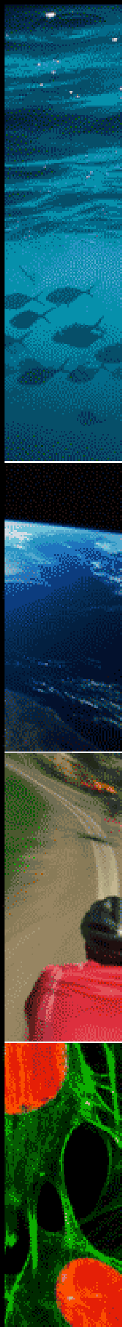
The Supercomputer Company

Compute Node OS for XT3

Jim Harrell
Cray, Inc

CUG 2006

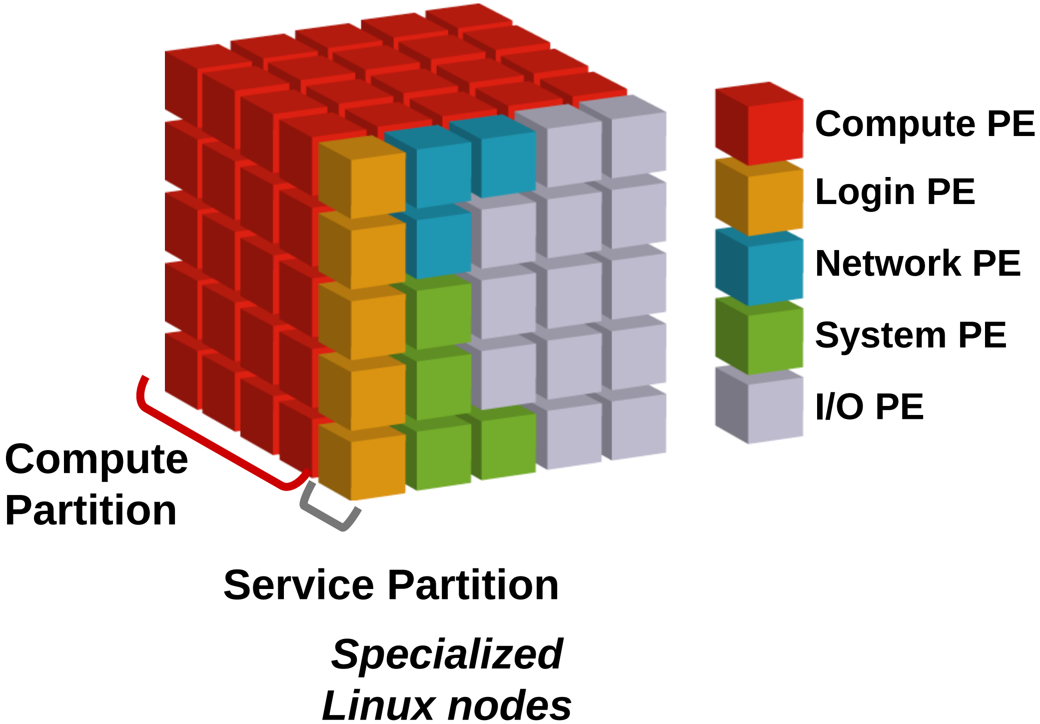
Presenter



Agenda

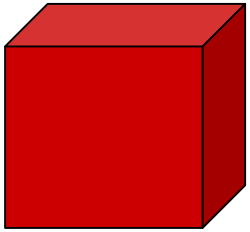
- XT Software Architecture
- Rationale and requirements for an alternate Compute Node Operating System (CNOS)
- Review of Catamount
- Review of Linux
- Other Alternatives
- Current Status

Scalable Software Architecture: Unicos/lc



- Division of labor – distributing the system services on Nodes in the Service group
  - Combines with HW Architecture
  - Supports application profile
  - Provides the opportunity to evolve services and how they are distributed
- Service Nodes are currently using configurations of Linux
  - Fits the HPC community use of Unix/Linux
  - Provide all system services
- Compute Nodes are support application processes
  - Limited functions delegated to the compute nodes in order to allow user processes priority

