The Cray XT4 Programming Environment

CRAY
THE SUPERCOMPUTER COMPANY

Getting to know CNL

CRAY
THE SUPERCOMPUTER COMPANY

Disclaimer

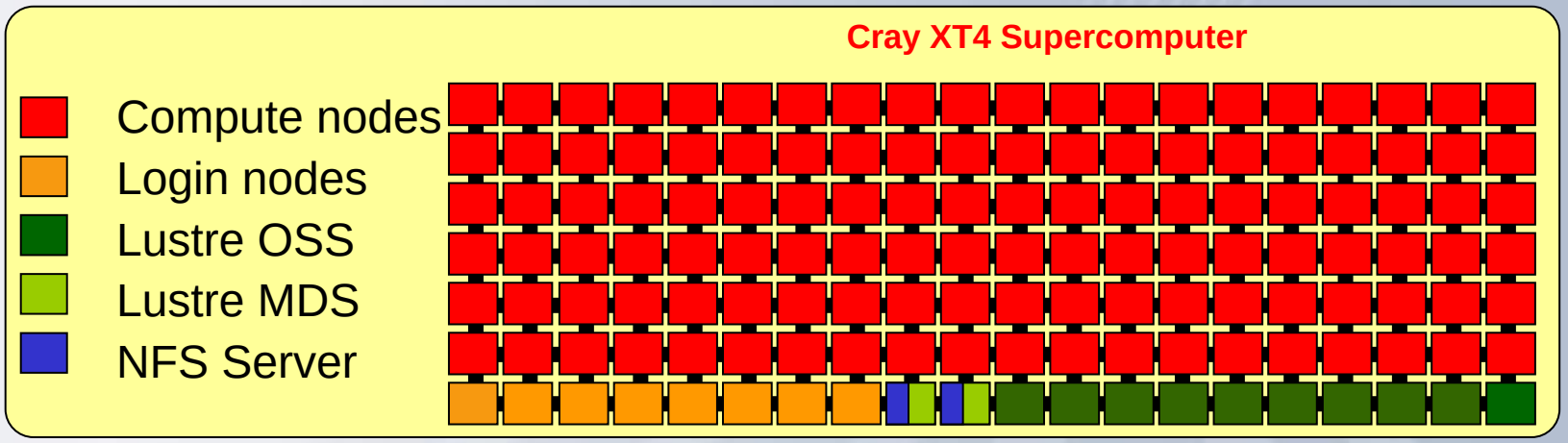
- This talk is not a conversion course from Catamount, it makes assumptions that attendees know Linux.
- This talk documents Cray's tools and features for CNL. There will be a number of locations which will be highlighted where optimizations could have been made under Catamount that are no longer needed with CNL. There will be many publications documenting these and it is important to know that these no longer apply.
- There is a tar file of scripts and test codes that are used to test various features of the system as the talk progresses
- This talk as it stands is specific to HECToR, and will be continued to be maintained whilst HECToR is CNL.

Agenda

- Brief XT4 Overview
  - Hardware, Software, Terms
- Getting in and moving around
  - System environment
  - Hardware setup
- Introduction to CNL features (**NEW**)
- Programming Environment / Development Cycle
  - Job launch (**NEW**)
  - modules
- Compilers
  - PGI, Pathscale compilers: common flags, optimization
- CNL programming (**NEW**)
  - system calls
  - timings
- I/O optimization
  - I/O architecture overview
  - Lustre features
  - lfs command
- Topology

The Processors

- The login nodes run a full Linux distribution
- There are a number of nodes dedicated to I/O (we'll talk about those later)
- The compute nodes run Compute Node Linux (CNL)
- We will need to cross-compile our codes to run on the compute nodes from the login nodes.



Glossary

- ALPS
  - Application Level Placement Scheduler
- CNL
  - Compute Node Linux
- RSIP
  - Realm-Specific Internet Protocol
  - The framework or architecture as defined in RFC 3102 for enabling hosts on private IP networks to communicate across gateways to hosts on public IP networks.

Getting In

- Getting in
  - The only recommended way of accessing Cray systems is ssh for security
  - Other sites have other methods for security including key codes and Grid certificates.
- Cray XT systems have separated service work from compute intensive batch work.
- You login in to anyone of a number of login or service nodes.
  - `hostname` can be different each time
  - Load balancing is done to choose which node you login to
- You are still sharing a fixed environment with a number of others
  - Which may still run out of resources
- Successive login sessions may be on different nodes
  - I/O needs to go to disk, etc.

Moving Around

- You start in your home directory, this is where most things live
  - ssh keys
  - Files
  - Source code for compiling
  - Etc
- The home directories are mounted via NFS to all the service nodes

- The /work file system is the main lustre file system,
  - This file system is available to the compute nodes
  - Optimized for big, well formed I/O.
  - Small file interactions have higher costs.

- /opt is where all the Cray software lives
  - In fact you should never need to know this location as all software is controlled by modules so it is easier to upgrade these components

- /var is usually for spooled or log files
  - By default PBS jobs spool their output here until the job is completed (/var/spool/PBS/spool)
- /proc can give you information on
  - the processor
  - the processes running
  - the memory system
- Some of these file systems are not visible on backend nodes and maybe be memory resident so use sparingly!
  - You can use homegrown tool apls to investigate backend node file systems and permissions

Exercise 1:
Look around at the backend nodes look at the file systems and what is there, look at the contents of /proc.

```
make apls
aprun ./apls /
```

Introduction to CNL

- Most HPC systems run a full OS on all nodes.
- Cray have always realised that to increase performance, more importantly parallel performance, you need to minimize the effect of the OS on the running of your application.
- This is why CNL is a lightweight operating system
- CNL should be considered as a full Linux operating system with components that increase the OS interventions removed.
  - There has been much more work than this but this is a good view to take

Introduction to CNL

- The requirements for a compute node are based on Catamount functionality and the need to scale
  - Scaling to 20K compute sockets
  - Application I/O equivalent to Catamount
  - Start applications as fast as Catamount
  - Boot compute nodes almost as fast as Catamount
  - Small memory footprint

CNL

- CNL has the following features missing:
  - NFS – you cannot launch jobs from an NFS mounted directory
  - Dynamic libraries
  - A number of services may not be available also

- If you are not sure if something is supported, try man pages, e.g.:

```
> man mmap
NAME
      mmap, munmap - map or unmap files or devices into memory

IMPLEMENTATION
      UNICOS/lc  operating  system  -  not supported on Cray XT series
      compute nodes
```

CNL

- Has solved the requirement for threaded programs – OpenMP, pthreads
- Uses Linux I/O buffering for better I/O performance
- Has sockets for internal communication – RSIP can be configured for external communication
- Has become more Linux like for user convenience
- Cray can optimize based on proven Linux environment
- Some of the features could be enabled (but with a performance cost) at some point in the future.
- Some unsupported features may currently work but this can not be guaranteed in the future.
  - Some may not have worked under catamount but may under CNL
  - Some may cause your code to crash (particularly look for errno)

The Compute Nodes

- You do not have any direct access to the compute nodes
  - Work that requires batch processors needs to be controlled via ALPS (Application Level Placement Scheduler)
  - This has to be done via the command aprun
  - All the ALPS commands begin with ap…
- The batch nodes require access through PBS (which is a new version from that which was used with Catamount)
- Or on the interactive nodes using aprun directly

Cray XT4 programming environment is SIMPLE

- Edit and compile MPI program (no need to specify include files or libraries)

```
$ vi pippo.f
$ ftn –o pippo pippo.f
```

- Edit PBSPro job file (pippo.job)

```
#PBS -N myjob
#PBS -l mppwidth=256
#PBS –l mppnppn=2
#PBS -j oe
cd $PBS_O_WORKDIR
aprun –n 256 –N 2 ./pippo
```

- Run the job (output will be myjob.oxxxxx)

```
$ qsub pippo.job
```

Job Launch

XT4 User

Login PE

SDB Node

Job Launch

XT4 User

Login &

Start App

Login PE

qsub

PBS Pro

SDB Node

Job Launch

XT4 User

Login &

Start App

Login PE

qsub

PBS Pro

Login Shell

aprun

apbasil

SDB Node

apsched

apinit

apshepherd

Application

...

...

Job Launch

XT4 User

Login &

Start App

Login PE

qsub

PBS Pro

Login Shell

aprun

apbasil

SDB Node

apsched

apinit

apshepherd

Application

Application Runs

IO Nodes Implement Request

IO Node
IO daemons

IO Node
IO daemons

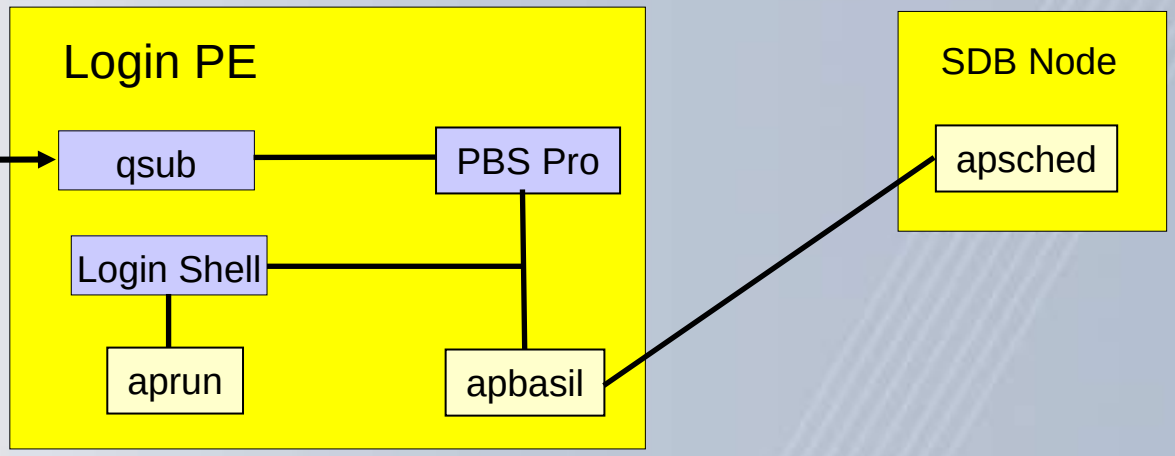
IO Node
IO daemons

Job Launch

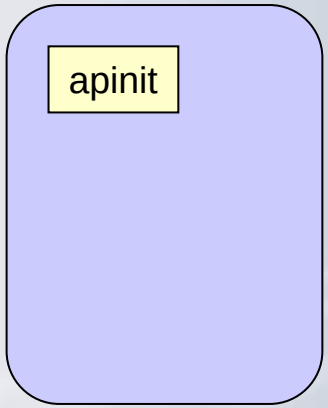
XT4 User

Login &

Start App



Job is cleaned up



Job Launch

XT4 User

Login &

Start App

Login PE

qsub

PBS Pro

Login Shell

aprun

apbasil

Nodes returned

SDB Node

apsched

apinit

Cray XT4 programming environment overview

- PGI compiler suite (the default supported version)
- Pathscale compiler suite
- Optimized libraries:
  - 64 bit AMD Core Math library (ACML): Level 1,2,3 of BLAS, LAPACK, FFT
  - SciLib: Scalapack, BLACS, SuperLU (increasing in functionality)
- MPI-2 message passing library for communication between nodes
  (derived from MPICH-2, implements MPI-2 standard, except for support of dynamic process functions)
- SHMEM one-sided communication library

