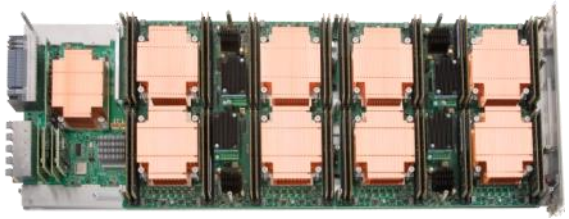
Cray XC40 Architecture Overview

Ilias Katsardis
ikatsardis@cray.com



Agenda

- XC40 – The basics

- Packaging
- Board-level
- Processor
- Network
- Cooling
- Lustre Storage

Cray's recipe for a good supercomputer

- **Select best microprocessor**
  - Function of time
- **Surround it with a bandwidth-rich environment**
  - Interconnection network
  - Local memory
- **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

Nodes: The building blocks

The Cray XC40 is a Massively Parallel Processor (MPP) supercomputer design. It is therefore built from many thousands of individual nodes.

There are two basic types of nodes in any Cray XC40:

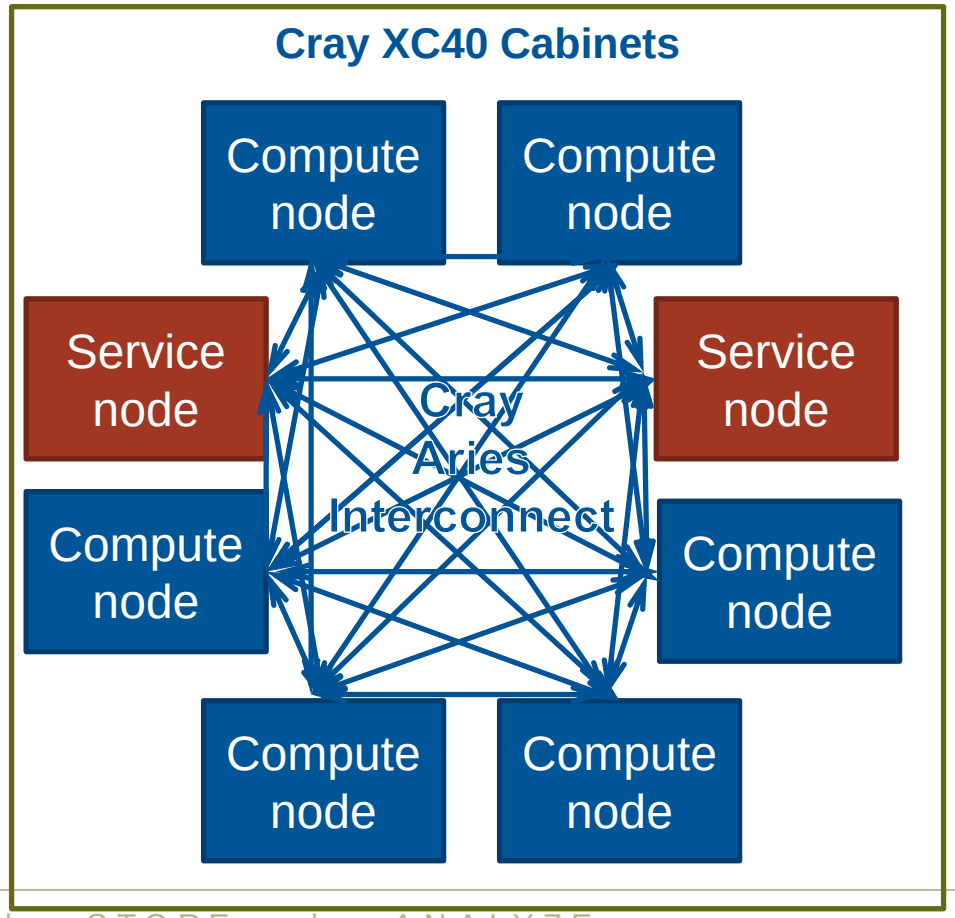
- **Compute nodes**
  - These only do user computation and are always referred to as “Compute nodes”
- **Service nodes**
  - These provide all the additional services required for the system to function, and are given additional names depending on their individual task:
    - Login nodes – allow users to log in and perform interactive tasks
    - PBS Mom nodes – run and managing PBS batch scripts
    - Service Database node (SDB) – holds system configuration information
    - LNET Routers - connect to the external filesystem.

There are usually many more compute than service nodes

Connecting nodes together: Aries

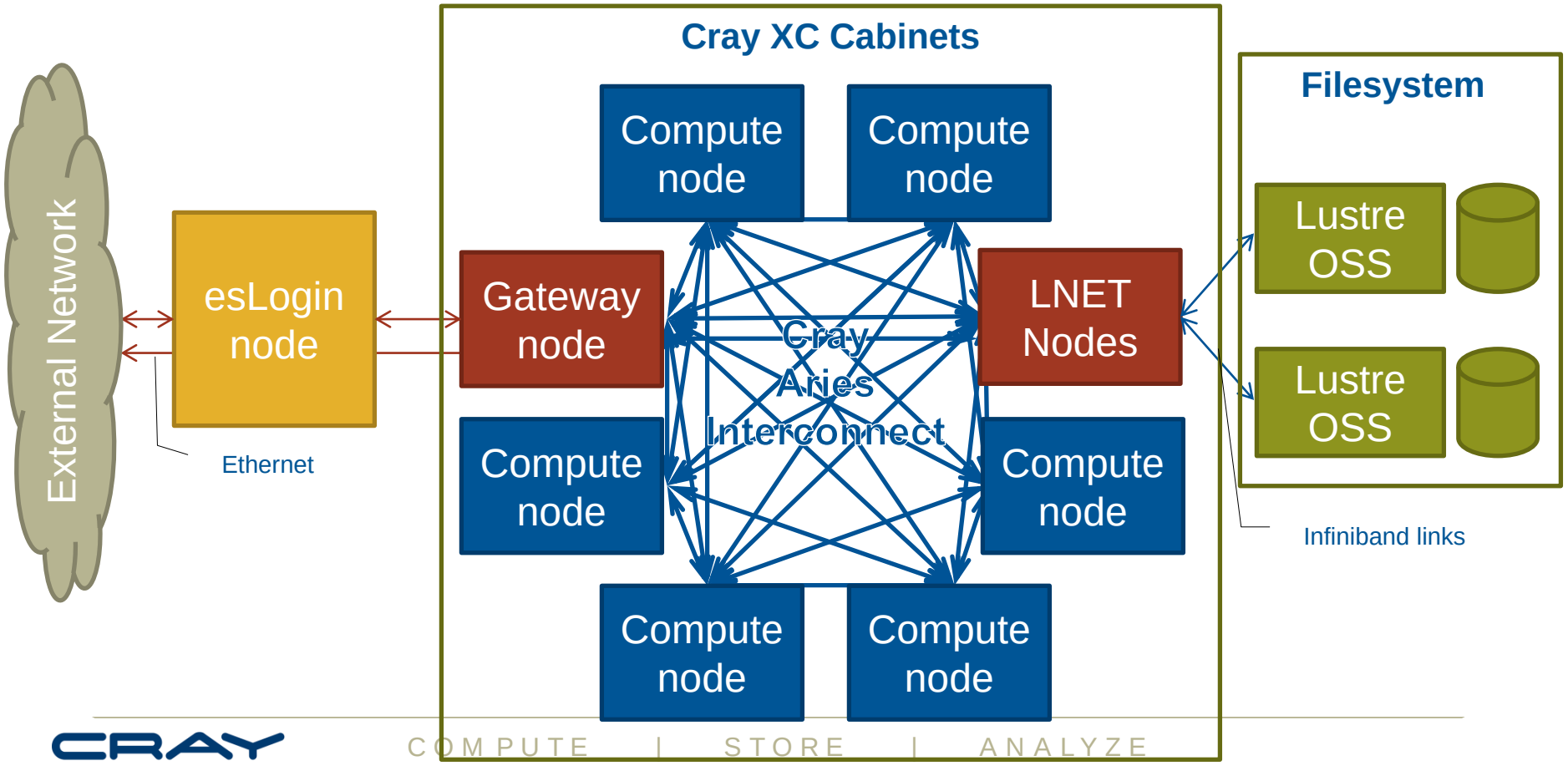
Obviously, to function as a single supercomputer, the individual nodes must have method to communicate with each other.

All nodes in the interconnected by the high speed, low latency Cray Aries Network.



Interacting with the system

Users do not log directly into the system. Instead they run commands via an esLogin server. This server will relay commands and information via a service node referred to as a "Gateway node"



Differences between nodes

Service nodes

- This is the node you access when you first log in to the system.
- They run a full version of the CLE operating system (all libraries and tools available)
- They are used for editing files, compiling code, submitting jobs to the batch queue and other interactive tasks.
- They are shared resources that may be used concurrently by multiple users.
- There may be many service nodes in any Cray XC30 and can be used for various system services (login nodes, IO routers, daemon servers).

Compute nodes

- These are the nodes on which production jobs are executed
- They run Compute Node Linux, a version of the OS optimised for running batch workloads
- They can only be accessed by submitting jobs through a batch management system (e.g. PBS Pro, Moab, SLURM)
- They are exclusive resources that may only be used by a single user.
- There are many more compute nodes in any Cray XC30 than login or service nodes.

EXCEPTION: When is a compute node not a compute node?

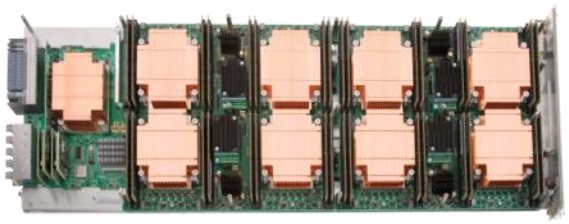
- When it is a MAMU Node.
- Some customers call these pre-/post- processing or shared nodes

- These are compute nodes running the full Cray Linux Environment operating system.
- Used for sharing hardware between jobs/users running on less than a whole node.
  - i.e. serial jobs, small parallel jobs
- Essentially looks like a service node using compute node hardware.
- Accessible via PBS setup

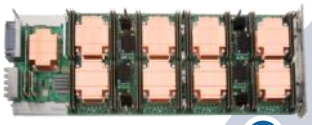
Further Node Terms

- **Users cannot interact with these nodes directly**
- **SMW – System Management Workstation**
  - Used by system administrators to perform system tasks.
- **RSIP Nodes – Realm-Specific IP Nodes**
  - Service nodes that provide connectivity to external networks
- **Boot Node**
- **SDB Node – System Data Base**