A Compute Node Operating System (CNOS), A Light Weight Kernel (LWK), A Rose…



- A Definition of CNOS would be based on scaling and efficiency
  - Supports an application process (or processes)
  - No demand paging, timesharing, sockets, …
- Compute Node characteristics –
  - Provides support for the machine dependent aspects of the node
  - Supports the High Speed network (HSN) connection – HSN is “allocated” to application usage
  - Provides support for application processes allocated to the node
  - Provides limited set of system call functions – most system calls forwarded to Service nodes
  - Small memory footprint and CPU utilization (when not servicing requests)

Rationale for a different CNOS

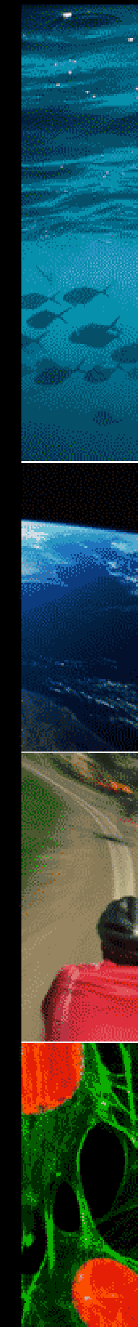
- New or different hardware
  - More processors
  - Different processor characteristics
  - Unique hardware
- New requirements for functionality
  - Networking applications
  - Different file systems
  - Global Arrays, UPC, CAF, etc.
  - OpenMP
- Common CNOS for all Cray Products
  - Leverage development and support
  - Fits with long term directions for specialized nodes

Cray Requirements for a new LWK

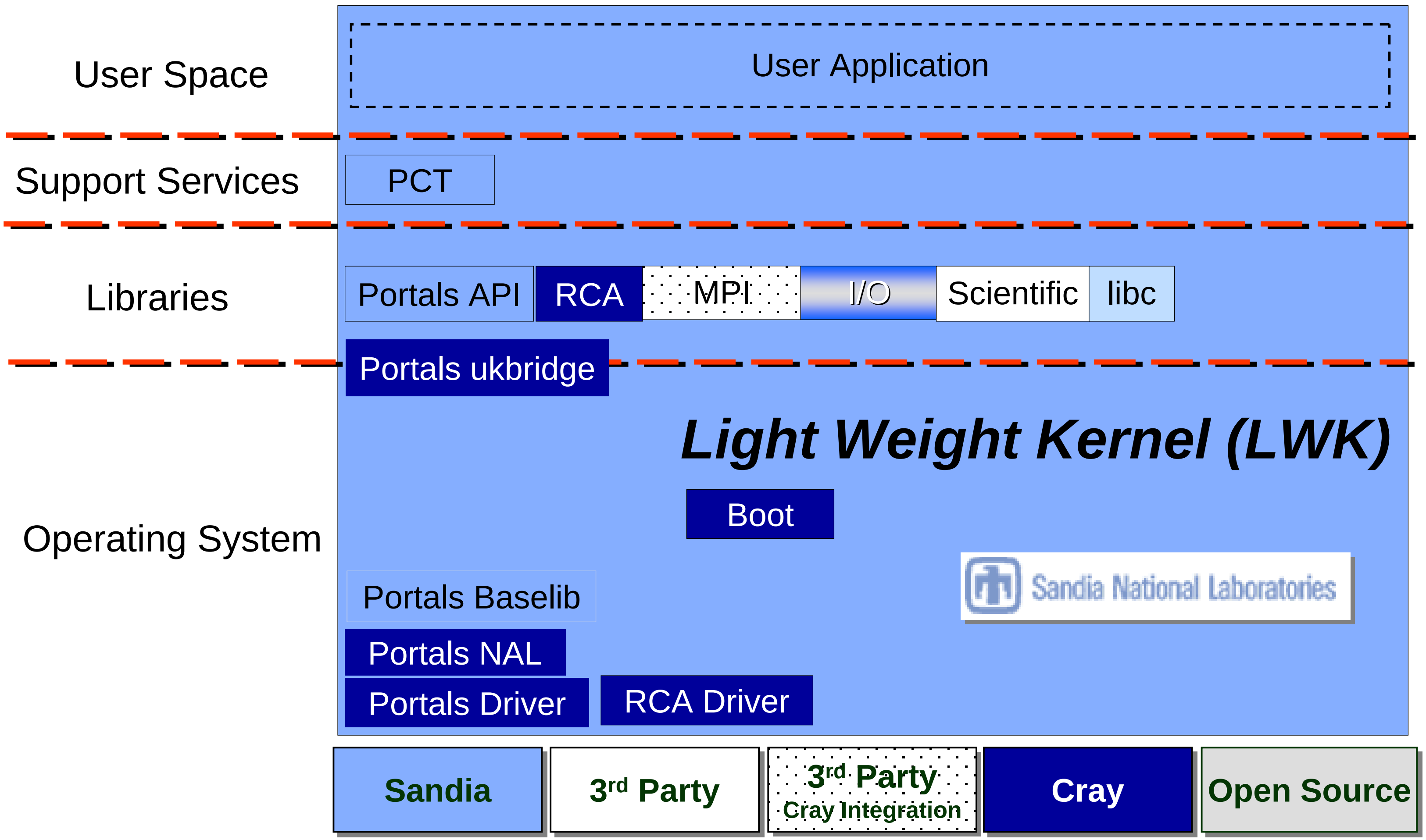
- The requirements for a compute node are based on Catamount functionality and the need to scale
- 30K compute nodes – Doesn't have to be 30K on day 1
- MPI, SHMEM, UPC, CAF, Global Arrays, etc.
- Application I/O performance and functionality at least equivalent to Catamount
- Signals need to work to control applications
- Support for sockets
- OpenMP
- Start applications as fast as Catamount – bigger hurdle
- Boot compute nodes – almost as fast…
- Can have user environment changes