Cray XT4 programming environment overview

- GNU C library, gcc, g++
- aprun command to launch jobs; similar to mpirun command. There are subtle differences compared to yod, so think of aprun as a new command
- PBSPro batch system
  - needed newer versions to be able to more accurately specify resources in a node, thus there is a significant syntax change
- Performance tools: CrayPat, Apprentice2
- Totalview debugger

The module tool on the Cray XT4

- How can we get appropriate Compiler and Libraries to work with?

- module tool used on XT4 to handle different versions of packages (compiler, tools,...):
  e.g.: module load compiler1
  e.g.: module switch compiler1 compiler2
  e.g.: module load totalview
  .....
- taking care of changing of PATH, MANPATH, LM_LICENSE_FILE,.... environment.
- users should not set those environment variable in his shell startup files, makefiles,....
- keep things flexible to other package versions

- It is also easy to setup your own modules for your own software

Cray XT4 programming environment: module list

```
nid00004> module list
Currently Loaded Modulefiles:
  1) modules/3.1.6          7) xt-pe/2.0.10          13) xt-boot/2.0.10
  2) MySQL/4.0.27           8) PrgEnv-pgi/2.0.10     14) xt-lustre-ss/2.0.10
  3) acml/3.6.1             9) xt-service/2.0.10     15) Base-opts/2.0.10
  4) pgi/7.0.4             10) xt-libc/2.0.10        16) pbs/8.1.1
  5) xt-libsci/10.0.1      11) xt-os/2.0.10          17) gcc/4.1.2
  6) xt-mpt/2.0.10         12) xt-catamount/2.0.10   18) xtpe-target-cnl
```

- Current versions
  - CNL 2.0.10
  - PGI 7.0.4
  - ACML 3.6.1
  - PBS 8.1.1 (Significant update)

Cray XT4 programming environment: module show

```
nid00004> module show pgi
-------------------------------------------------------------------
/opt/modulefiles/pgi/7.0.4:

setenv          PGI_VERSION 7.0
setenv          PGI_PATH /opt/pgi/7.0.4
setenv          PGI /opt/pgi/7.0.4
prepend-path    LM_LICENSE_FILE /opt/pgi/7.0.4/license.dat
prepend-path    PATH /opt/pgi/7.0.4/linux86-64/7.0/bin
prepend-path    MANPATH /opt/pgi/7.0.4/linux86-64/7.0/man
prepend-path    LD_LIBRARY_PATH /opt/pgi/7.0.4/linux86-64/7.0/lib
prepend-path    LD_LIBRARY_PATH /opt/pgi/7.0.4/linux86-64/7.0/libso

-------------------------------------------------------------------
```

Cray XT4 programming environment: module avail

```
nid00004> module avail
------------------------------------------------ /opt/modulefiles -------------------------------------------------
Base-opts/1.5.39                    gmalloc                         xt-lustre-ss/1.5.44
Base-opts/1.5.44                    gnet/2.0.5                      xt-lustre-ss/1.5.45
Base-opts/1.5.45                    iobuf/1.0.2                     xt-lustre-ss/2.0.05
Base-opts/2.0.05                    iobuf/1.0.5(default)            xt-lustre-ss/2.0.10
Base-opts/2.0.10(default)           java/jdk1.5.0_10(default)       xt-mpt/1.5.39
MySQL/4.0.27                        libscifft-pgi/1.0.0(default)    xt-mpt/1.5.44
PrgEnv-gnu/1.5.39                   modules/3.1.6(default)          xt-mpt/1.5.45
PrgEnv-gnu/1.5.44                   papi/3.2.1(default)             xt-mpt/2.0.05
PrgEnv-gnu/1.5.45                   papi/3.5.0C                     xt-mpt/2.0.10
PrgEnv-gnu/2.0.05                   papi/3.5.0C.1                   xt-mpt-gnu/1.5.39
PrgEnv-gnu/2.0.10(default)          papi-cnl/3.5.0C(default)        xt-mpt-gnu/1.5.44
PrgEnv-pathscale/1.5.39             papi-cnl/3.5.0C.1               xt-mpt-gnu/1.5.45
PrgEnv-pathscale/1.5.44             pbs/8.1.1                       xt-mpt-gnu/2.0.05
PrgEnv-pathscale/1.5.45             pgi/6.1.6                       xt-mpt-gnu/2.0.10
PrgEnv-pathscale/2.0.05             pgi/7.0.4(default)              xt-mpt-pathscale/1.5.39
PrgEnv-pathscale/2.0.10(default)    pkg-config/0.15.0               xt-mpt-pathscale/1.5.44
PrgEnv-pgi/1.5.39                   totalview/8.0.1(default)        xt-mpt-pathscale/1.5.45
PrgEnv-pgi/1.5.44                   xt-boot/1.5.39                  xt-mpt-pathscale/2.0.05
PrgEnv-pgi/1.5.45                   xt-boot/1.5.44                  xt-mpt-pathscale/2.0.10
PrgEnv-pgi/2.0.05                   xt-boot/1.5.45                  xt-os/1.5.39
PrgEnv-pgi/2.0.10(default)          xt-boot/2.0.05                  xt-os/1.5.44
acml/3.0                            xt-boot/2.0.10                  xt-os/1.5.45
acml/3.6.1(default)                 xt-catamount/1.5.39             xt-os/2.0.05
acml-gnu/3.0                        xt-catamount/1.5.44             xt-os/2.0.10
acml-large_arrays/3.0               xt-catamount/1.5.45             xt-pbs/5.3.5
acml-mp/3.0                         xt-catamount/2.0.05             xt-pe/1.5.39
apprentice2/3.2(default)            xt-catamount/2.0.10             xt-pe/1.5.44
apprentice2/3.2.1                   xt-crms/1.5.39                  xt-pe/1.5.45
craypat/3.2(default)                xt-crms/1.5.44                  xt-pe/2.0.05
craypat/3.2.3beta                   xt-crms/1.5.45                  xt-pe/2.0.10
dwarf/7.2.0(default)                xt-libc/1.5.39                  xt-service/1.5.39
elf/0.8.6(default)                  xt-libc/1.5.44                  xt-service/1.5.44
fftw/2.1.5(default)                 xt-libc/1.5.45                  xt-service/1.5.45
fftw/3.1.1                          xt-libc/2.0.05                  xt-service/2.0.05
gcc/3.2.3                           xt-libc/2.0.10                  xt-service/2.0.10
gcc/3.3.3                           xt-libsci/1.5.39                xtgdb/1.0.0(default)
gcc/4.1.1                           xt-libsci/1.5.44                xtpe-target-catamount
gcc/4.1.2(default)                  xt-libsci/1.5.45                xtpe-target-cnl
gcc-catamount/3.3                   xt-libsci/10.0.1(default)
```

Useful module commands

- Use profiling
  - module load craypat
- Change PGI compiler version
  - module swap pgi/7.0.4 pgi/6.1.6
- Load GNU environment
  - module swap PrgEnv-pgi PrgEnv-gnu
- Load Pathscale environment
  - module load pathscale
  - module swap PrgEnv-pgi PrgEnv-pathscale

Creating your own Modules

- Modules are incredibly powerful for managing software
  - You can apply them to your own applications and software

```
--------------------------------------------- /opt/modules/3.1.6 ------------------------
modulefiles/modules/dot         modulefiles/modules/module-info modulefiles/modules/null
modulefiles/modules/module-cvs  modulefiles/modules/modules     modulefiles/modules/use.own
```

- If you load the use.own modulefile it looks in your private modules directory for modulefiles (~/privatemodules)
- The contents of the file are very basic and can be developed using the examples from the compilers
- There is also “man modulefiles” which is much more verbose

Compiler Module File as a Template

```
#%Module
#
# pgi module
#
set sys         [uname sysname]
set os          [uname release]

set m [uname machine]
if { $m == "x86_64" } {
   set bits 64
   set plat linux86-64
} else {
   set bits 32
   set plat linux86
}

set PGI_LEVEL 7.0.4
set PGI_CURPATH /opt/pgi/$PGI_LEVEL

setenv PGI_VERSION 7.0
setenv PGI_PATH $PGI_CURPATH
setenv PGI $PGI_CURPATH

# Search for demo license before searching flexlm servers
# prepend-path LM_LICENSE_FILE /opt/pgi/license.dat
prepend-path LM_LICENSE_FILE $PGI_CURPATH/license.dat

set pgidir $PGI_CURPATH/$plat/$env(PGI_VERSION)

prepend-path PATH               $pgidir/bin
prepend-path MANPATH            $pgidir/man
prepend-path LD_LIBRARY_PATH    $pgidir/lib
prepend-path LD_LIBRARY_PATH    $pgidir/libso
```