XC Architecture and Packaging



Cray XC System Building Blocks



Compute Blade

4 Compute Nodes

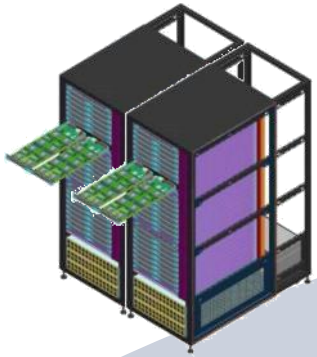
Chassis

Rank 1 Network

16 Compute Blades

No Cables

64 Compute Nodes



Group

Rank 2 Network

Passive Electrical Network

2 Cabinets

6 Chassis

384 Compute Nodes

System

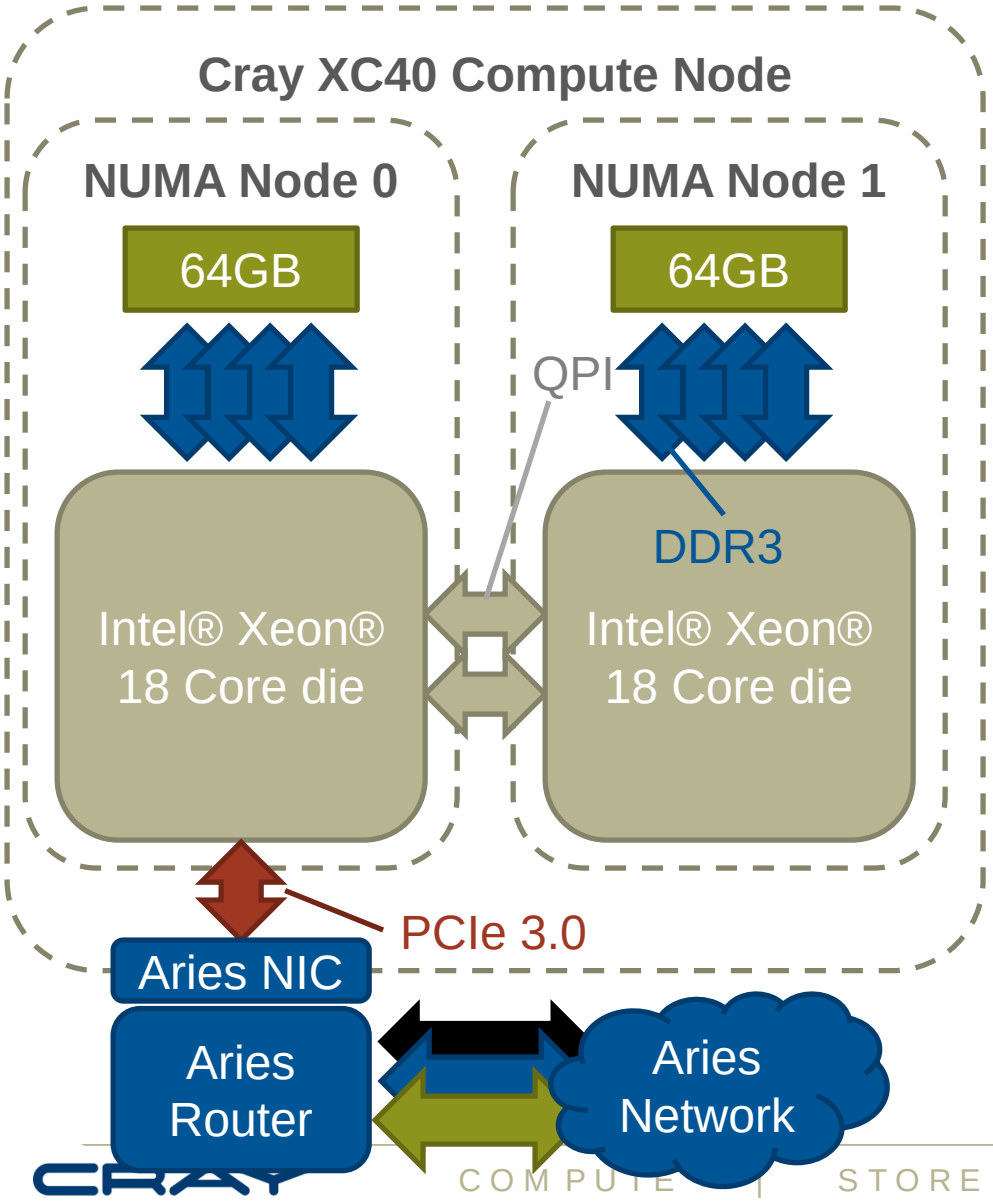
Rank 3 Network

Active Optical Network

Hundreds of Cabinets

Up to 10s of thousands of nodes

Cray XC40 Intel® Xeon® Compute Node



The XC40 Compute node features:

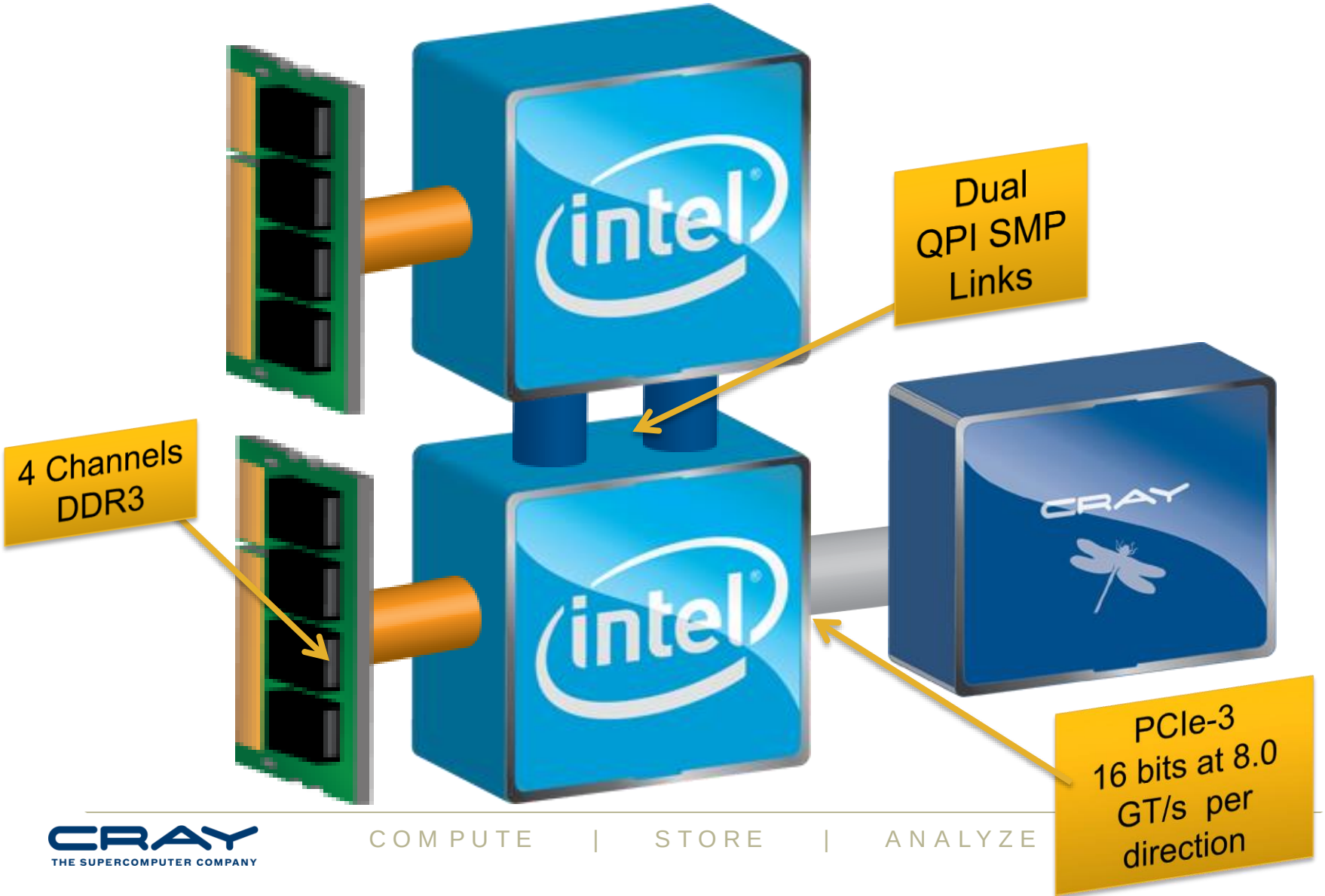
- **2 x Intel® Xeon® Sockets/die**
  - 18 core Broadwell
  - QPI interconnect
  - Forms 2 NUMA nodes
- **8 x 2400MHz DDR4**
  - 16 GB per Channel
  - 128 GB total
- **1 x Aries NIC**
  - Connects to shared Aries router and wider network
  - PCI-e 3.0

Broadwell CPU

- **Intel Advanced Vector Extensions 2 (AVX2)**
  - 256-bit integer vectors
  - Fused Multiply-Add (FMA)
  - Full-width element permutes
  - Gather
- **2x cache bandwidth**
  - 32-byte load/store for L1
  - 2x L2-to-L1 bandwidth

	Broadwell
Instruction set	AVX2 & FMA
DP Flop / cycle	16
L1 Data cache	32kB 8-way
Load BW	64 B/cycle
Store BW	32 B/cycle
L2 Unified cache	256 kB 8-way
BW to L1	64 B/cycle