Catamount Review

CUG 2006



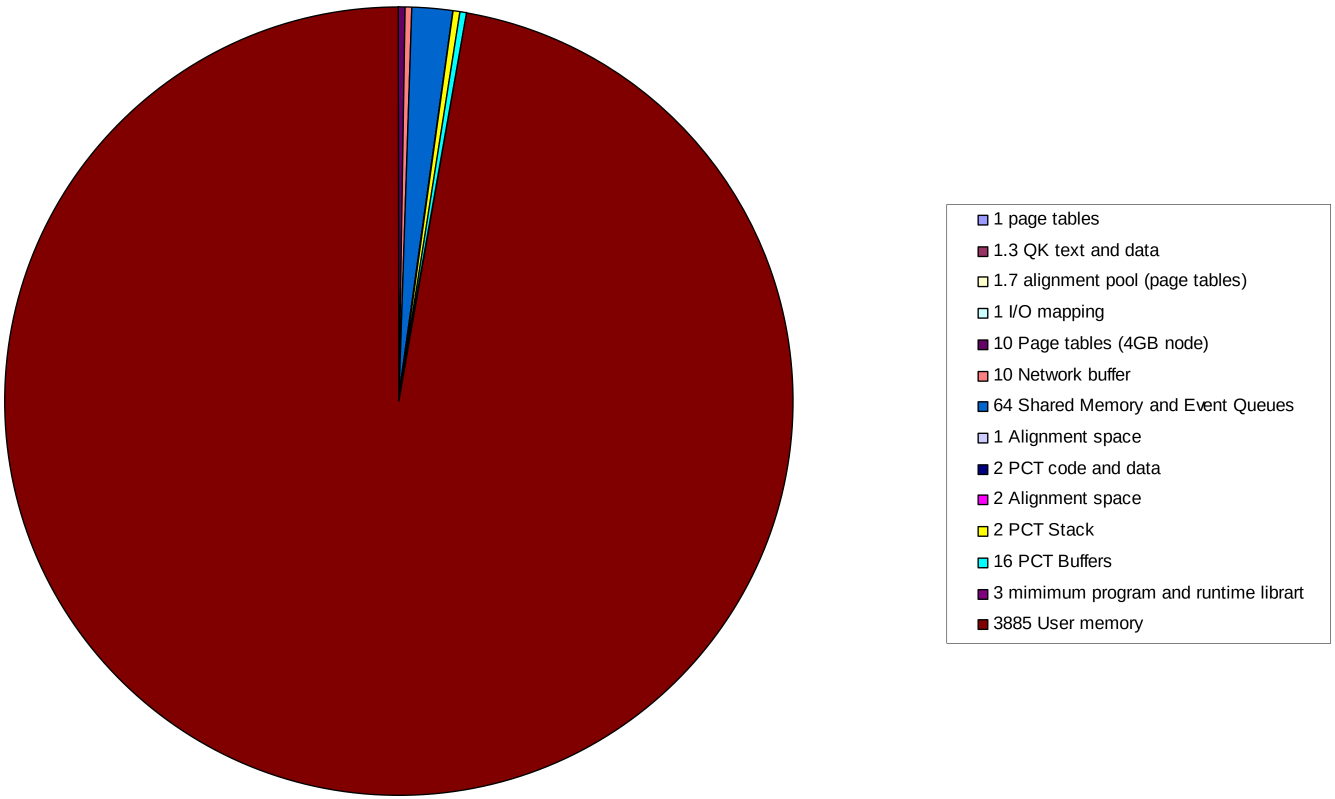
Catamount Component View



Runs on all Compute Nodes

It's all about the Application…

Catamount Memory Usage



Catamount - Positives

- Scales – We have real systems running at 5K and 10K nodes
- OS CPU usage does not interfere with applications – next slides