Compiler drivers to create CNL executables

- When the PrgEnv is loaded the compiler drivers are also loaded
  - By default PGI compiler under compiler drivers
  - the compiler drivers also take care of loading appropriate libraries (-lmpich, -lsci, -lacml, -lpapi)

- Available drivers (also for linking of MPI applications):
  - Fortran 90/95 programs ftn
  - Fortran 77 programs f77
  - C programs cc
  - C++ programs CC

- Cross compiling environment
  - Compiling on a Linux service node
  - Generating an executable for a CNL compute node
  - Do not use pgf90, pgcc unless you want a Linux executable for the service node
  - Information message:

```
ftn: INFO: linux target is being used
```

PGI compiler flags for a first start

Overall Options:

-Mlist	creates a listing file
-Wl,-M	generates a loader map (to stdout)

Preprocessor Options:

-Mpreprocess	runs the preprocessor on Fortran files (default on .F, .F90, or .fpp files)

Optimisation Options:

-fast	chooses generally optimal flags for the target platform
-fastsse	chooses generally optimal flags for a processor that supports the SSE, SSE3 instructions.
-Mipa=fast,inline	Inter Procedural Analysis
-Minline=levels:number	number of levels of inlining

man pgf90, man pgcc, man pgCC
PGI User's Guide (Chapter 2) http://www.pgroup.com/doc/pgiug.pdf
Optimization Presentation

Other programming environments

- GNU
  - **module swap PrgEnv-pgi PrgEnv-gnu**
  - Default compiler is gcc/4.1.1
  - gcc/4.1.2 module available
- Pathscale
  - **module load pathscale**
  - Pathscale version is 3.0
- Using autoconf configure script on the XT4
  - Define compiler variables

    setenv CC cc
    setenv CXX CC
    setenv F90 ftn
  - **--enable-static**
    build only statically linked executables
  - If it is serial code then it *can* be tested on the login node
  - If it is parallel then you will need to launh test jobs with aprun

Using System Calls

- System calls are now available
- They are not quite the same as login node commands
- A number of commands are now available in “BusyBox mode”
  - Busybox is a memory optimized version of the command
- This is different from Catamount where this was not available

Memory Allocation Options

- Catamount malloc
  - Default malloc on Catamount was a custom implementation of the malloc() function tuned to Catamount's non-virtual-memory operating system and favoured applications allocating large, contiguous data arrays.
  - Not always the fastest
- Glibc malloc
  - Could be faster in some cases
- CNL uses Linux features (glibc version)
  - It also has an associated routine to tune performance (mallopt)
  - A default set of options is set when you use –Msmartalloc
- Use –Msmartalloc with care
  - It can grab memory from the OS ready for user mallocs and does not return it to the OS until the job finishes
  - It reduces the memory that can be used for IO buffers and MPI buffers

CNL programming considerations

- there is a name conflict between stdio.h and MPI C++ binding in relation to the names SEEK_SET, SEEK_CUR, SEEK_END

- Solution:
  - your application does not use those names:
    - work with -DMPICH_IGNORE_CXX_SEEK to come around this
  - your application does use those names:

```
#undef SEEK_SET
<include mpi.h>
```

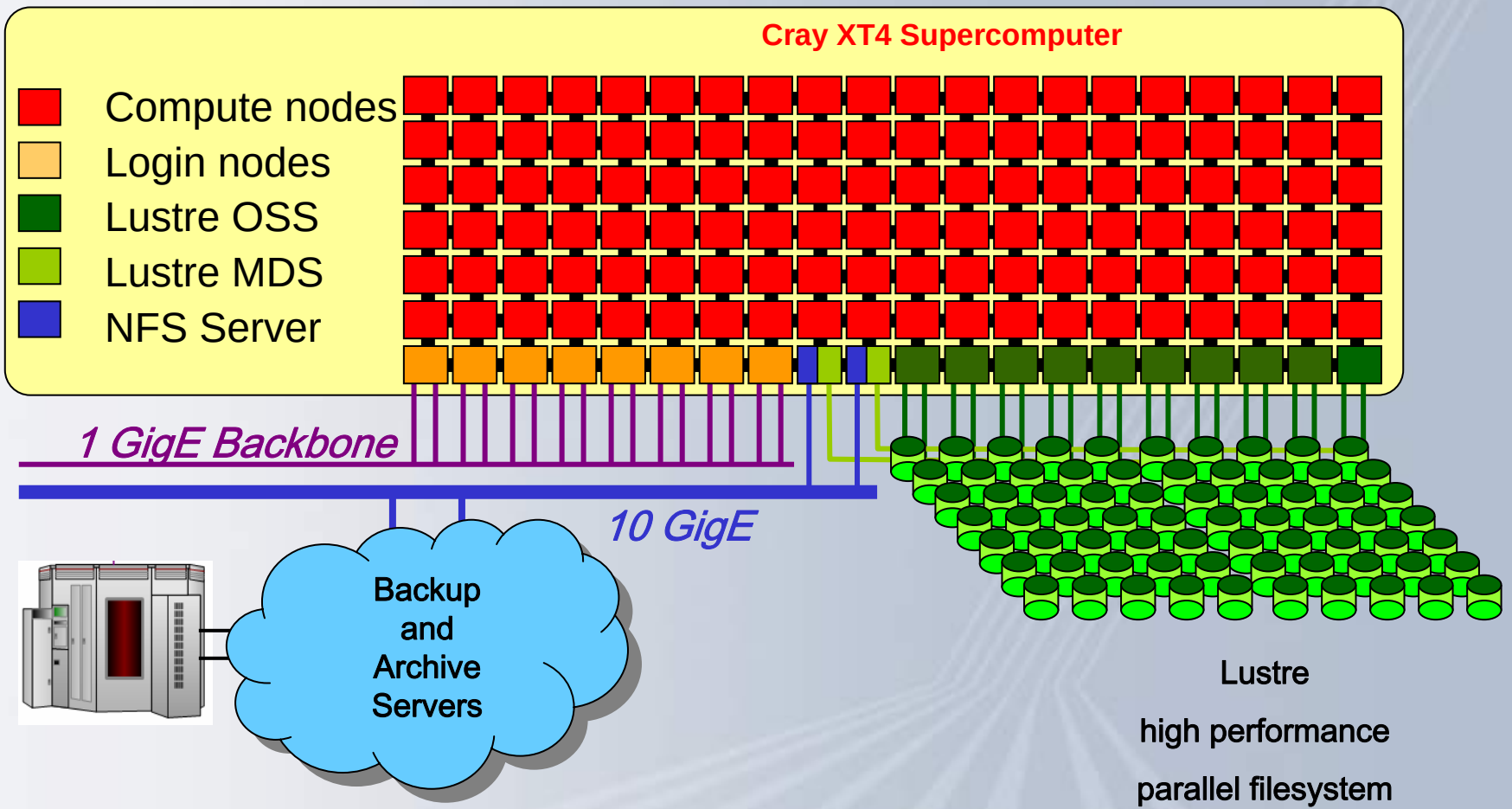
    - or change order of includes: mpi.h before stdio.h or iostream

Timing support in CNL

- CPU time:
  - supported is: getrusage, cpu_time,
  - not supported: times
- Elapsed/wall clock time support:
  - supported: gettimeofday, MPI_Wtime, system_clock, omp_get_wtime
  - not supported: times, clock, dclock, etime

There may be a bit of work to do here as dclock was the recommended timer on Catamount

The Storage Environment

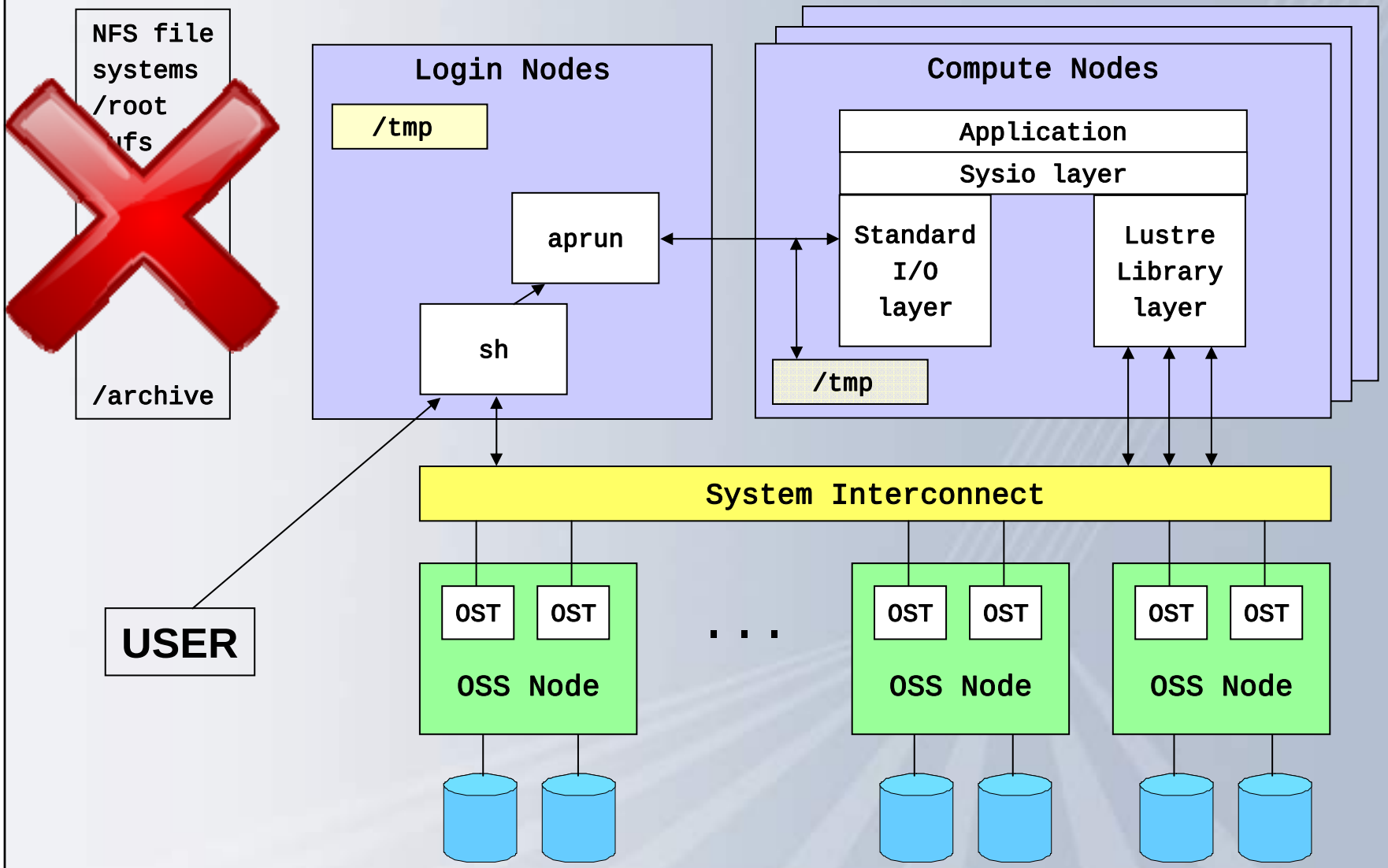


- Cray provides high performance local file system
- Cray enables vendor independent integration for backup and archival