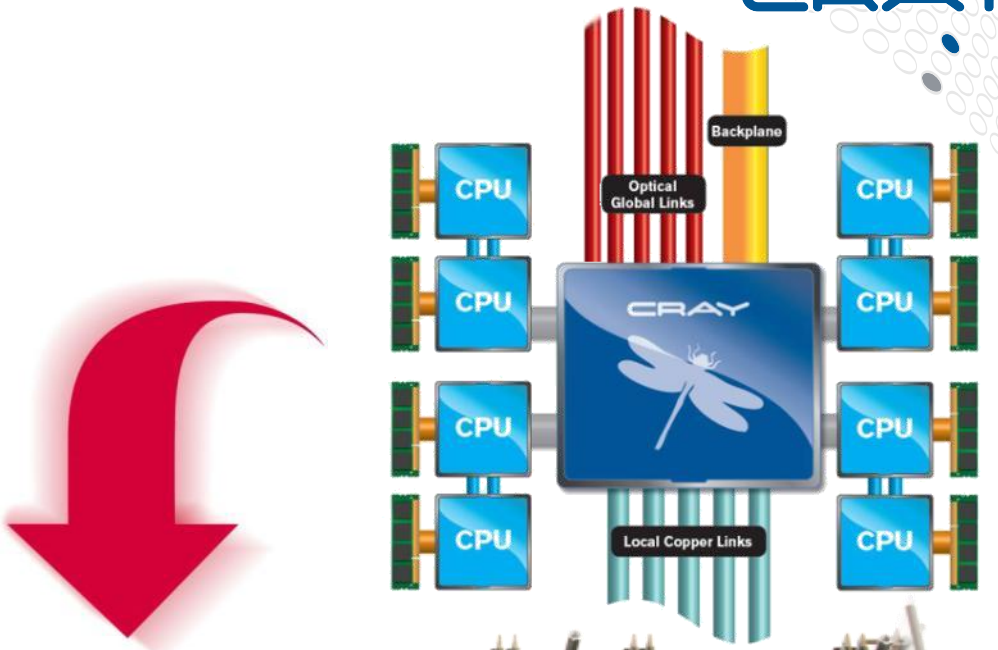
Cray XC40 Compute Node Architecture

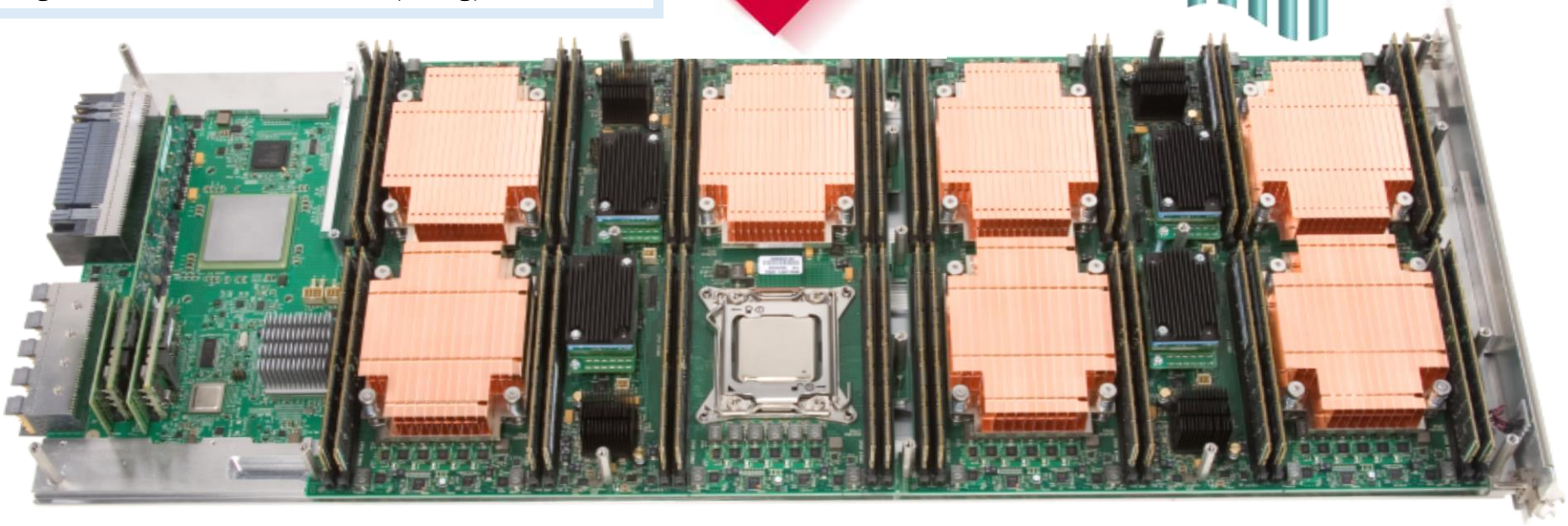


Cray XC Fully Populated Compute Blade

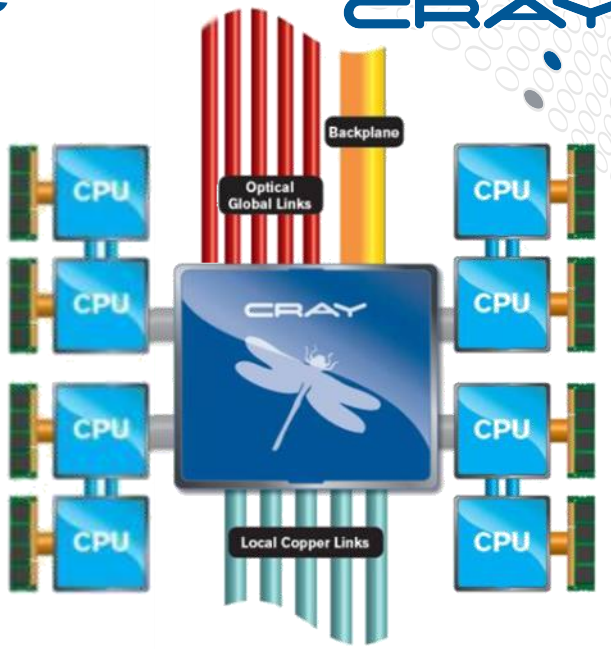
SPECIFICATIONS

Module power:	2014 Watts
PDC max. power:	900 Watt
Air flow req.:	275 cfm (7.8 m³/min)
Size:	2.125 in x 12.95 in x 33.5 in
Weight:	<40 lbm (18 kg)

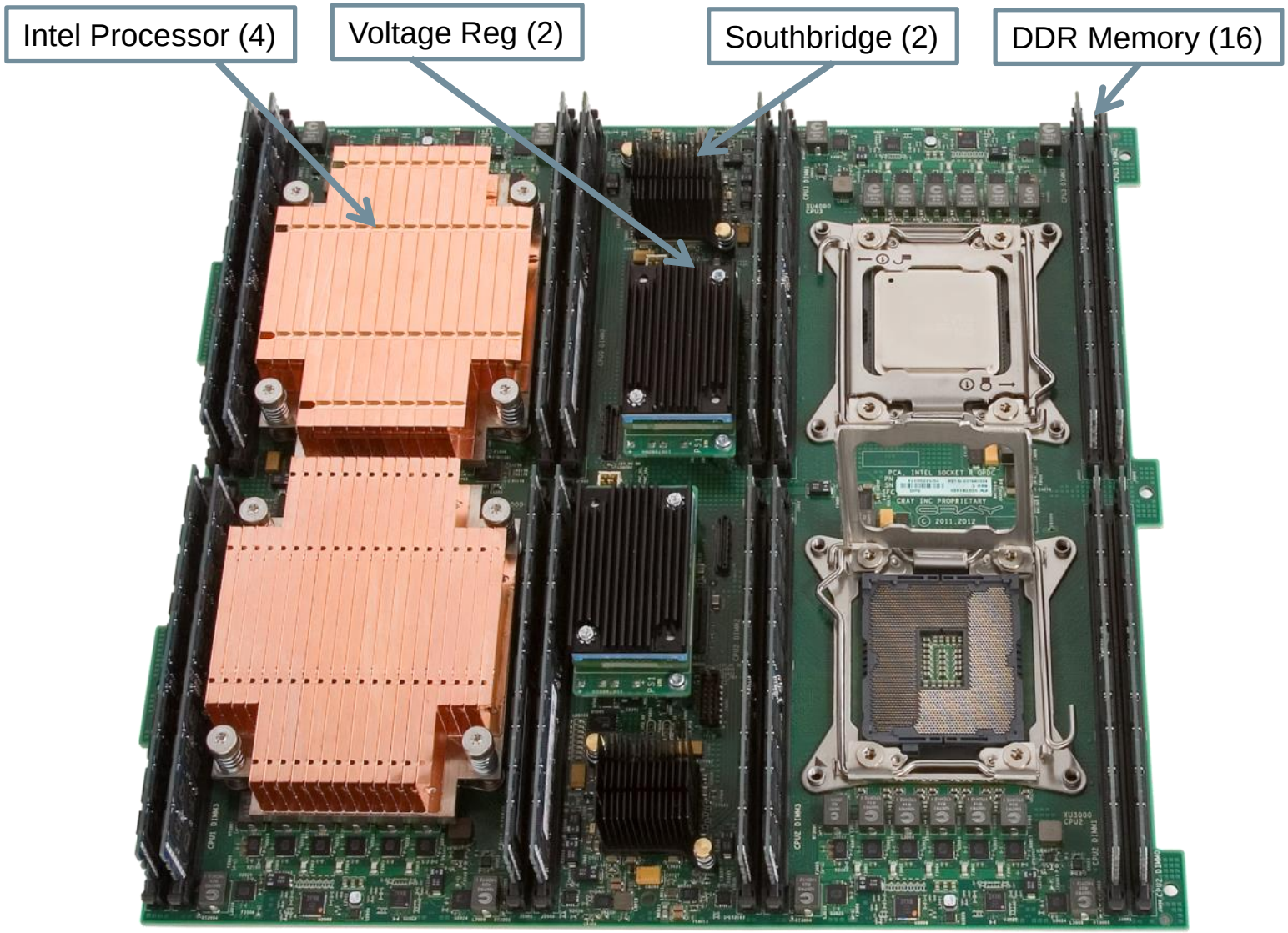




PDC's (Processor Daughter Card) are Upgradeable to New Technology



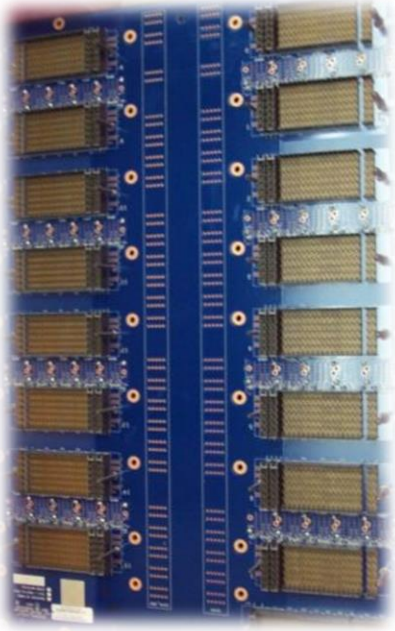
Cray XC Quad Processor Daughter Card



Aries and the Dragonfly topology

Cray XC Network

- The Cray XC system is built around the idea of optimizing interconnect bandwidth and associated cost at every level