- Small memory footprint ~115MB
- Provides support for many HPC application models
- "Mature" code base with limitations that are understood

Catamount - Positives

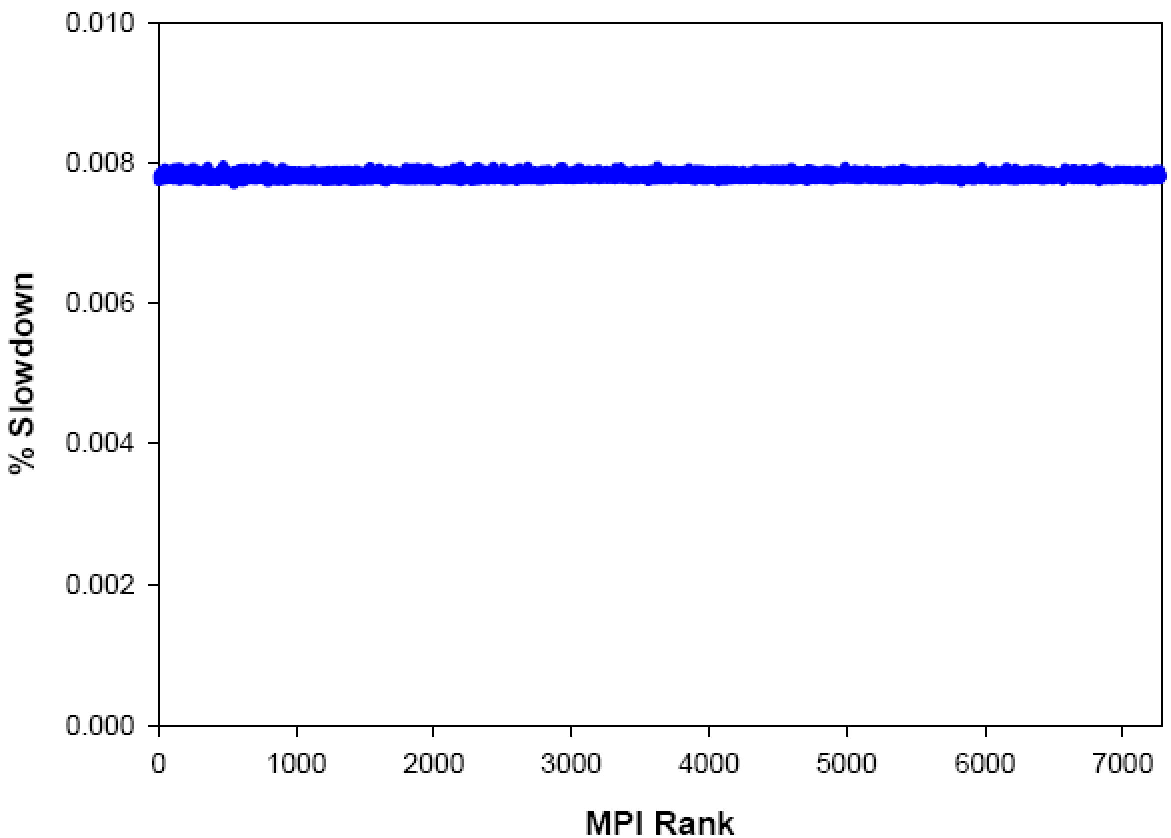


Figure 3: Slowdown caused by computational “noise”

Source - LA-UR-05-7435 Current Status of Red Storm.., Barker, et.al.