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
- Terminology
  - **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

Cray XT4 I/O architecture



Cray XT4 I/O Architecture Characteristics

- All I/O is offloaded to service nodes
- Lustre – High performance parallel I/O file system
  - Direct data transfer between Compute nodes and files
  - User level library ➔ Relink on software upgrade
- Stdin/Stdout/Stderr goes via ALPS task on the login node
  - Single stdin descriptor ➔ cannot be read in parallel
  - Not defined in any standard
- No local disks on compute nodes,
  - reduces number of moving parts in compute blades
- /tmp is a MEMORY file system, on each node
  - Use $TMPDIR (*) to redirect large files
  - They are different /tmp directories

Cray XT4 I/O Architecture Limitations

- No I/O with named pipes on CNL
- PGI Fortran run-time library
  - Fortran SCRATCH files are not unique per PE
  - No standard exists
- By default stdio is unbuffered (not quite true - at least line buffered)

Lustre File Striping

- Stripes defines the number of OSTs to write the file across
  - Can be set on a per file or directory basis
- CRAY recommends that the default be set to
  - not striping across all OSTs, but
  - set default stripe count of one to four
- But not always the best for application performance.
  As a general rule of thumbs :
  - If you have one large file
    => stripe over all OSTs
  - If you have a large number of files (~2 times #OSTs)
    => turn off striping (#stripes=1)
- Common default
  - Stripe size = 1 MB
  - Stripe count = 2

Lustre lfs command

- **lfs** is a lustre utility that can be used to create a file with a specific striping pattern, displays file striping patterns, and find file locations

- The most used options are :
  - setstripe
  - getstripe
  - df

- For help execute **lfs** without any arguments

```
$ lfs
lfs > help
Available commands are:
          setstripe
          find
          getstripe
          check
      ...
```

lfs setstripe

- Sets the stripe for a file or a directory
- **lfs setstripe <file|dir> <size> <start> <count>**
  - stripe size: Number of bytes on each OST (0 filesystem default)
  - stripe start: OST index of first stripe (-1 filesystem default)
  - stripe count: Number of OSTs to stripe over (0 default, -1 all)
- Comments
  - The stripes of a file is given when the file is created. It is not possible to change it afterwards.
  - If needed, use lfs to create an empty file with the stripes you want (like the touch command)

Lustre striping hints

- For maximum aggregate performance: Keep all OSTs occupied

- Many clients, many files: Don't stripe

  If number of clients and/or number of files >> number of OSTs:

  Better to put each object (file) on only a single OST.

- Many clients, one file: Do stripe

  When multiple processes are all accessing one large file:

  Better to stripe that single file over all of the available OSTs.

- Some clients, few large files: Do stripe

  When a few processes access large files in large chunks:

  Stripe over enough OSTs to keep the OSTs busy on both write and read paths.

lfs getstripe

- Shows the stripe for a file or a directory
- Syntax : **`lfs getstripe <filename|dirname>`**
- Use –verbose option to get stripe size

```
louhi> lfs getstripe --verbose /lus/nid00131/roberto/pippo
OBDS:
0: ost0_UUID ACTIVE
<lines removed>
31: ost31_UUID ACTIVE
/lus/nid00131/roberto/pippo
lmm_magic:          0x0BD10BD0
lmm_object_gr:      0
lmm_object_id:      0x697223e
lmm_stripe_count:   2
lmm_stripe_size:    1048576
lmm_stripe_pattern: 1
        obdidx           objid           objid           group
            14           42575         0xa64f               0
            15           42585         0xa659               0
```

lfs df

- shows the current status of a lustre filesystem

```
kroy@nid00004:~/lustre> lfs df
UUID                 1K-blocks      Used Available  Use% Mounted on
mds1_UUID            249964396  14848316 235116080    5% /work[MDT:0]
ost0_UUID            1922850100 108527440 1814322660    5% /work[OST:0]
ost1_UUID            1922850100 110297980 1812552120    5% /work[OST:1]
ost2_UUID            1922850100 114369912 1808480188    5% /work[OST:2]
ost3_UUID            1922850100 104407112 1818442988    5% /work[OST:3]
ost4_UUID            1922850100 111024884 1811825216    5% /work[OST:4]
ost5_UUID            1922850100 105603904 1817246196    5% /work[OST:5]
ost6_UUID            1922850352 106531460 1816318892    5% /work[OST:6]
ost7_UUID            1922850352 109677076 1813173276    5% /work[OST:7]
ost8_UUID            1922850352 1442137764 480712588   75% /work[OST:8]

filesystem summary:  17305651656 975429728 16330221928    5% /work
```

Artificially increased to show data being prioritised in one ost

IOBUF Library

- IOBUF previously gained great benefit for applications
  - This was as a result of IO initiating a syscall each write statement
  - In CNL it uses Linux buffering
  - IOBUF can still get some performance increases

- IOBUF worked because if you know what you are doing then setting up the correct sized buffers gives great performance. Linux buffering is very sophisticated and gets very good buffering across the board.

I/O hints

- Cray PAT
  - Use Cray PAT options to collect I/O information
  - Select proper buffer size and match it to Lustre striping parameters
- Striping
  - Select the striping according to the I/O pattern
  - Experiment with different solutions
- Performance
  - One single I/O task is limited to about 1 GB/sec
  - Increase I/O tasks if lustre filesystem can sustain more
  - If too many tasks access the filesystem at the same time, the performance per task will drop
  - It might be better to use a few tasks doing the IO (IO Servers).

Running an application on the Cray XT4

- ALPS (aprun) is the XT4 application launcher
  - It must be used to run application on the XT4
  - If aprun is not used, the application is launched on the login node (and likely fails)
- aprun has several parameters and some of them are redundant
  - aprun –n (number of mpi tasks)
  - aprun –N (number of MPI tasks per node)
  - aprun –d (depth of each task – separation)
- aprun supports MPMD
  Launching several executables on the same MPI_COMM_WORLD

```
$ aprun –n 4 –N 2 ./a.out : -n 8 –N 2 ./b.out
```

Running an interactive application

- Only aprun is needed
- The number of required processors must be specified
  - If not, default is to use 1 node

```
$ aprun –n 8 ./a.out
```

- It is possible to specify the processor partition
  - If some node is already used, aprun aborts

```
$ aprun –n 8 –L 152..159 ./a.out
```

- Limited resources

xtprocadmin: tds1 service nodes (8)

```
kroy@nid00004:~> xtprocadmin | grep -e service -e NID ; xtshowcabs
Connected
   NID    (HEX)    NODENAME     TYPE    STATUS        MODE PSLOTS FREE
     0      0x0  c0-0c0s0n0  service        up interactive      4    0
     3      0x3  c0-0c0s0n3  service        up interactive      4    0
     4      0x4  c0-0c0s1n0  service        up interactive      4    4
     7      0x7  c0-0c0s1n3  service        up interactive      4    0
    32     0x20  c0-0c1s0n0  service        up interactive      4    4
    35     0x23  c0-0c1s0n3  service        up interactive      4    0
    36     0x24  c0-0c1s1n0  service        up interactive      4    0
    39     0x27  c0-0c1s1n3  service        up interactive      4    0

Compute Processor Allocation Status as of Mon Aug 13 11:33:58 2007

     C0-0
  n3 --------
  n2 --------
  n1 --------
c2n0 --------
  n3 SS------
  n2   ------
  n1   ------
c1n0 SS------
  n3 SS;;;;--
  n2   ;;;;--
  n1   ;;;;--
c0n0 SS;;;;--
    s01234567
```

xtprocadmin: tds1 interactive nodes (8)

```
kroy@nid00004:~> xtprocadmin | grep -e interactive -e NID | grep -e compute -e
   NID
Connected
   NID    (HEX)    NODENAME     TYPE    STATUS        MODE PSLOTS FREE
     8      0x8  c0-0c0s2n0  compute        up interactive      4    4
     9      0x9  c0-0c0s2n1  compute        up interactive      4    4
    10      0xa  c0-0c0s2n2  compute        up interactive      4    4
    11      0xb  c0-0c0s2n3  compute        up interactive      4    4
    12      0xc  c0-0c0s3n0  compute        up interactive      4    4
    13      0xd  c0-0c0s3n1  compute        up interactive      4    4
    14      0xe  c0-0c0s3n2  compute        up interactive      4    4
    15      0xf  c0-0c0s3n3  compute        up interactive      4    4
    16     0x10  c0-0c0s4n0  compute        up interactive      4    4
    17     0x11  c0-0c0s4n1  compute        up interactive      4    4
    18     0x12  c0-0c0s4n2  compute        up interactive      4    4
    19     0x13  c0-0c0s4n3  compute        up interactive      4    4
    20     0x14  c0-0c0s5n0  compute        up interactive      4    4
    21     0x15  c0-0c0s5n1  compute        up interactive      4    4
    22     0x16  c0-0c0s5n2  compute        up interactive      4    4
    23     0x17  c0-0c0s5n3  compute        up interactive      4    4
```

xtshowcabs: tds1 interactive node locations

```
kroy@nid00004:~> xtshowcabs
Compute Processor Allocation Status as of Mon Aug 13 11:40:46 2007
     C0-0
  n3 --------
  n2 --------
  n1 --------
c2n0 --------
  n3 SS------
  n2   ------
  n1   ------
c1n0 SS------
  n3 SS;;;;--
  n2   ;;;;--
  n1   ;;;;--
c0n0 SS;;;;--
    s01234567

Legend:
   nonexistent node                S  service node
;  free interactive compute CNL    -  free batch compute node CNL
A  allocated, but idle compute node ?  suspect compute node
X  down compute node               Y  down or admindown service node
Z  admindown compute node          R  node is routing
Available compute nodes:      16 interactive,    64 batch
```