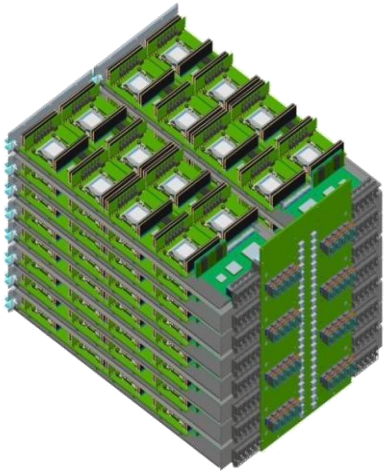


Rank-1
PC Board: ¢¢¢

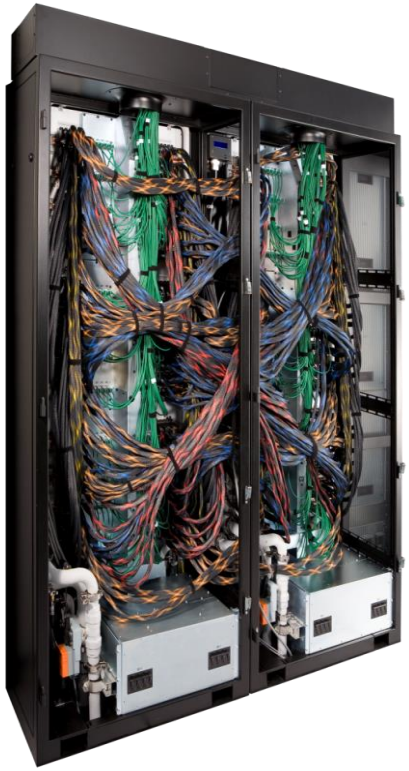
Rank-2
Passive CU: $

Rank-3
Active Optics: $$$$

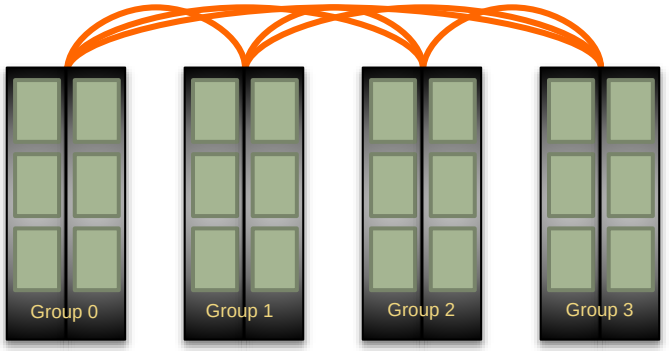
Cray XC Packaging Review



Rank-1
Chassis



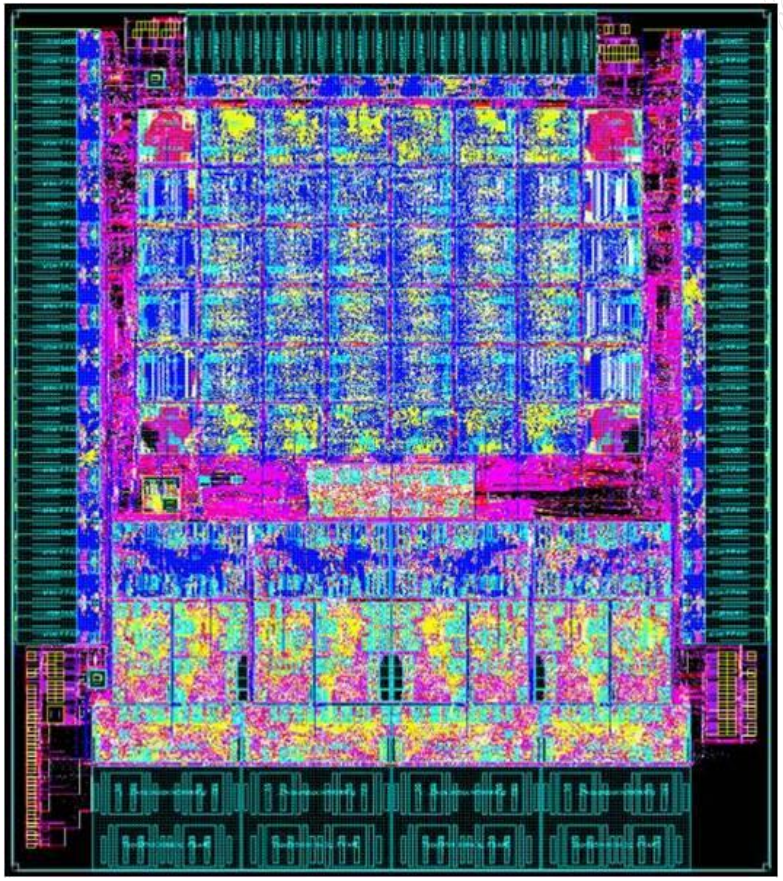
Rank-2
2 Cabinet Group



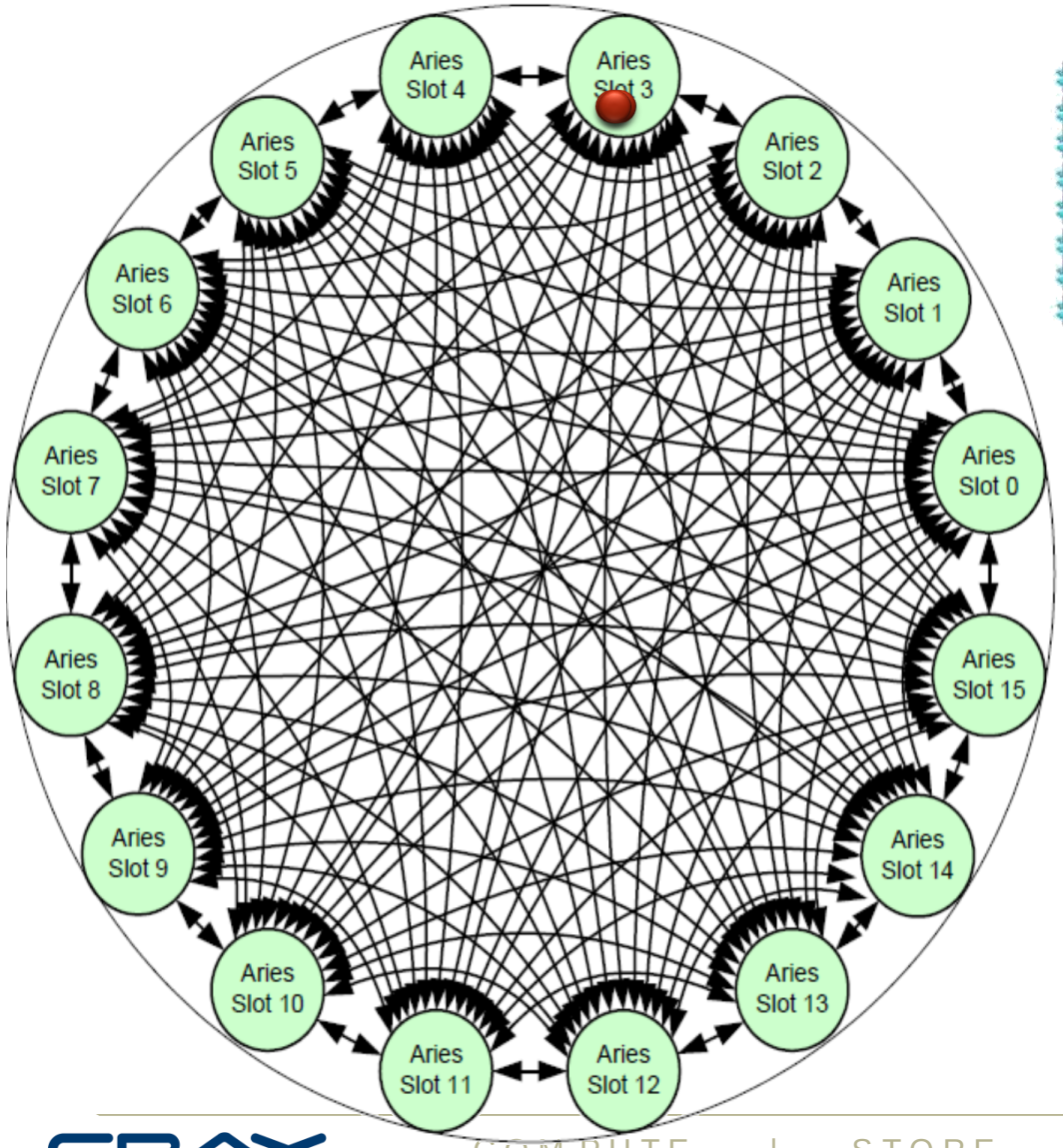
Rank-3
Between Groups

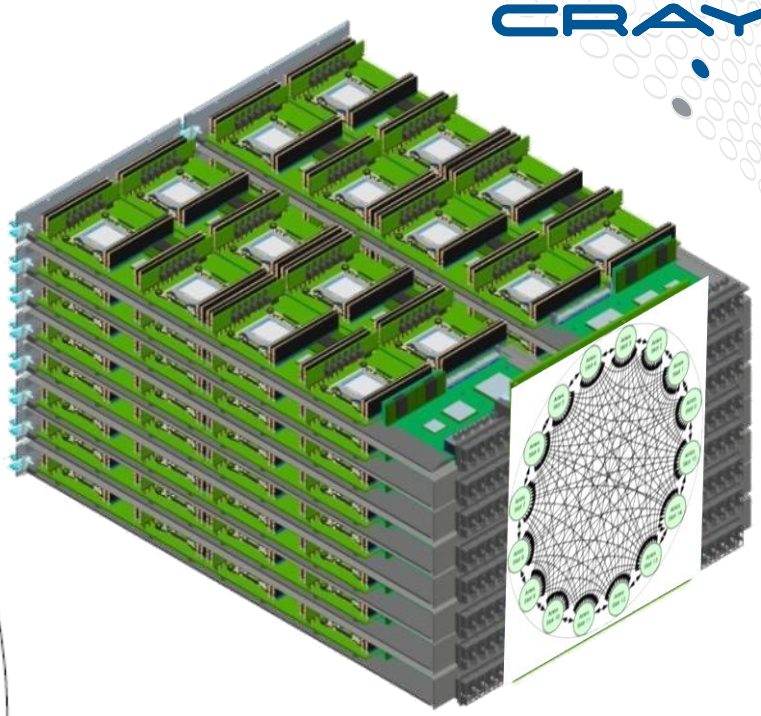
Aries

- Aries is the Cray custom interconnect ASIC used in the Cray XC product family
  - 40nm process
  - Die size: 16.6 x 18.9mm
  - Gate count: 217M
  - 184 lanes of high speed SerDes
    - SerDes=Serializer/Deserializer (SerDes pronounced sir-deez)
  - 30 optical network lanes
  - 90 electrical network lanes
  - 64 PCI Express lanes
- **4 NICs**
  - Each Aries connects 4 nodes to the interconnect (Gemini connects 2)



Cray XC Rank1 Network



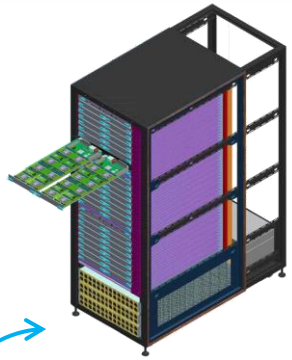


- Chassis with 16 compute blades
- 128 Sockets
- Inter-Aries communication over backplane
- Per-Packet adaptive Routing

Cray XC Rank-2 Copper Network

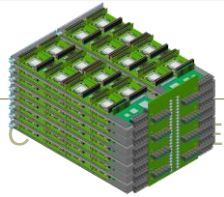
2 Cabinet Group
768 Sockets

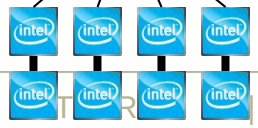
6 backplanes connected with copper cables in a 2-cabinet group: "Black Network"



Active optical cables interconnect groups "Blue Network"

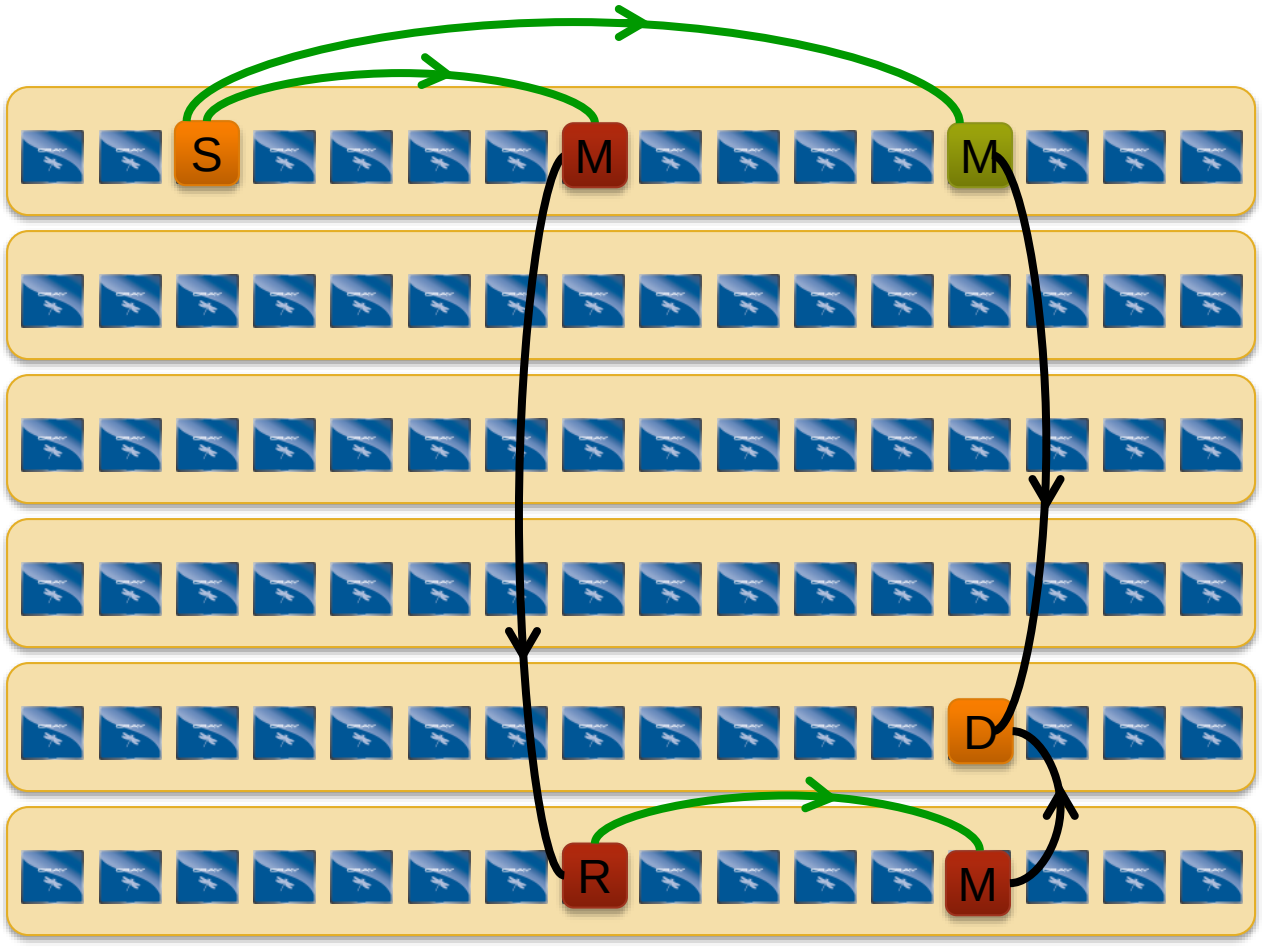
16 Aries connected by backplane "Green Network"





4 nodes connect to a single Aries

Cray XC Routing



Minimal routes between any two nodes in a group are just two hops

Non-minimal route requires four hops.