Catamount - Positives

3. Computational noise

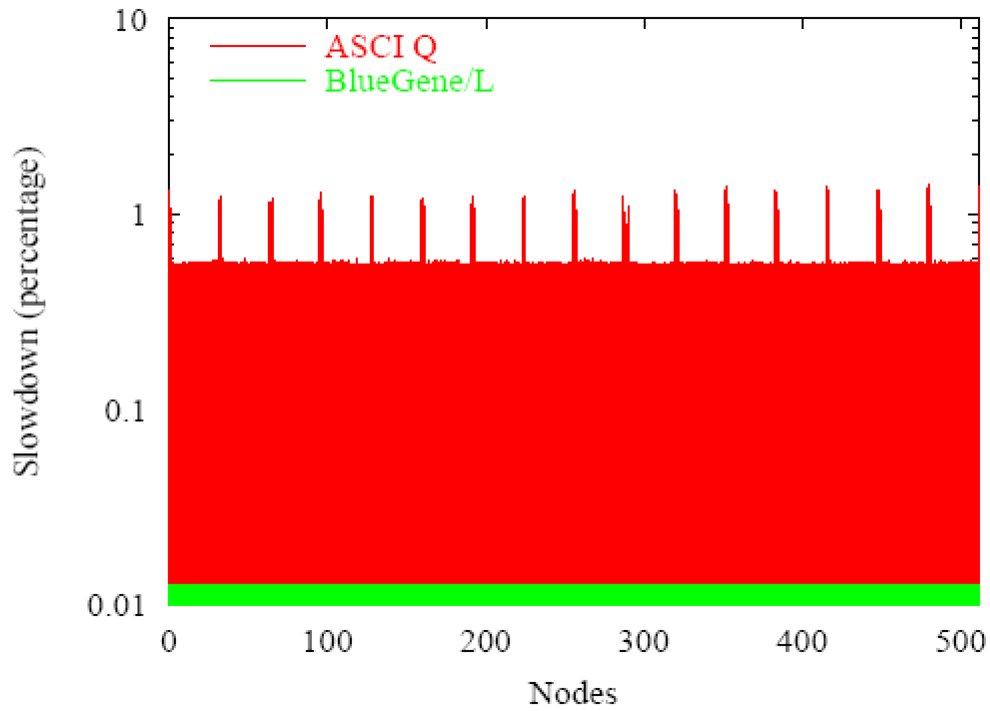


Figure 2. Slowdown due to computational noise on a per node basis.

Source - LA-UR-04-1114 Blue Gene A Performance and Scalability.. Davis, et.al.

Catamount – Issues

- Catamount was developed to support a set of MPI applications. In order to broaden the application set a number of features need to be considered.
- Sandia and Cray are sole sources for Catamount development and support
  - The design and development of any new feature or fix is done by one or the other organization
- Support for new processors and devices have to be added after the hardware becomes available – Support for other software is available at processor or device release
- Fully productizing Catamount is substantial work
  - Logging and debugging are still difficult problems

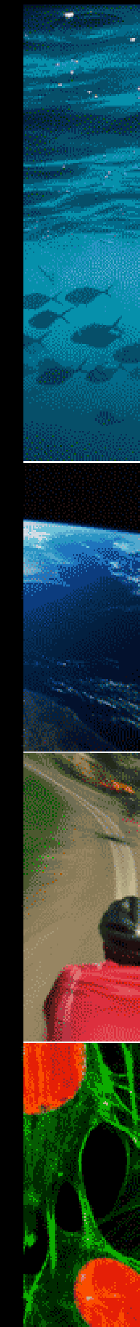
Catamount – Issues

- Catamount design has some limitations that will be difficult to overcome:
  - File System support
  - Application Network Support
  - Many system calls missing
    - User/system time
  - Debugging support (kernel)
  - Signal support – for exit
  - Multiple threads in the kernel
  - Lack of support for multi-core or SMP
  - Robustness problems
- Sandia had not been planning future versions of Catamount to support 4 cores
- The support for more than 1 or 2 processors/cores is required

Catamount – Extensions

- Catamount might be used as a base for quad core. Decisions about the Architecture are being discussed. This could be an alternative or a stopgap…
- A limited set of features could be added to Catamount. OpenMP as an example – might be possible. This could be an alternative to a heavier weight, but fully featured kernel.
- It may be possible to support Global Arrays with Catamount