Remember that the number of nodes in a service blade is less than those in compute blades, this is why there are gaps.

xtshowcabs: tds1 Showing CPA Reservations

```
kroy@nid00004:~> xtshowcabs
Compute Processor Allocation Status as of Mon Aug 13 11:44:37 2007
     C0-0
  n3 aaaa----
  n2 aaaa----
  n1 aaaa----
c2n0 aaaa----
  n3 SS--aaaa
  n2   --aaaa
  n1   --aaaa
c1n0 SS--aaaa
  n3 SSAA;;--
  n2   AA;;--
  n1   AAA;--
c0n0 SSAAA;--
    s01234567

Legend:
   nonexistent node                S  service node
;  free interactive compute CNL    -  free batch compute node CNL
A  allocated, but idle compute node ?  suspect compute node
X  down compute node               Y  down or admindown service node
Z  admindown compute node          R  node is routing

Available compute nodes:       6 interactive,    32 batch
ALPS JOBS LAUNCHED ON COMPUTE NODES
Job ID     User       Size   Age              command line
--- ------ --------   -----  ---------------  ---------------------------------
 a  4726   mfoster       32  0h02m            funky.exe
```

Running a batch application

- PBSPro is the batch environment
- The number of required MPI processes must be specified in the job file

```
#PBS -l mppwidth=256
```

- The number of processes per node also needs to be specified

```
#PBS -l mppnppn=2
```

- It is NOT possible to specify the processor partition. The partition is determined by PBS-CPA interaction and given to aprun.
- The job is submitted by the qsub command
- At the end of the exection output and error files are returned to submission directory

Single-core vs Dual-core

- aprun -N 1|2

  -N 1 single core

  -N 2 Virtual Node: 2 cores in the node

- Default is site dependent:

SINGLE CORE

```
#PBS -N SCjob
#PBS -l mppwidth=256
#PBS –l mppnppn=1
#PBS -j oe
#PBS –l mppdepth=2
…
aprun –n 256 –N 1 pippo
```

DUAL CORE

```
#PBS -N DCjob
#PBS -l mppwidth=256
#PBS –l mppnppn=2
#PBS -j oe
…
aprun –n 256 –N 2 pippo
```

PBSPro parameters

- #PBS -N <job_name>
  - the job name is used to determine the name of job output and error files
- #PBS -l walltime=<hh:mm:ss>
  - Maximum job elapsed time should be indicated whenever possible: this allows PBS to determine best scheduling startegy
- #PBS -j oe
  - job error and output files are merged in a single file
- #PBS -q <queue>
  - request execution on a specific queue: usually not needed
- #PBS –A <project>
  - Specifies the account you wish to run the job under

Useful PBSPro environment variables

- At job startup some environment variables are defined for the PBS application

- $PBS_O_WORKDIR
  - Defined as the directory from which the job has been submitted
- $PBS_ENVIRONMENT
  - PBS_INTERACTIVE, PBS_BATCH
- $PBS_JOBID
  - Job Identifier

aprun: specifying the number of processors

- Question: what happens submitting the following PBSPro job ?

```
#PBS -N hog
#PBS -l nodes=256
#PBS -j oe

cd $PBS_O_WORKDIR
aprun –n 8 ./pippo
```

- First of all we‘re using PBS 5.3 syntax, so it won‘t even submit properly!
- Secondly we‘re wasting resources we‘ve asked for 256 yet only used 8.
  - you generate a lot of **A allocated, but idle** compute nodes

aprun: memory size issues

- -m <size>
  - Specifies the per processing element maximum Resident Set Size memory limit in megabytes.
  - If a program overruns the stack allocation, behavior is undefined.

- When a dual core compute node job is launched they both compete for the memory.
- Once its gone that is it!
  - No paging
- One core can access all the memory

aprun: page sizes

- Catamount and Linux handle differently the way memory gets mapped
  - Catamount always attempts to use 2 MB mappings, but could be swapped to use smaller pages
  - Linux always uses 4 KB mappings.

- Catamount specific TLB pages policy
  - Intended to minimize TLB trashing by specifying large 2MB pages
  - Unfortunately Opteron has only 8 2MB pages (16 MB reach)
  - Opteron has 512 entries for 4 KB mappings (2MB reach)

- CNL currently has no option to do this so there is only the default method which uses the same method as the fast version of Catamount.

Catamount could gain huge performance increases using `yod -small_pages` but this is no longer necessary.

For those codes which gained benefit from large pages this it is not possible to use them.

Monitoring aprun on the Cray XT4 – PBS job

- PBS qstat command
- qstat –r
  - check running jobs
- qstat –n
  - check running and queued jobs
- qstat –s <job_id>
  - reports also comments provided by the batch administrator or scheduler
- qstat –f <job_id>
  - Returns the information on your job, this can be used to pull out all the information on the job.
- This only monitors the state of the batch request not the actual code itself.

PBSPro: qstat -r

```
                                                  Time In Req'd  Req'd   Elap
Job ID Username Queue    Jobname    SessID  Queue  Nodes  Time  S Time
------ -------- -------- ---------- ------ ------- ------ ----- - -----
45083  mluscher normal   run3c_14    32304  032:05     64 10:00 R 04:50
45168  hasen    normal   fluctP      28243  022:27     64 12:00 R 10:34
45169  hasen    normal   fluctR      21979  022:26     64 12:00 R 09:42
45281  hasen    normal   fluctC      29550  010:02     64 12:00 R 08:33
45295  ymantz   normal   sim_ann     25352  009:02     64 12:00 R 04:08
45297  ymantz   normal   sim_ann       141  008:49     64 12:00 R 04:02
45302  urakawa  normal   RuH2_2CO2i  26288  008:28     64 12:00 R 02:56
45310  tkuehne  normal   Silica_QS   22859  008:11     24 12:00 R 08:10
……
45414  flankas  normal   mic_10ps_4  27471  001:01      4 02:00 R 01:00
45416  ballabio lm       lm_f-8-16-   2856  000:35    132 08:00 R 00:34

Total generic compute nodes allocated: 795
```

Monitoring a job on the Cray XT4 – aprun

- xtshowcabs
  - Shows XT4 nodes allocation and aprun processes
- xtshowcabs –j
  - Shows only running ALPS requests
  - Both interactive and PBS
- xtps –Y
  - Similar to xtshowcabs -j

xtshowcabs -j

```
YODS LAUNCHED ON CATAMOUNT NODES
Job ID     User        Size   Start              yod command line and argu
--- ------ --------    -----  ---------------    ------------------------
 y  70380  hasen          64  Feb  8 07:36:59    yod -sz 64 ./full_qcd
 n  70394  hasen          64  Feb  8 08:28:08    yod -sz 64 ./full_qcd
 B  70421  hasen          64  Feb  8 09:37:10    yod -sz 64 ./full_qcd
 G  70500  tkuehne        24  Feb  8 10:00:14    yod -sz 24 /nfs/xt3-homes/
 q  70561  broqvist       32  Feb  8 11:48:54    yod -sz 32 /users/broqvist
 w  70594  mluscher       64  Feb  8 13:20:22    yod -sz 64 run3c -i run3c.
 b  70596  tthoenen        1  Feb  8 13:31:02    yod -size 1 /users/tthoene
 g  70605  hasen          64  Feb  8 13:56:21    yod -sz 64 ./full_qcd
 i  70609  ymantz         64  Feb  8 14:03:07    yod -size 64 ../RUN/cp2k.
 D  70612  ymantz         64  Feb  8 14:09:02    yod -size 64 ../RUN/cp2k.
 x  70635  praiteri        8  Feb  8 14:34:11    yod -size 8 /users/praite
 v  70752  knechtli        1  Feb  8 16:56:02    yod /nfs/xt3-homes/users/
```

xtps -Y

```
NID          PID      USER     START                  CMD
136         29038    broqvist2006-02-13 08:11:20   yod -sz 64 /users/broq
136         30691    hasen    2006-02-13 11:41:29   yod -sz 64 ./full_qcd
136         31803    ymantz   2006-02-13 13:56:39   yod -size 64 ../RUN/cp2k
136         28300    ymantz   2006-02-13 06:22:06   yod -size 96 ../RUN/cp2k
136         29292    marci    2006-02-13 08:26:25   yod -size 64 cpmd.x /scr
136         30331    sebast   2006-02-13 10:51:34   yod -sz 9 /lus/nid00140/
136         28323    urakawa  2006-02-13 06:22:09   yod -sz 32 /apps/cpmd/b
136         30307    tkuehne  2006-02-13 10:51:32   yod -sz 24 /nfs/xt3-home
```

Which processors am I using ?