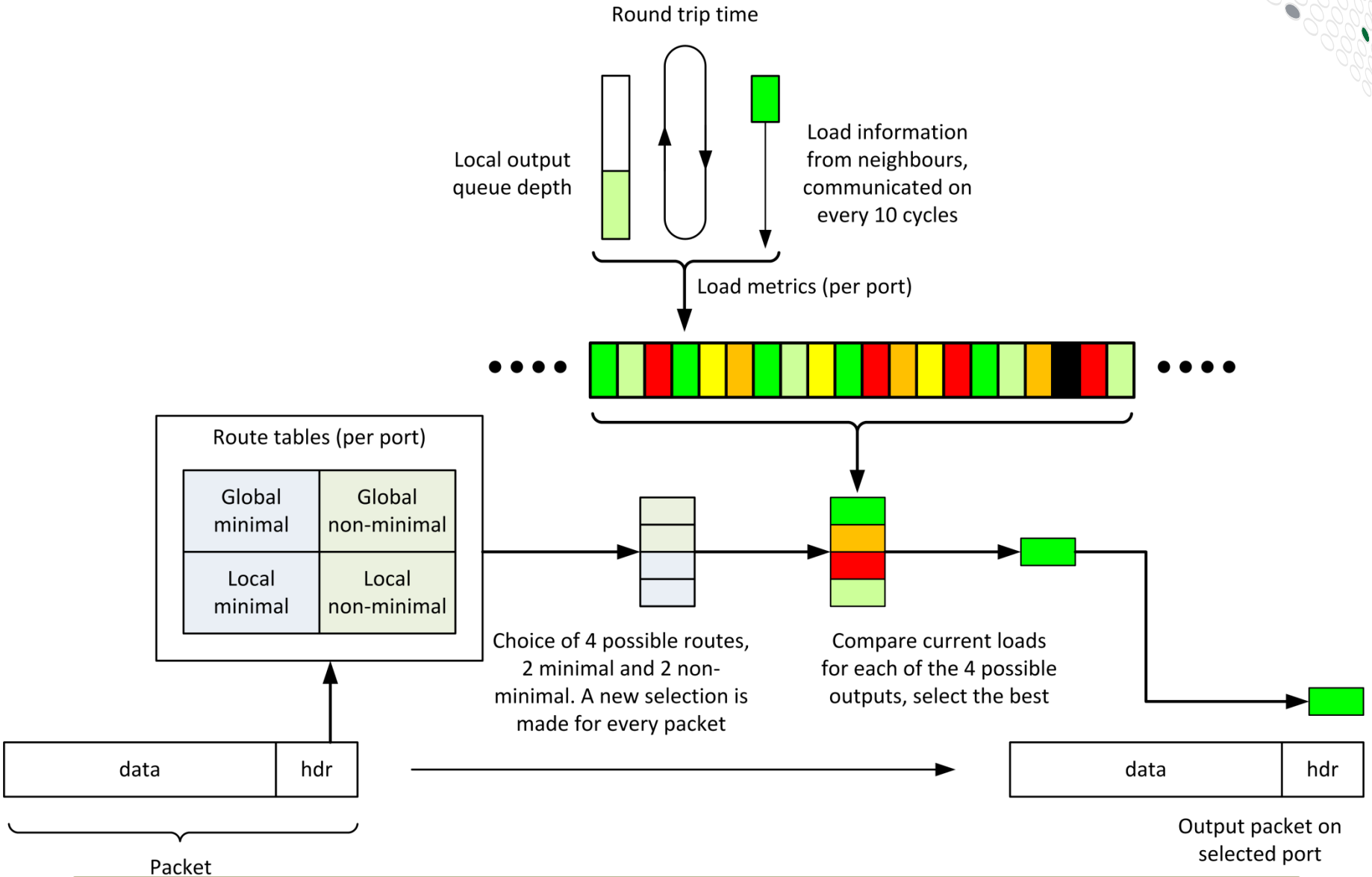
With adaptive routing we select between minimal and non-minimal paths based on load

The Cray XC Class-2 Group has sufficient bandwidth to support full injection rate for all 384 nodes with non-minimal routing

Cray XC40 Rank-2 Cabling

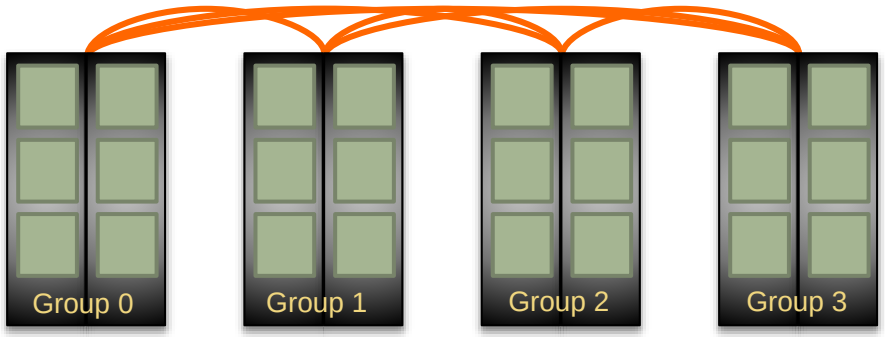
- **Cray XC30 two-cabinet group**
  - 768 Sockets
  - 96 Aries Chips
- **All copper and backplanes signals running at 14 Gbps**

Aries Adaptive Routing Algorithm



Cray XC Network Overview – Rank-3 Network

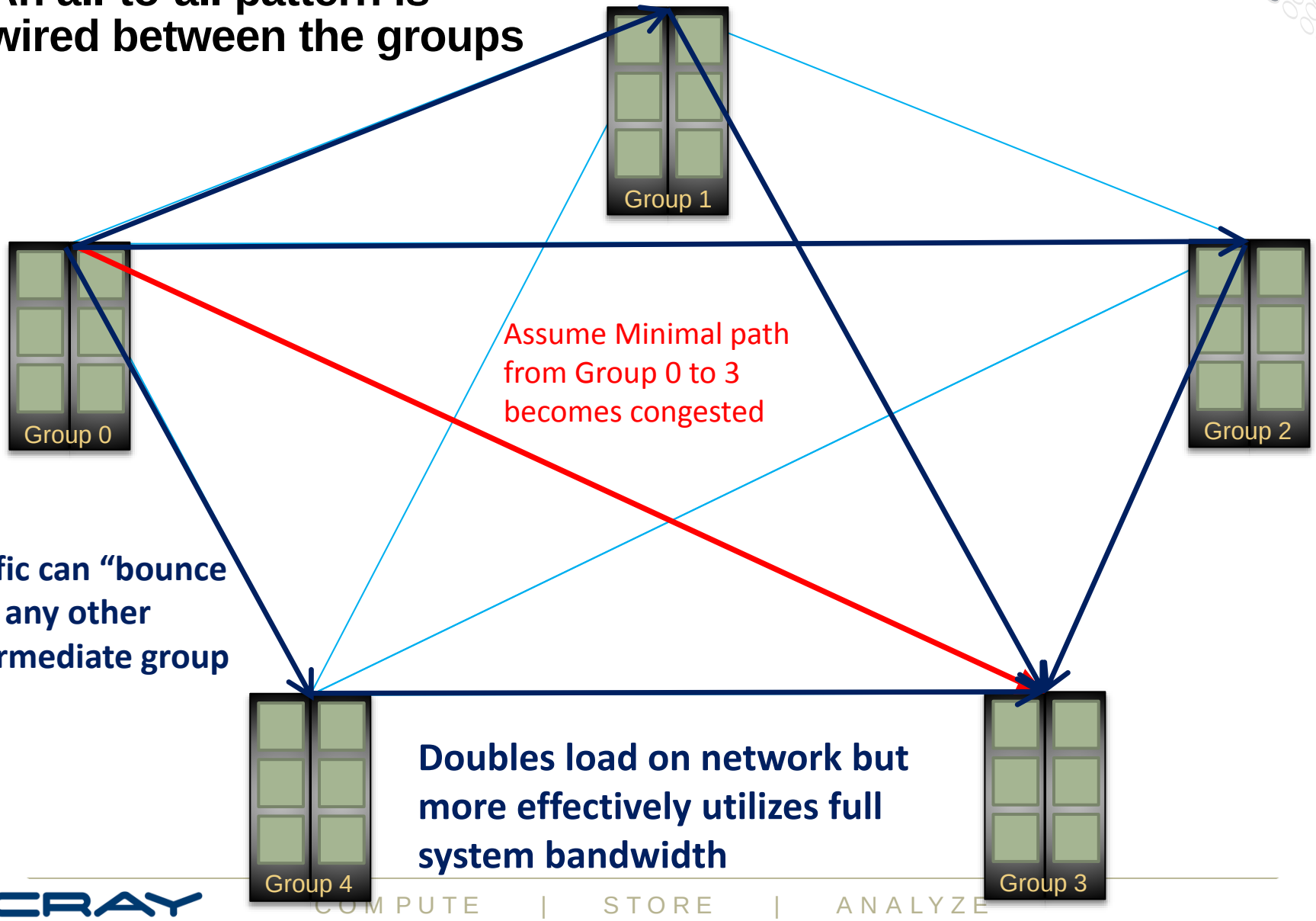
- An all-to-all pattern is wired between the groups using optical cables (blue network)
- Up to 240 ports are available per 2-cabinet group
- The global bandwidth can be tuned by varying the number of optical cables in the group-to-group connections



Example: A 4-group system is interconnected with 6 optical "bundles". The "bundles" can be configured between 20 and 80 cables wide

