselect –c
84
```


Thank you !