- CPA allocation strategy
- xtshowcabs tutorial
- XT3 flat performance machine

xtshowcabs

```
      C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
   n3 iiiXiiii onqqoggg ||||||||| wwwyyyyy nnnBBBBB DDDDDDxB CCCzzzzz GGGGGGww
   n2 iiiiiiii inqqoggg ||||||||| wwwyyyyy nnnzBBBB |DDDDDxB CCCCzzzz GGGGGGww
   n1 iiiiiiii nnnrqogg ||||||||| wwwyyyyy nnnnBBBB |DDDDDxB CCCCzzzz GGGGGGww
 c2n0 iiiiiiii npqqqogg ||||||||| wwwxyyyy nnnnBBBB |DDDDDDx BCCCzzzz wGGGGGGX
   n3 aegggiii iiiiiinn ggsitv|| q||||ppw nnnnnnnn ||||||||| yyyBBBBB zzzzzBFw
   n2 adgggiii kiiiimnn gggsout| qw|||npw nnnnnnnn ||||||||| yyyBBBBB zzzzzBwF
   n1 acgggiii jiiilinn gggsott| qq|||||w nnnnnnnn ||||||||| yyyyBBBB zzzzzBwF
 c1n0 abfgghii iiiiiinn gggsott| qq|||||q nnnnnnnn ||||||||| yyyyBBBB zzzzzzwF
   n3 SSSSSSS: SSSSSS:: gggggggg |||||qqq yyyyyyyn qpppCBB| BBBwwwwy xEzzzzzz
   n2       :       :: gggggggg |||||qqq yyyyyyyn qqpppBB| BBBwwwwy xEzzzzzz
   n1       :       :: gggggggg ||||||qq yyyyyyyn qqpppBB| BBBBwwww xEzzzzzz
 c0n0 SSSSSSS: SSSSSS:: gggggggg ||||||qq yyyyyyyy qqpppCBB BBBBwwww zEzzzzzz
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567
```

Cabinet 3

nodename: c3

xtshowcabs

```
      C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
  n3 iiiXiiii onqqoggg ||||||||| wwwyyyyy nnnBBBBB DDDDDDxB CCCzzzzz GGGGGGww
  n2 iiiiiiii inqqoggg ||||||||| wwwyyyyy nnnzBBBB |DDDDDxB CCCCzzzz GGGGGGww
  n1 iiiiiiii nnnrqogg ||||||||| wwwyyyyy nnnnBBBB |DDDDDxB CCCCzzzz GGGGGGww
c2n0 iiiiiiii npqqqogg ||||||||| wwwxyyyy nnnnBBBB |DDDDDDx BCCCzzzz wGGGGGGX
  n3 aegggiii iiiiiinn ggsitv|| q||||ppw nnnnnnnn |||||||| yyyBBBBB zzzzzBFw
  n2 adgggiii kiiiimnn gggsout| qw|||npw nnnnnnnn |||||||| yyyBBBBB zzzzzBwF
  n1 acgggiii jiiilinn gggsott| qq|||||w nnnnnnnn |||||||| yyyyBBBB zzzzzBwF
c1n0 abfgghii iiiiiinn gggsott| qq|||||q nnnnnnnn |||||||| yyyyBBBB zzzzzzwF
  n3 SSSSSSS: SSSSSS:: gggggggg |||||qqq yyyyyyyn qpppCBB| BBBwwwwy xEzzzzzz
  n2       :       :: gggggggg |||||qqq yyyyyyyn qqpppBB| BBBwwwwy xEzzzzzz
  n1       :       :: gggggggg ||||||qq yyyyyyyn qqpppBB| BBBBwwww xEzzzzzz
c0n0 SSSSSSS: SSSSSS:: gggggggg ||||||qq yyyyyyyy qqpppCBB BBBBwwww zEzzzzzz
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567
```

Cabinet 3, chassis 1

nodename: c3-0c1

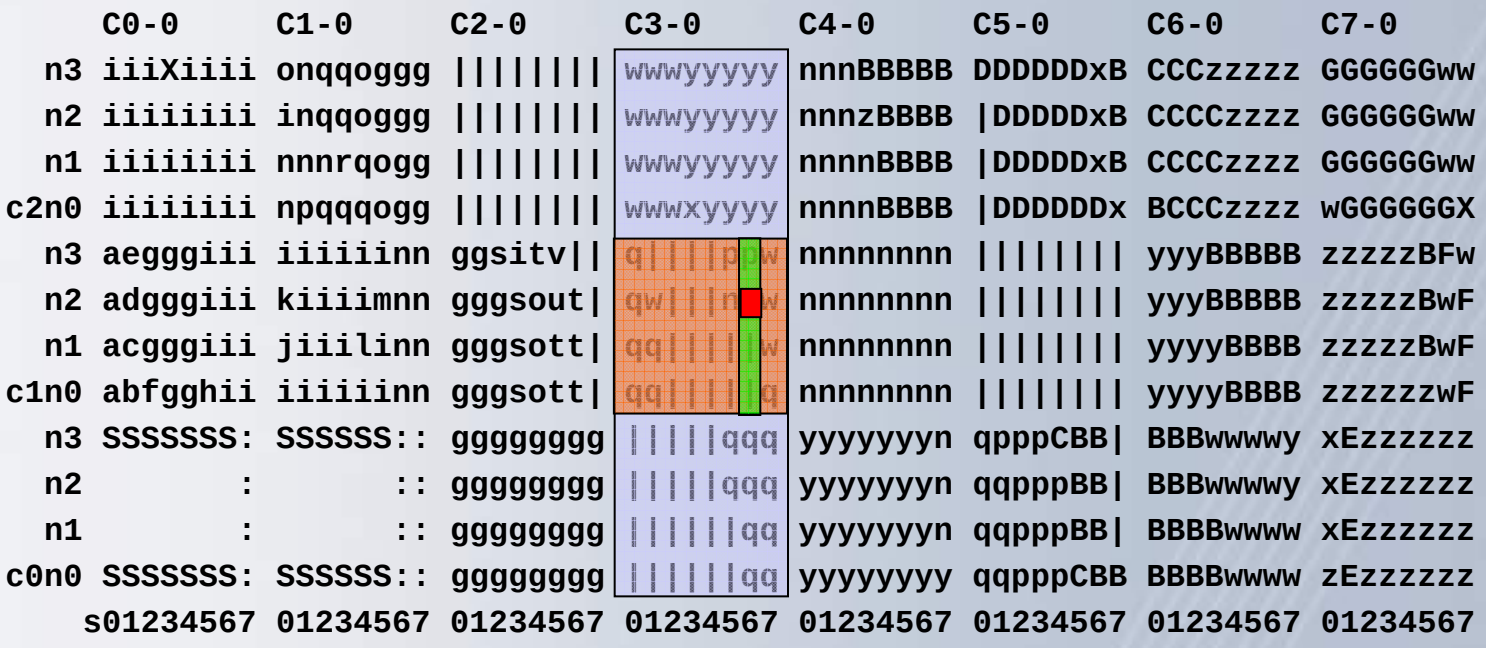
xtshowcabs

```
      C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
  n3 iiiXiiii onqqoggg ||||||||  wwwyyyyy nnnBBBBB DDDDDDxB CCCzzzzz GGGGGGww
  n2 iiiiiiii inqqoggg ||||||||  wwwyyyyy nnnzBBBB |DDDDDxB CCCCzzzz GGGGGGww
  n1 iiiiiiii nnnrqogg ||||||||  wwwyyyyy nnnnBBBB |DDDDDxB CCCCzzzz GGGGGGww
c2n0 iiiiiiii npqqqogg ||||||||  wwwxyyyy nnnnBBBB |DDDDDDx BCCCzzzz wGGGGGGX
  n3 aegggiii iiiiiinn ggsitv||  q||||pqw nnnnnnnn |||||||| yyyBBBBB zzzzzBFw
  n2 adgggiii kiiiimnn gggsout|  qw|||nqw nnnnnnnn |||||||| yyyBBBBB zzzzzBwF
  n1 acgggiii jiiilinn gggsott|  qq||||w nnnnnnnn |||||||| yyyyBBBB zzzzzBwF
c1n0 abfgghii iiiiiinn gggsott|  qq||||q nnnnnnnn |||||||| yyyyBBBB zzzzzzwF
  n3 SSSSSSS: SSSSSS:: gggggggg  |||||qqq yyyyyyyn qpppCBB| BBBwwwwy xEzzzzzz
  n2        :       :: gggggggg  |||||qqq yyyyyyyn qqpppBB| BBBwwwwy xEzzzzzz
  n1        :       :: gggggggg  ||||||qq yyyyyyyn qqpppBB| BBBBwwww xEzzzzzz
c0n0 SSSSSSS: SSSSSS:: gggggggg  ||||||qq yyyyyyyy qqpppCBB BBBBwwww zEzzzzzz
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567
```