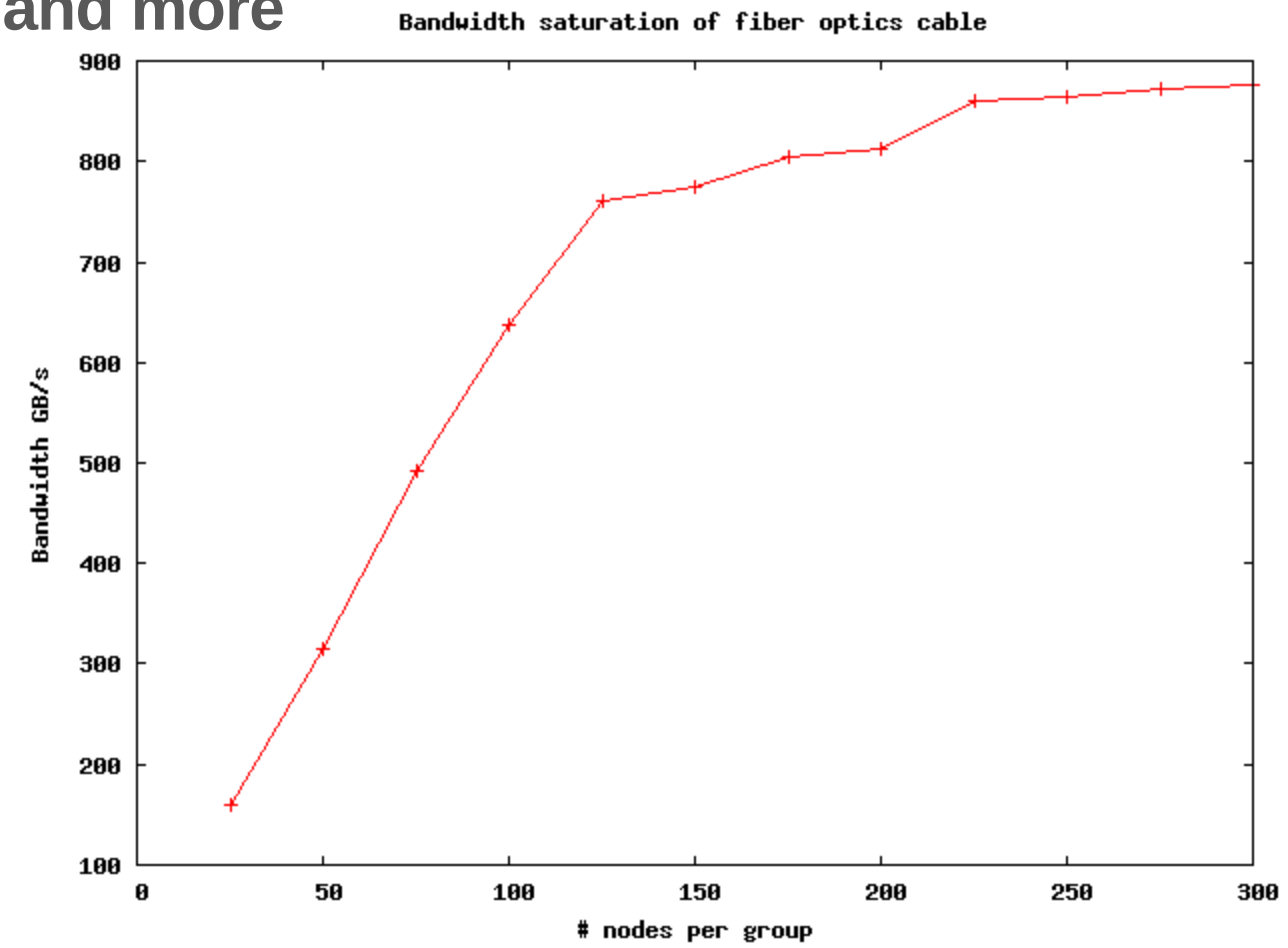
Adaptive Routing over the Blue Network

- **An all-to-all pattern is wired between the groups**



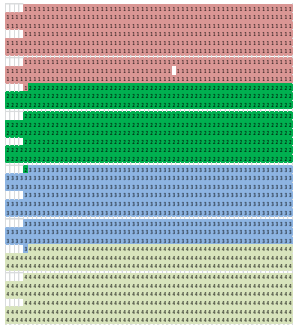
Optical network saturation using the OSU MPI BM This runs were done on Hornet (HLRS) with the 25% opt. cables config

- Saturating the optical network using communications between more and more nodes within 2 groups:
  - 2x300 nodes: 875 GB/s



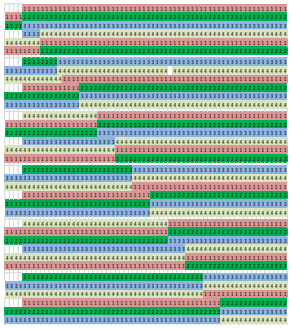
Dragonfly is placement insensitive

- Example: Sandia miniApp, miniGhost
- Running on 2256 node (12 Cabinets) CSCS system (¼ global bandwidth)
  - Runtime in seconds for 100 cycles

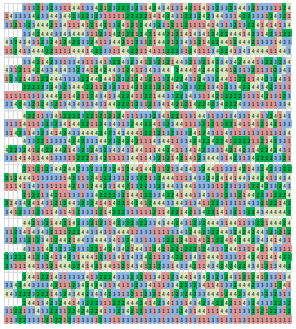


Contiguous Blocks of 512 nodes			
69.0	68.8	68.9	68.9

Perfect Placement



Random blocks of 64 nodes			
69.4	69.4	69.4	69.5



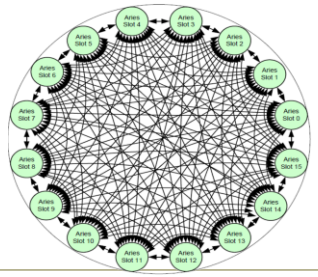
Random layout of nodes			
70.9	71.0	70.6	70.5

< 3% variance from best-case to worst-case placement

Worst-Case Placement

Does Aries handle MPI Traffic with I/O Traffic ?

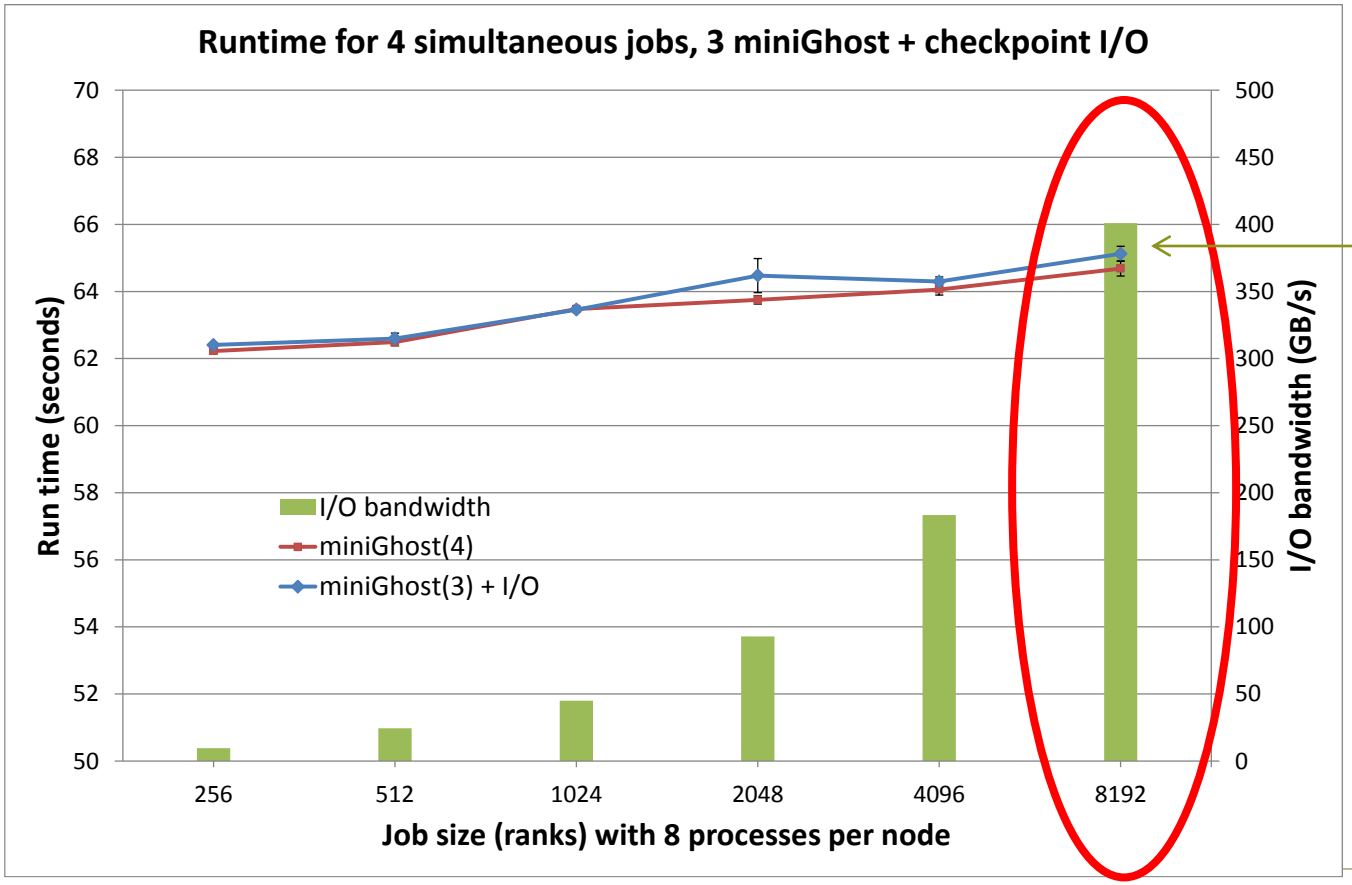
I/O Messages



MPI Messages

Mix of application and streaming I/O traffic

- Analysis of the impact of big I/O traffic on performance of other codes
- Compared two runs
  1. Four miniGhost jobs spread out across the whole machine vs.
  2. Three miniGhost plus one performing big many-to-few I/O



I/O Job sustaining 400GB/sec (95% clients to 5% servers)

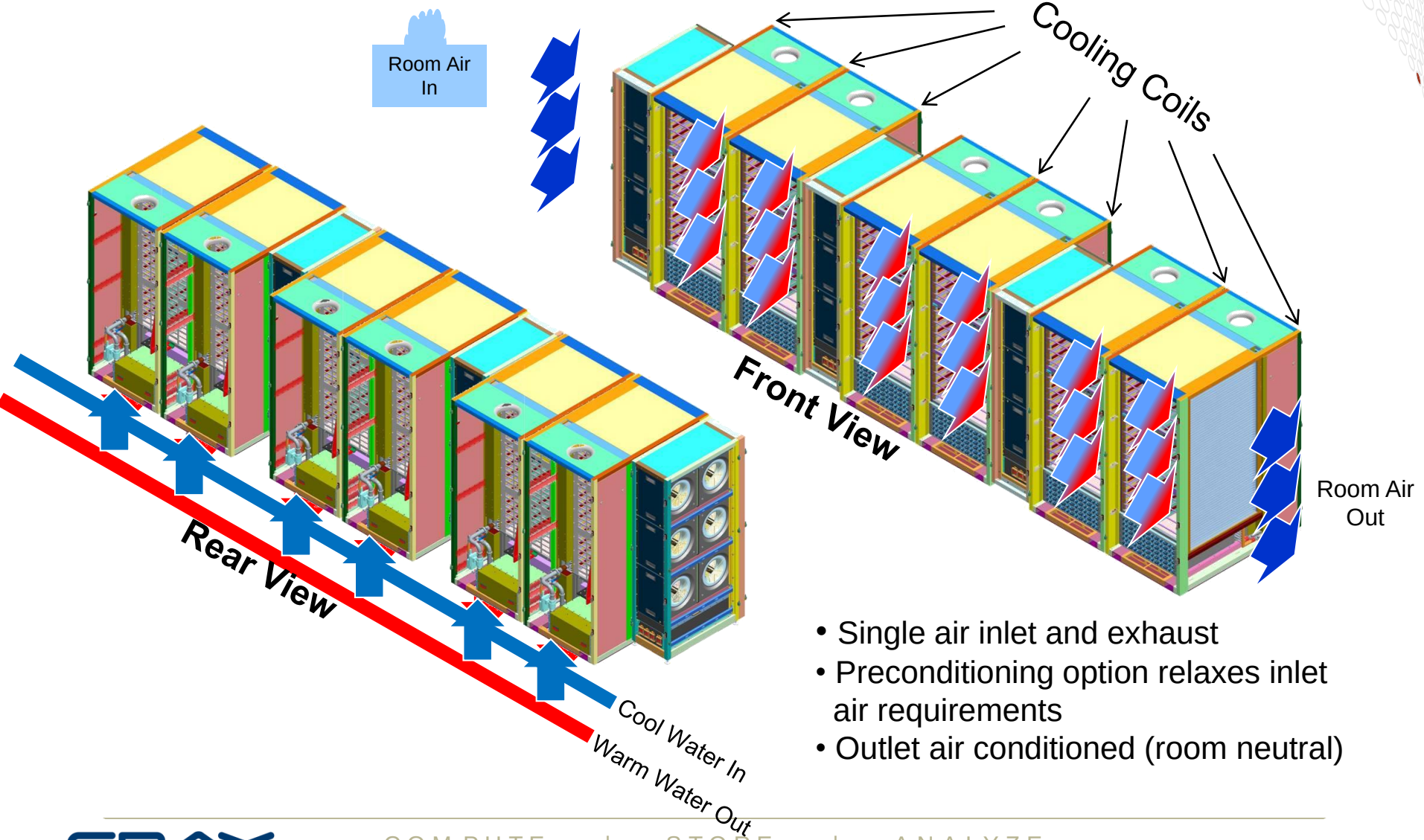
Impact to compute jobs is tiny (64.5 sec to 65 sec)

Why is the Dragonfly topology a good idea?