Linux Review

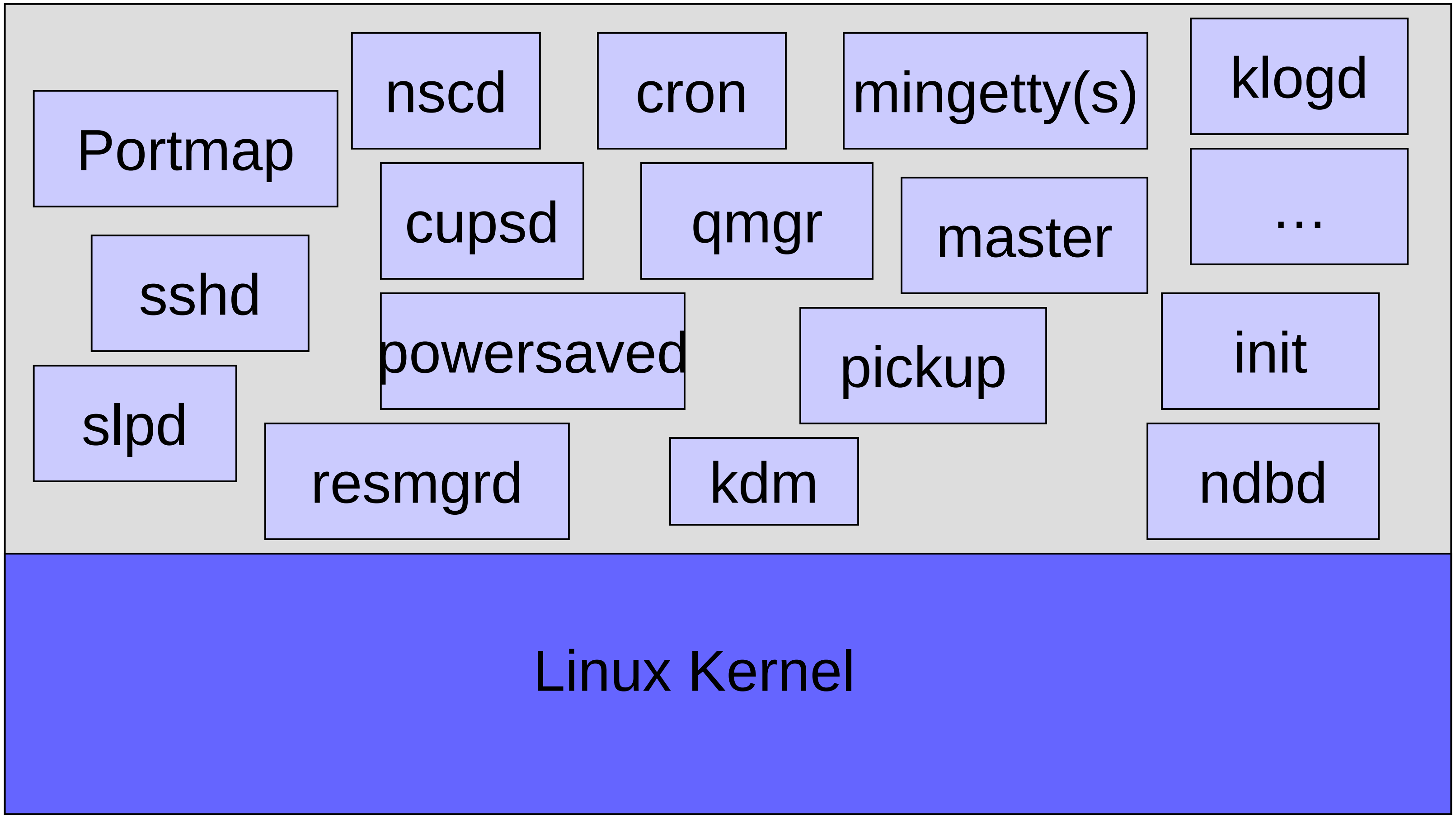


CUG 2006

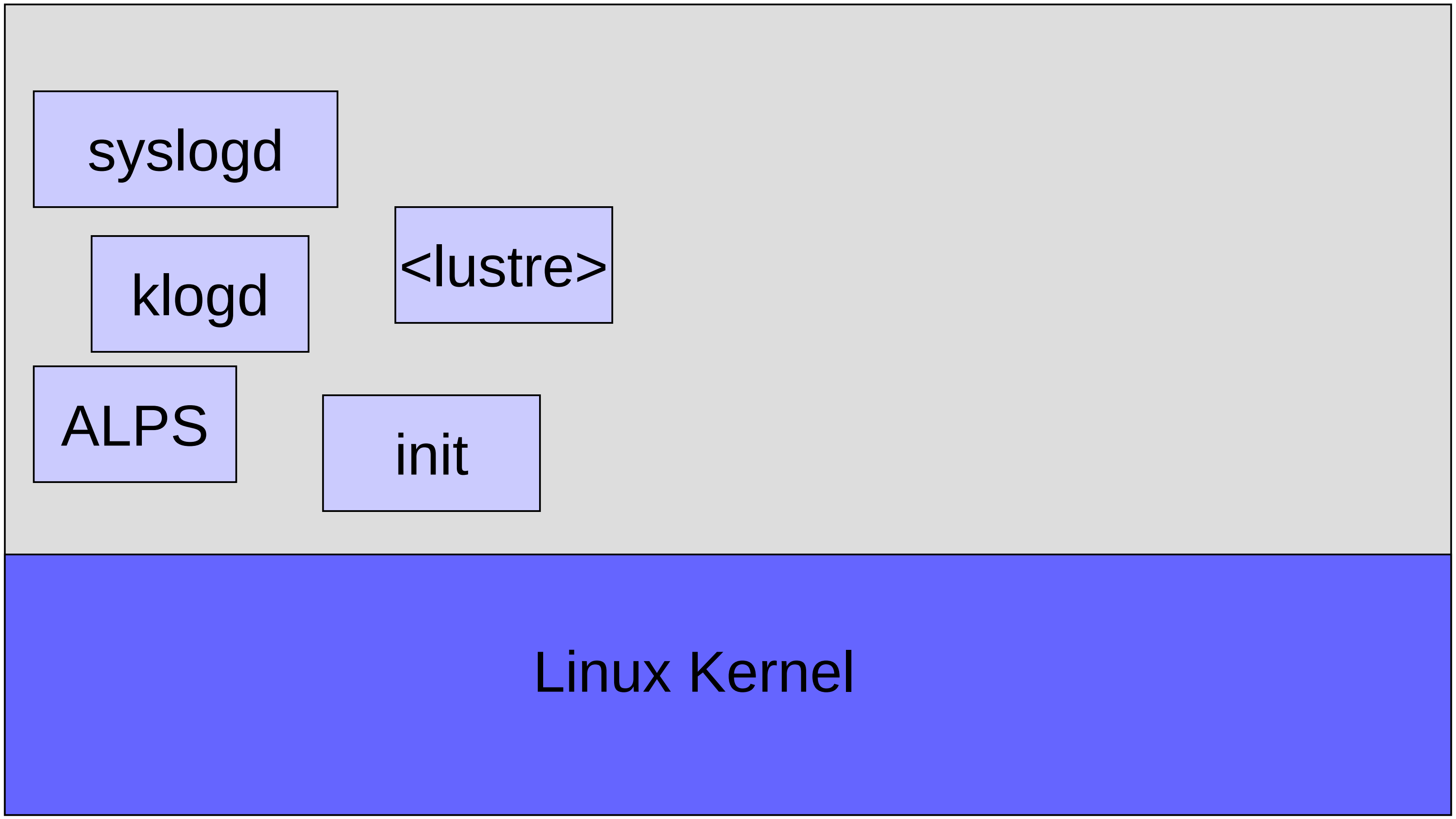
Compute Node Linux – is not Linux..

- Linux as planned for a compute node OS is not Linux as you would expect on a server
  - Only services required to run application processes
  - Modified configuration to fit only application process support

Linux – The Standard Server



Linux – The Compute Node



Linux - Positives

- Linux is an accepted standard for x86 architectures (for Unix)
  - Almost all of HPC is using Linux or a Unix variant
- Development is “Shared” among the Linux community
  - HPC does have a limited following and limited influence
- Linux supports a broad variety of applications and application models
- New hardware and new features are available in Linux before other systems
- Linux is still a light weight architecture