Cabinet 3, chassis 1, slot 6

nodename: c3-0c1s6

xtshowcabs



Cabinet 3, chassis 1, slot 6, node 2

nodename: c3-0c1s6n2

nid: 442 (0x1ba)

xtshowcabs: service nodes

```
     C0-0     C0-1     C1-0     C1-1     C2-0     C2-1     C3-0     C3-1
  n3 bbbbeeee aacccccc iihhiihc bbbbbjjb cccccccc ooooolll ddnnlnnn dnnnnlll
  n2 bbbbeeee aacccccc iihhiihc bbbbbbjj cccccccc ooooolld ddnnlknn dnnnnlll
  n1 bbbbbeee aacccccc gihhiihh bbbbbjjj cccccccc ooooolll dddnllnn ddnnnnll
c2n0 bbbbbeee aacccccc hiihhiih bbbbbjjj cccccccc ooooooll dddnnlnn ddnnnnll
  n3 bddddbcc gggggggа gggghhhh gggfgffb ccdddddc bboooooo dddddddd nggpiddd
  n2 bbddddcc gggggggа gggghhhh gggggggb ccdddddc bboooooo dddddddd ndgphida
  n1 bbddddcc gggggggа gggghhhh gggggggj cccddddc bboooooo dddddddd nngghidn
c1n0 bcddddcc gggggggа gggggghh gggggggj cccddddc bboooooo dddddddd nngghidd
  n3 SSSSSSSb eeeeffbg SSSSSScc ccccccccg jmmbmmbb ccnnnndd dddddddd SSnnnnnn
  n2        b eeeefffg       cc cccfcccg jmmbbmbb ccnnnnnd dddddddd   nnnnnn
  n1        b eeeeeffg       cc cccccccc jlmmbmbb cccnnnnd dddddddd   nnnnnn
c0n0 SSSSSSSa eeeeeffg SSSSSScc cccccccc jkmmmmmm cccnnnnd dddddddd SSnnnnnn
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567

Legend:
   nonexistent node                 S  service node
:  free interactive compute node    A  allocated, but idle compute node
|  free batch compute node          ?  suspect compute node
X  down compute node                Y  down or admindown service node
Z  admindown compute node           R  node is routing
```

xtshowcabs: free batch nodes

```
     C4-0     C4-1     C5-0     C5-1     C6-0     C6-1     C7-0     C7-1
  n3 lppppppp pppppppp ppnnllll iiiiiiii llllllqq ssssssss bbuvvvvw BBB|
  n2 lppppppp pppppppp ppnnllll iiiiiiii llllllqq ssssssss bbuuvvvv BBBB
  n1 llppppp pppppppp ppinllll iiiiiiii llllllqq ssssssss ubuuvvvv BBBB
c2n0 llppppp pppppppp ppinnlll iiiiiiii llllllqq ssssssss bbbuvvvv BBBB
  n3 laallllll pppppppp pppppppp iiiiiiii ggnnnnll ggssssss tttuguub yyyzzzzB
  n2 llallllll pppppppp pppppppp iiiiiiii ggnnnnll ggssssss ttttguub yyyyzzzz
  n1 llallllll pppppppp pppppppp iiiiiiii ggnnnnnl ggssssss ttttguub yyyyzzzz
c1n0 llaallll pppppppp pppppppp iiiiiiii gglnnnnl ggssssss ttttgguu yyyyzzzz
  n3 lllllaal Sppppppp pppppppp nniiiiii iiiigggg qqrrgggg sssskkkt wwwxxxxy
  n2 llllllaa  ppppppp pppppppp nniiiiii iiiiiggg qqrrgggg sssskkk| wwwwxxxx
  n1 llllllaa  ppppppp pppppppp nniiiiii iiiiiggg qqgrggrg sssskkkk wwwwxxxx
c0n0 llllllaa Sppppppp pppppppp nniiiiii iiiiiggg qqgrrggg sssskkkk wwwwxxxx
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567
Legend:
   nonexistent node                S  service node
:  free interactive compute node   A  allocated, but idle compute node
|  free batch compute node         ?  suspect compute node
X  down compute node               Y  down or admindown service node
Z  admindown compute node          R  node is routing
```

xtshowcabs: down compute nodes

```
      C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
   n3 aaaaaaaa dddddddd ggghhhhh hhhhhiii hhhhhhhh iiiiiiii iiiiiiii iiiiiiii
   n2 aaaaaaaa dddddddd ggghhhhh hhhhhiii hhhhhhhh iiiiiiii iiiiiiii iiiiiiii
   n1 aaaaaaaa dddddddd ggghhhhh hhhhhiii hhhhhhhh iiiiiiii iiiiiiii iiiiiiii
 c2n0 aaaaaaaa dddddddd ggghhhhh hhhhhiii hhhhhhhh iiiiiiii iiiiiiii iiiiiiii
   n3 ::aaaaba aaaacccc ffffffff hhhhhhhh hhhhhhhh hhhhhhii iiiiiiii iiiiiiii
   n2 ::aaaaba aaaacccc ffffffff hhhhhhhh hhhhhhhh hhhhhhii iiiiiiii iiiiiiii
   n1 ::aaaaba aaaacccc ffffffff hhhhhhhh hhhhhhhh hhhhhhii iiiiiiii iiiiiiii
 c1n0 ::aaaaba aaaacccc ffffffff hhhhhhhh hhhhhhhh hhhhhhii iiiiiiii iiiiiiii
   n3 SSSSSS:: SSSSSaaa ddeeeeef hhhhhhhh iiihhhhh hhhhhhhh iiiiiiii iiiiiiii
   n2       ::      aaa ddeeeeef hhhhhhhh iiihhhhh hhhhhhhh iiiiiiii iiiiiiii
   n1       ::      aaa ddeeeeef hhhhhhhh iiihhhhh hhhhhhhh iiiiiiii iiiiiiii
 c0n0 SSSSSS:: SSSSSaaa ddeeeeef hhhhhhhh iiihhhhh hhhhhhhh iiiiiiii iiiiiiii
     s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567

      C8-0     C9-0     C10-0
   n3 jjjjjjjf kkk|||||  ||||||||
   n2 jjjjjjjf kkk|||||  ||||||||
   n1 jjjjjjjf kkk|||||  ||||||||
 c2n0 jjjjjjjf kkk|||||  ||||||||
   n3 jjjjjjjj |||||||k ||||||||
   n2 jjjjjjjj |||||||k ||||||||
   n1 jjjjjjjj |||||||k ||||||||
 c1n0 jjjjjjjj |||||||k ||||||||
   n3 hhhggggj fffffff| ||||||||
   n2 hhhggggj fffffff| ||||||||
   n1 hhhggggj fffffff| ||||||||
 c0n0 hhhggggj fffffff| ||||||||
     s01234567 01234567 01234567
```

Sorry, could not find any of them!

Legend:

X down compute node

Y down or admindown service node

Z admindown compute node

R node is routing

CPA allocation algorithm

- CPA gets the first available compute processors, scanning the processor list sequentially by NID
- NID sequence has no relationship with XT4 topology

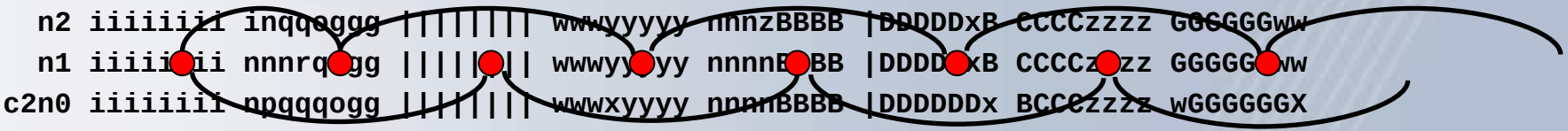
```
$ xtprocadmin | grep compute| grep batch| grep up| grep '4$' | head -10

   206     0xce  c1-0c2s3n2  compute        up       batch      4    4
   207     0xcf  c1-0c2s3n3  compute        up       batch      4    4
   208     0xd0  c1-0c2s4n0  compute        up       batch      4    4
   209     0xd1  c1-0c2s4n1  compute        up       batch      4    4
   210     0xd2  c1-0c2s4n2  compute        up       batch      4    4
   211     0xd3  c1-0c2s4n3  compute        up       batch      4    4
   212     0xd4  c1-0c2s5n0  compute        up       batch      4    4
   213     0xd5  c1-0c2s5n1  compute        up       batch      4    4
   214     0xd6  c1-0c2s5n2  compute        up       batch      4    4
   215     0xd7  c1-0c2s5n3  compute        up       batch      4    4
```

Processor allocation to applications

```
     C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
  n3 iiiXiiii onqqoggg |||||||| wwwyyyyy nnnBBBBB DDDDDDxB CCCzzzzz GGGGGGww
  n2 iiiiiiii inqqoggg |||||||| wwwyyyyy nnnzBBBB |DDDDDxB CCCCzzzz GGGGGGww
  n1 iiiiiiii nnnrqogg |||||||| wwwyyyyy nnnnBBBB |DDDDDxB CCCCzzzz GGGGGGww
c2n0 iiiiiiii npqqqogg |||||||| wwwxyyyy nnnnBBBB |DDDDDDx BCCCzzzz wGGGGGGX
  n3 aeggiiii iiiiiinn ggsitv|| q||||ppw nnnnnnnn |||||||| yyyBBBBB zzzzzBFw
  n2 adgggiii kiiiimnn gggsout| qw|||npw nnnnnnnn |||||||| yyyBBBBB zzzzzBwF
  n1 acgggiii jiiilinn gggsott| qq|||||w nnnnnnnn |||||||| yyyyBBBB zzzzzBwF
c1n0 abfgghii iiiiiinn gggsott| qq|||||q nnnnnnnn |||||||| yyyyBBBB zzzzzzwF
  n3 SSSSSSS: SSSSSS:: gggggggg |||||qqq yyyyyyyn qpppCBB| BBBwwwwy xEzzzzzz
  n2       :       :: gggggggg |||||qqq yyyyyyyn qqpppBB| BBBwwwwy xEzzzzzz
  n1       :       :: gggggggg ||||||qq yyyyyyyn qqpppBB| BBBBwwww xEzzzzzz
c0n0 SSSSSSS: SSSSSS:: gggggggg ||||||qq yyyyyyyy qqpppCBB BBBBwwww zEzzzzzz
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567
```

```
YODS LAUNCHED ON CATAMOUNT NODES
Job ID     User       Size   Start            yod command line and arguments
--- ------ --------   -----  --------------   ------------------------------
 i  70609  ymantz        64  Feb  8 14:03:07  yod -size 64 ../RUN/cp2k.popt
```