- **Scalability**
  - Topology scales to very large systems
- **Performance**
  - More than just a case of clever wiring, this topology leverages state-of-the-art adaptive routing that Cray developed with Stanford University
  - Smoothly mixes small and large messages
  - *Cray invested in bringing it to market – IBM and Mellanox have not*
- **Simplicity**
  - Implemented *without* external switches
  - No HBAs or separate NICs and Routers
- **Cost**
  - Dragonfly maximizes the use of backplanes and passive copper components
  - Dragonfly minimizes the use of active optical components

XC
Cooling

XC Cooling Overview

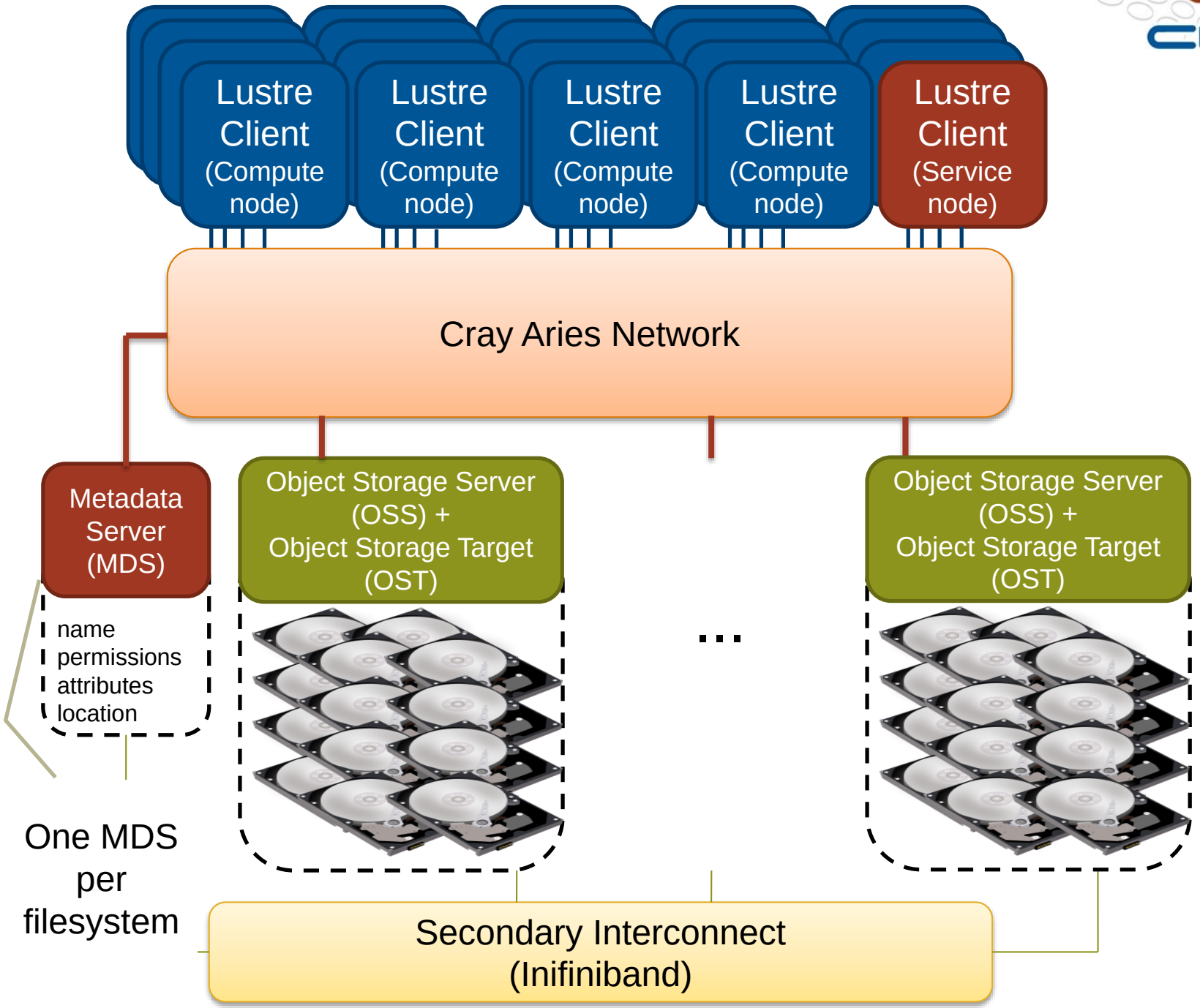


- Single air inlet and exhaust
- Preconditioning option relaxes inlet air requirements
- Outlet air conditioned (room neutral)

Cray XC Transverse Cooling Advantages

- **Performance**
  - Transverse cooling and graduated heat sink pitch ensure that all processors operate in the same thermal envelope
  - “Turbo mode” works like it should in a parallel job
- **Simplicity**
  - No airflow issues to manage or adjust
  - System is 100% water-cooled
  - No pumps, refrigerant, treated water, or plumbing on the blades
- **Cost of Ownership**
  - Excellent PUE characteristics
  - 25% better density than other ‘direct’ water cooled solution
  - All cooling infrastructure is retained across multiple generations of computing technology
- **Maintainability**
  - Blades can be warm-swapped without disturbing any plumbing
  - Blowers can be hot-swapped if required and can provide N+1 redundancy

Storage


Lustre Client (Compute node)
Lustre Client (Compute node)
Lustre Client (Compute node)
Lustre Client (Compute node)
Lustre Client (Service node)
Cray Aries Network
Metadata Server (MDS)
name
permissions
attributes
location
Object Storage Server (OSS) + Object Storage Target (OST)
...
Object Storage Server (OSS) + Object Storage Target (OST)
One MDS per filesystem
Secondary Interconnect (Inifiniband)

Sonexion: Only Three Components

MMU: *Metadata Management Unit*

1

Fully integrated metadata module

- Lustre Metadata software
- Metadata disk storage
- Dual redundant management servers
- Metadata storage target RAID

SSU: *Scalable Storage Unit*

2

Fully integrated storage module

- Storage controller, Lustre server
- Disk controller, RAID engine
- High speed storage
- Provides both capacity and performance

3

Fully prepared rack

- Prewired for InfiniBand, Ethernet and power
- Ready for instant expansion

Programming Environment

Vision

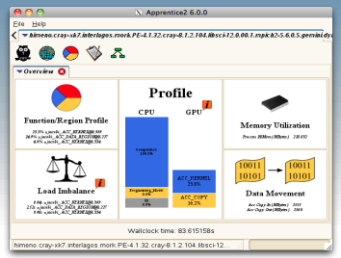
- **Cray systems are designed to be High Productivity as well as High Performance Computers**
- **The Cray Programming Environment (PE) provides a simple consistent interface to users and developers.**
  - Focus on improving scalability and reducing complexity
- **The default Programming Environment provides:**
  - the highest levels of application performance
  - a rich variety of commonly used tools and libraries
  - a consistent interface to multiple compilers and libraries
  - an increased automation of routine tasks
- **Cray continues to develop and refine the PE**
  - Frequent communication and feedback to/from users
  - Strong collaborations with third-party developers