Linux - Issues

- Linux has no concrete examples of its use on an HPC system at 5K or 10K nodes
- Linux as a compute node will not provide all the services that a standard Linux system would provide
  - Most services and demons would be removed to ensure proper performance
- Changes to Linux to provide a better compute node LWK will have to be supported outside of the distributions
- Linux memory management does not handle alternate page sizes as well as would be desired
  - Some “patches” from other vendors may help, but the patches would then be a support burden
- Linux scheduler and clock tick features will likely need to be modified to optimize the performance
- Portals performance issues in Linux need to be investigated and fixed

Linux - Issues

- Root file system is required for Linux – but
  - Use of a RAM file system takes memory
  - RAM file system would have to be created and maintained
  - RAM file system would limit shared library usage
  - Shared root – like NFS or Lustre may simply not work at 30K nodes
- Work in Programming Environment – MPI, libraries, and tools
  - Although much of this work would make us more standard
- Networking adds complexity
  - Each node needs an IP address
  - ARP and other noise needs to be eliminated from the HSN network
  - NAT, RSIP, or alternative for routing outside the machine
- Memory footprint is larger. How to constrain it?
- User credentials – Need to support some mechanisms
- File Systems – Need to aid Lustre client scaling to 30K nodes

Linux – Decisions and Investigations

- Basic System Architecture remains the same –
  - Service nodes, compute nodes, …
- Chosen to start by aiming for scalable Compute node Linux
  - Initial Linux LWK will have functionality greater than Catamount, but less than a full Linux
- Using CRMS infrastructure – exactly as it exists in development
  - No serious changes for admins
- The ALPS package will be used as the Runtime
  - Shared support with other Cray products
  - Similar to yod/CPA/PCT – transition should be simple
- Job scheduling – PBS, LSF, MOAB, etc.
- Lustre is the supported file system for user files
- Network is automatically set up
  - Fixed hosts and IP addresses
  - Static routes – no ARP
  - Looking at RSIP

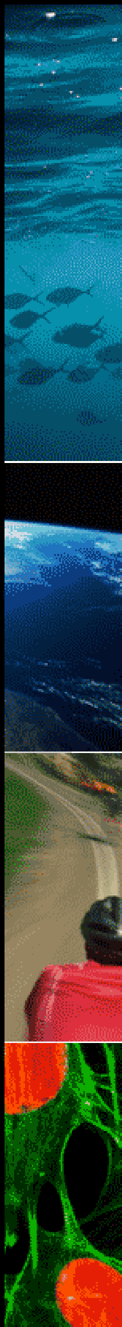
Linux – Decisions and Investigations

- Boot a Linux kernel on a compute node in 15 seconds
  - Needs attention
- Have reduced the set of services to:
  - Init -
  - Appinit - ALPS
  - RCA kernel module – for heartbeat and event notification
  - Sshd or console – for remote access
  - busybox utilities – minimal user support for admin or debugging
  - Still under investigation -
    - ntp, syslogd, getty
- Using a small – but not optimized RAM root
  - Debugging support and test services
- User credentials – currently using ALPS for all credentials
- Testing with shared library support, but committing to static binaries
  - Capability to have dynamic libraries in the RAM root file system exists but costs memory

Linux – Decisions and investigations

- Time synchronization across nodes – needs to be considered
- Portals performance improvements – GART/MRT
- Kernel configuration – What can be removed
- Lustre – currently working with standard Lustre clients
  - Scaling concerns around mount
  - Tuning planning
  - LibLustre as a fallback
- Scaling studies – just starting
  - Currently running on 128 nodes
  - Some early testing at ~2K nodes
- Measurements - These need to be defined/designed
  - Job start
  - I/O performance
  - Network performance
  - OS jitter/noise on nodes
  - Boot times
  - Console/CRMS load
  - Memory usage/availability
  - Syslog
  - Lustre client

Alternatives and Status



CUG 2006

Alternatives…

- Multiple CNOSs
  - Ability to run Catamount and Linux on different nodes (needs further study).
    - Need to provide some common allocation controls
    - Need to have software attributes – this feature is planned
  - Fat Linux and Fatter Linux
    - Could allow some compute nodes to have more services – but not impact overall scalability
  - Virtualization Layer
    - Xen
  - FastOS ideas

Current Status

- 1.4 provides support for Catamount Virtual Node
  - Master/Slave support for Dual Core
- 2.0 will provide support for Compute Node Linux in small (4 compute node cabinet) configurations
- A future release will provide scalable support for CNL to … a large number of nodes