X dimension links



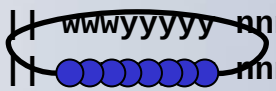
```
     C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
  n3 iiiXiiii onqqoggg |||||||| wwwyyyyy nnnBBBBB DDDDDDxB CCCzzzzz GGGGGGww
  n2 iiiiiiii inqqoggg |||||||| wwwyyyyy nnnzBBBB |DDDDDxB CCCCzzzz GGGGGGww
  n1 iiiiiiii nnnrqogg |||||||| wwwyyyyy nnnnBBBB |DDDDDxB CCCCzzzz GGGGGGww
c2n0 iiiiiiii npqqqogg |||||||| wwwxyyyy nnnnBBBB |DDDDDDx BCCCzzzz wGGGGGGX
  n3 aegggiii iiiiiinn ggsitv|| q||||ppw nnnnnnnn |||||||| yyyBBBBB zzzzzBFw
  n2 adgggiii kiiiimnn gggsout| qw|||npw nnnnnnnn |||||||| yyyBBBBB zzzzzBwF
  n1 acgggiii jiiilinn gggsott| qq|||||w nnnnnnnn |||||||| yyyyBBBB zzzzzBwF
c1n0 abfgghii iiiiiinn gggsott| qq|||||q nnnnnnnn |||||||| yyyyBBBB zzzzzzwF
  n3 SSSSSSS: SSSSSS:: gggggggg |||||qqq yyyyyyyn qpppCBB| BBBwwwwy xEzzzzzz
  n2        :       :: gggggggg |||||qqq yyyyyyyn qqpppBB| BBBwwwwy xEzzzzzz
  n1        :       :: gggggggg ||||||qq yyyyyyyn qqpppBB| BBBBwwww xEzzzzzz
c0n0 SSSSSSS: SSSSSS:: gggggggg ||||||qq yyyyyyyy qqpppCBB BBBBwwww zEzzzzzz
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567

Legend:
   nonexistent node                 S  service node
:  free interactive compute node    A  allocated, but idle compute node
|  free batch compute node          ?  suspect compute node
X  down compute node                Y  down or admindown service node
Z  admindown compute node           R  node is routing
```

Y dimension links

```
     C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
  n3 iiiXiiii onqqoggg |||||||| wwwyyyyy nnnBBBBB DDDDDDxB CCCzzzzz GGGGGGww
  n2 iiiiiiii inqqoggg |||||||| wwwyyyyy nnnzBBBB |DDDDDxB CCCCzzzz GGGGGGww
  n1 iiiiiiii nnnrqogg |||||||| wwwyyyyy nnnnBBBB |DDDDDxB CCCCzzzz GGGGGGww
c2n0 iiiiiiii npqqqogg |||||||| wwwxyyyy nnnnBBBB |DDDDDDx BCCCzzzz wGGGGGGX
  n3 aegggiii iiiiiinn ggsitv|| q||||pqw nnnnnnnn |||||||| yyyBBBBB zzzzzBFw
  n2 adgggiii kiiiimnn gggsout| qw||||qw nnnnnnnn |||||||| yyyBBBBB zzzzzBwF
  n1 acgggiii jiiilinn gggsott| qq||||qw nnnnnnnn |||||||| yyyyBBBB zzzzzBwF
c1n0 abfgghii iiiiiinn gggsott| qq||||qq nnnnnnnn |||||||| yyyyBBBB zzzzzzwF
  n3 SSSSSSS: SSSSSS:: gggggggg |||| qqq yyyyyyyn qpppCBB| BBBwwwwy xEzzzzzz
  n2        :       :: gggggggg ||||qqqq yyyyyyyn qqpppBB| BBBwwwwy xEzzzzzz
  n1        :       :: gggggggg ||||qqqq yyyyyyyn qqpppBB| BBBBwwww xEzzzzzz
c0n0 SSSSSSS: SSSSSS:: gggggggg ||||qqqq yyyyyyyy qqpppCBB BBBBwwww zEzzzzzz
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567
```

```
Legend:
   nonexistent node                 S  service node
:  free interactive compute node    A  allocated, but idle compute node
|  free batch compute node          ?  suspect compute node
X  down compute node                Y  down or admindown service node
Z  admindown compute node           R  node is routing
```

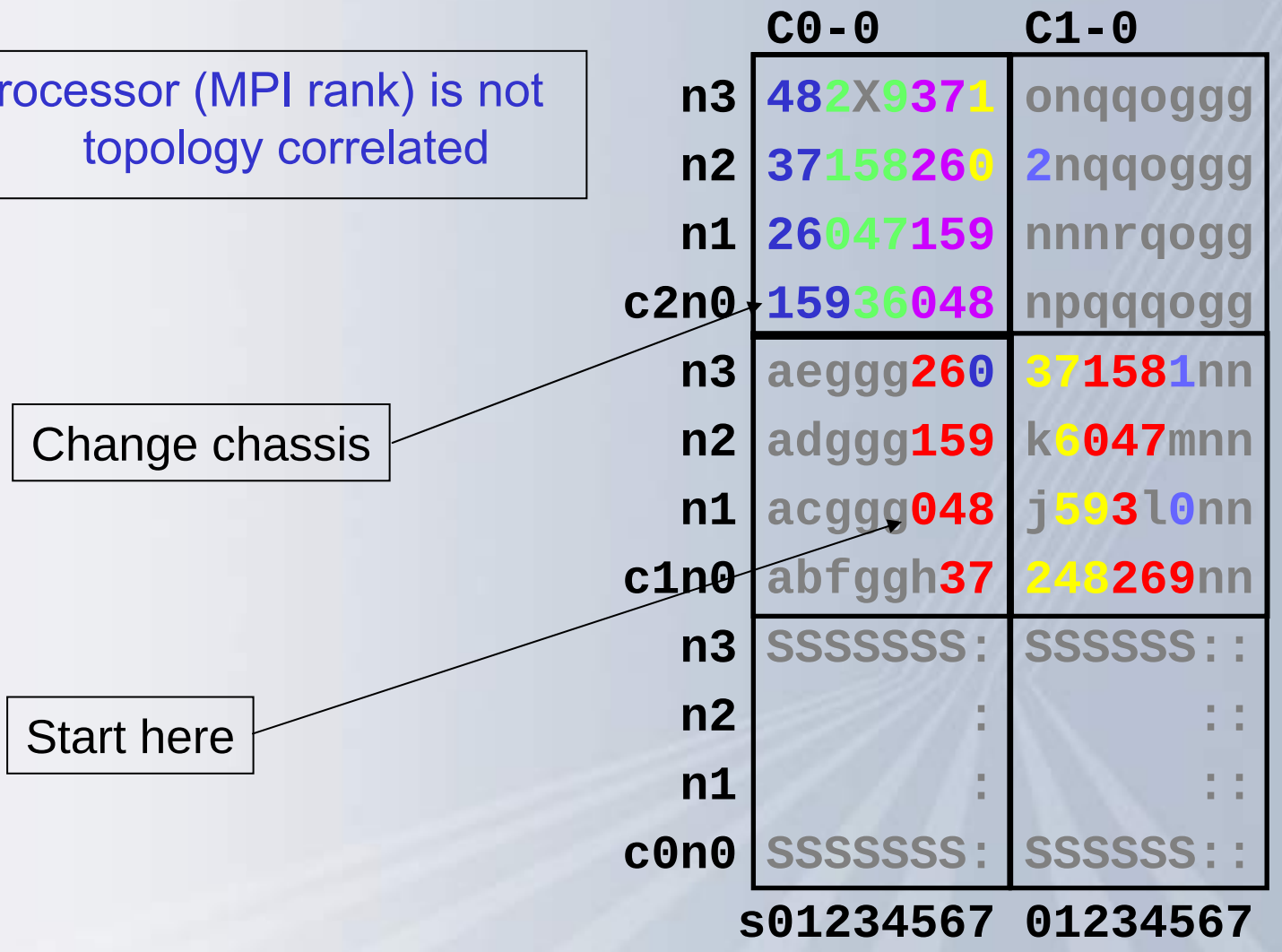
Z dimension links



```
      C0-0     C1-0     C2-0     C3-0     C4-0     C5-0     C6-0     C7-0
  n3 iiiXiiii onqqoggg |||||||| wwwyyyyy nnnBBBBB DDDDDDxB CCCzzzzz GGGGGGww
  n2 iiiiiiii inqqoggg |||||||| nnnzBBBB |DDDDDxB CCCCzzzz GGGGGGww
  n1 iiiiiiii nnnrqogg |||||||| wwwyyyyy nnnnBBBB |DDDDDxB CCCCzzzz GGGGGGww
c2n0 iiiiiiii npqqqogg |||||||| wwwxyyyy nnnnBBBB |DDDDDDx BCCCzzzz wGGGGGGX
  n3 aegggiii iiiiiinn ggsitv|| q||||ppw nnnnnnnn |||||||| yyyBBBBB zzzzzBFw
  n2 adgggiii kiiiimnn gggsout| qw|||npw nnnnnnnn |||||||| yyyBBBBB zzzzzBwF
  n1 acgggiii jiiilinn gggsott| qq|||||w nnnnnnnn |||||||| yyyyBBBB zzzzzBwF
c1n0 abfgghii iiiiiinn gggsott| qq|||||q nnnnnnnn |||||||| yyyyBBBB zzzzzzwF
  n3 SSSSSSS: SSSSSS:: gggggggg |||||qqq yyyyyyyn qpppCBB| BBBwwwwy xEzzzzzz
  n2        :       :: gggggggg |||||qqq yyyyyyyn qqpppBB| BBBwwwwy xEzzzzzz
  n1        :       :: gggggggg ||||||qq yyyyyyyn qqpppBB| BBBBwwww xEzzzzzz
c0n0 SSSSSSS: SSSSSS:: gggggggg ||||||qq yyyyyyyy qqpppCBB BBBBwwww zEzzzzzz
    s01234567 01234567 01234567 01234567 01234567 01234567 01234567 01234567

Legend:
   nonexistent node                 S  service node
:  free interactive compute node    A  allocated, but idle compute node
|  free batch compute node          ?  suspect compute node
X  down compute node                Y  down or admindown service node
Z  admindown compute node           R  node is routing
```

Processor allocation to applications

Processor (MPI rank) is not topology correlated



Processors allocation does not matter so much

- CPA allocation strategy is not topology aware
  - Same CPA strategy on every XT4 systems (by NID)
  - Topology depends on the size (class)
- However application performance does not significantly suffer from that
  - Reproducible results on production workload
  - The Cray XT4 provides flat performance
- CPA allocation strategy is ... well... non-optimal, but
- The way processors are allocated does not affects significantly application performance

Online Cray docs

http://docs.cray.com/

http://docs.cray.com/cgi-bin/craydoc.cgi?mode=SiteMap;f=xt3_sitemap