Cray XC: Focus on User Productivity

Load & Go

Build & Go

Tune & Go



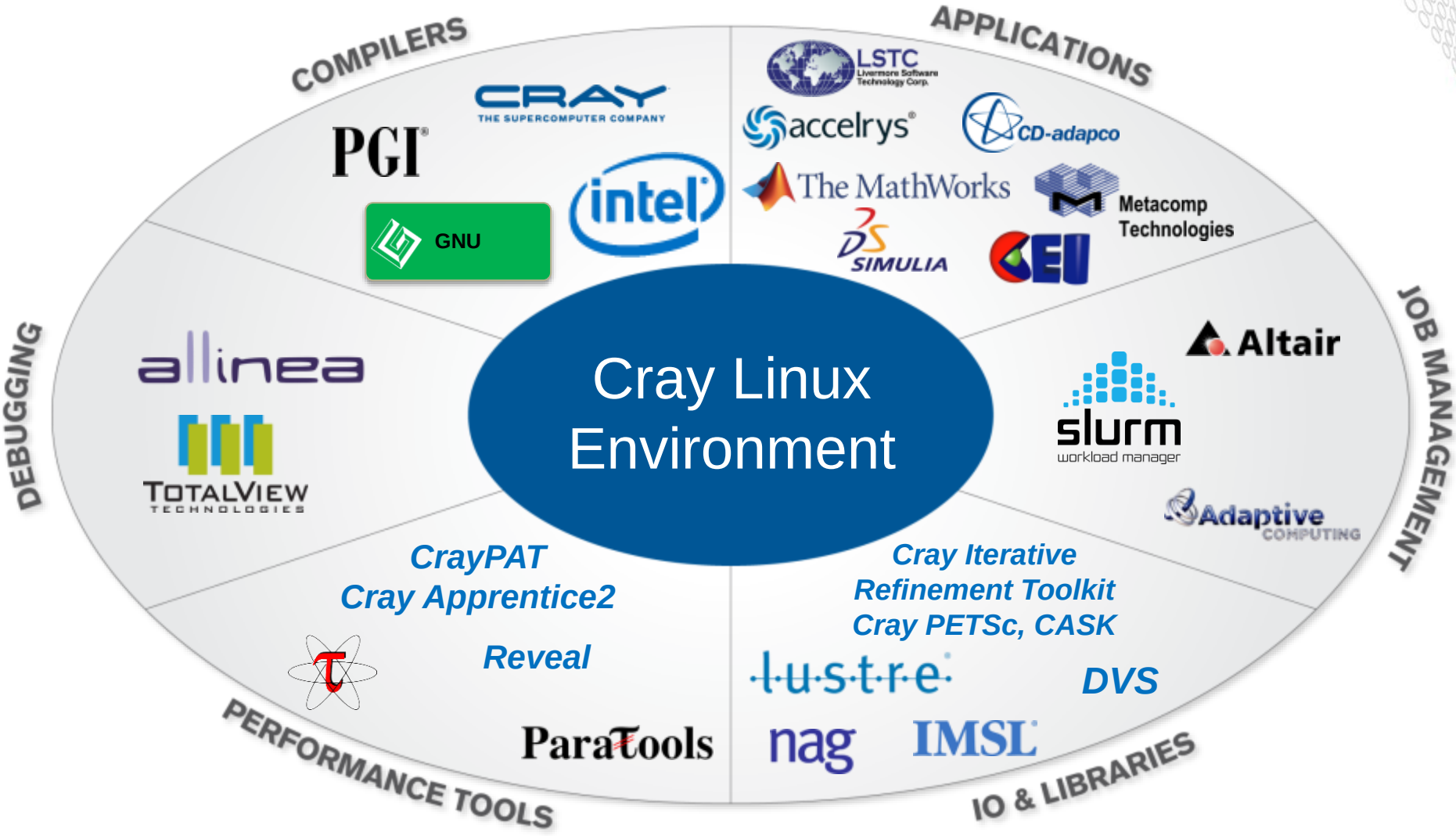
Code & Go

No Code Development

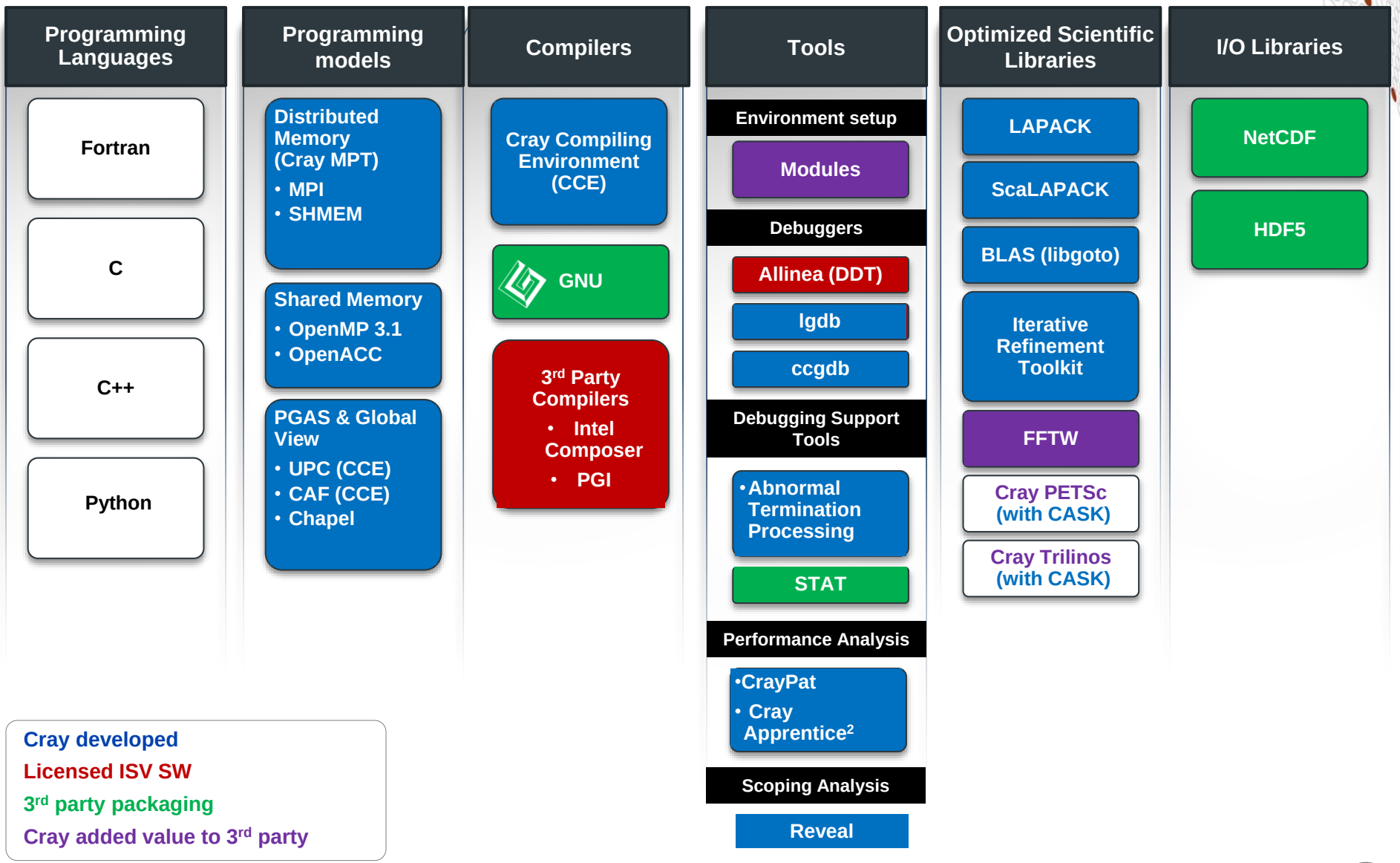
New Code Development

Cray XC provides great support across the full spectrum of HPC user types

Cray Software Ecosystem



Cray's Supported Programming Environment



The Cray Compilation Environment (CCE)

- **The default compiler on XE and XC systems**
  - Specifically designed for HPC applications
  - Takes advantage of Cray's experience with automatic vectorization and and shared memory parallelization

- **Excellent standards support for multiple languages and programming models**
  - Fortran 2008 standards compliant
  - C++98/2003 compliant (working on C++11)
  - OpenMP 3.1 compliant, working on OpenMP 4.0
  - OpenACC 2.0 compliant

- **Full integrated and optimised support for PGAS languages**
  - UPC 1.2 and Fortran 2008 coarray support
  - No preprocessor involved
  - Full debugger support (With Allinea DDT)

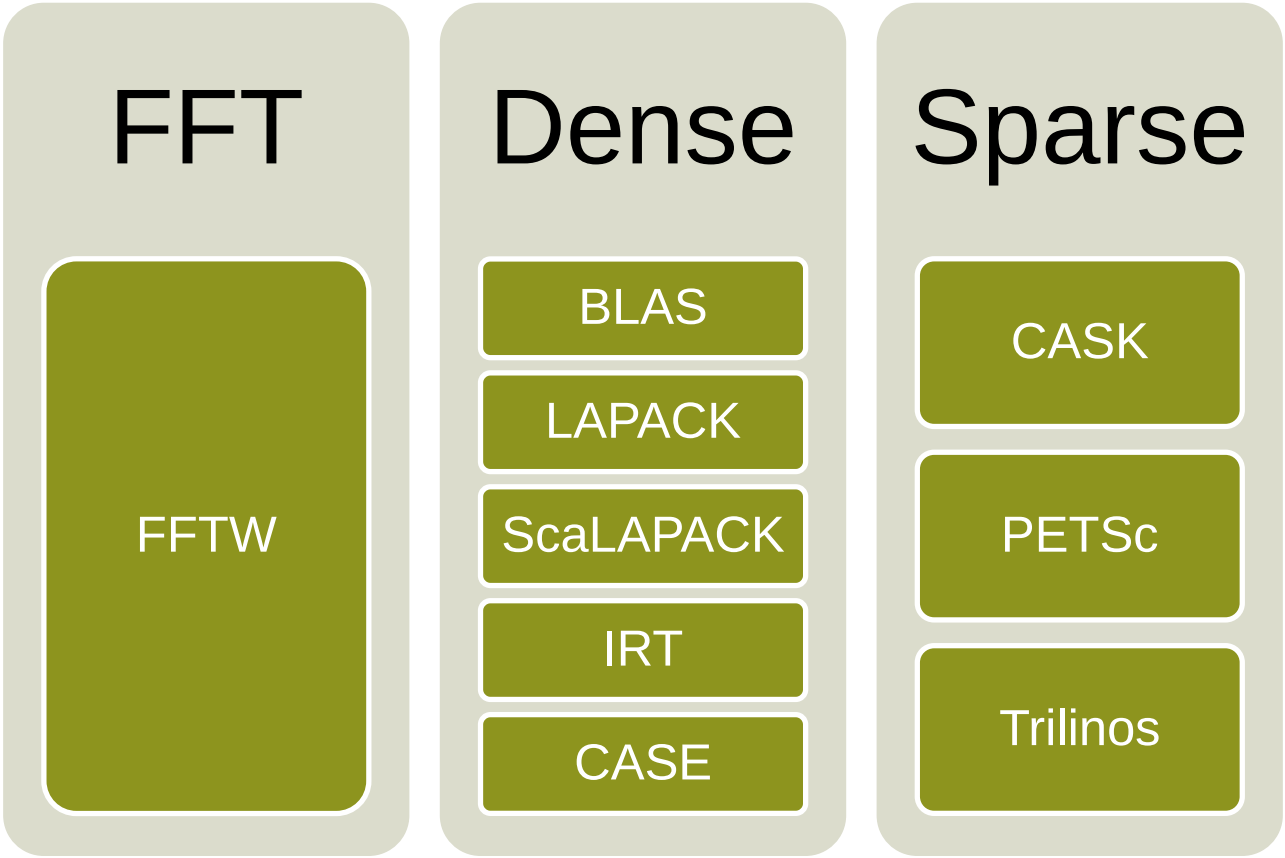
- **OpenMP and automatic multithreading fully integrated**
  - Share the same runtime and resource pool
  - Aggressive loop restructuring and scalar optimization done in the presence of OpenMP
  - Consistent interface for managing OpenMP and automatic multithreading

Cray MPI & SHMEM

- **Cray MPI**
  - Implementation based on MPICH3 source from ANL
  - Includes many improved algorithms and tweaks for Cray hardware
    - Improved algorithms for many collectives
    - Asynchronous progress engine allows overlap of computation and comms
    - Customizable collective buffering when using MPI-IO
    - Optimized Remote Memory Access (one-sided) fully supported including passive RMA
  - Full MPI-3 support with the exception of
    - Dynamic process management (eg. MPI_Comm_spawn)
    - MPI_LONG_DOUBLE and MPI_C_LONG_DOUBLE_COMPLEX for CCE
  - Includes support for Fortran 2008 bindings (from CCE 8.3.3)

- **Cray SHMEM**
  - Fully optimized Cray SHMEM library supported
    - Fully compliant with OpenSHMEM v1.0
    - Cray XC implementation close to the T3E model

Cray Scientific Libraries



IRT – Iterative Refinement Toolkit
CASK – Cray Adaptive Sparse Kernels
CASE – Cray Adaptive Simplified Eigensolver

Cray Performance Analysis Tools (PAT)

- **From performance measurement to performance analysis**
- **Assist the user with application performance analysis and optimization**
  - Help user identify important and meaningful information from potentially massive data sets
  - Help user identify problem areas instead of just reporting data
  - Bring optimization knowledge to a wider set of users
- **Focus on ease of use and intuitive user interfaces**
  - Automatic program instrumentation
  - Automatic analysis
- **Target scalability issues in all areas of tool development**

Debuggers on Cray Systems

- **Systems with hundreds of thousands of threads of execution need a new debugging paradigm**
  - Innovative techniques for productivity and scalability
    - Scalable Solutions based on MRNet from University of Wisconsin
    - STAT - Stack Trace Analysis Tool
      - Scalable generation of a single, merged, stack backtrace tree
      - running at 216K back-end processes
    - ATP - Abnormal Termination Processing
      - Scalable analysis of a sick application, delivering a STAT tree and a minimal, comprehensive, core file set.
    - Fast Track Debugging
      - Debugging optimized applications
      - Added to Allinea's DDT 2.6 (June 2010)
    - Comparative debugging
      - A data-centric paradigm instead of the traditional control-centric paradigm
      - Collaboration with Monash University and University of Wisconsin for scalability
  - Support for traditional debugging mechanism
    - TotalView, DDT, and gdb