Catamount Software Architecture with Dual Core Extensions

May 9, 2006
Ron Brightwell,
Sue Kelly, and John VanDyke
Sandia National Laboratories
SUNMOS, PUMA, Cougar, Catamount Design Goals

• Targeted at massively parallel environments comprised of thousands of processors with distributed memory and a tightly coupled network.
• Provide *necessary* support for scalable, performance-oriented scientific applications.
• Offer a suitable development environment for parallel applications and libraries.
• Emphasize efficiency over functionality.
• Maximize the amount of resources (e.g. CPU, memory, and network bandwidth) allocated to the application.
• Seek to minimize time to completion for the application.
Catamount is designed for an MPP environment with functional partitions.
Catamount Physical Memory layout

<table>
<thead>
<tr>
<th>QK text</th>
<th>QK data</th>
<th>Stack</th>
<th>Network buffer (qk heap)</th>
<th>Portals memory</th>
<th>PCT text</th>
<th>PCT data</th>
<th>Stack</th>
<th>PCT heap</th>
<th>Stack</th>
<th>User heap</th>
</tr>
</thead>
</table>

Note: not to scale
Quintessential Kernel (QK)

• Policy enforcer
• Initializes hardware
• Handles interrupts and exceptions
• Maintains hardware virtual addressing
• No virtual memory support
• Static size
• Non-blocking
• Few, well-defined entry points
Process Control Thread (PCT)

- Runs in user space
- More privileged than user applications
- Policy maker
  - Process loading (with yod)
  - Process scheduling
  - Virtual address space management
  - Fault handling
  - Signals
YOD runs in the service partition

• Functions
  – Controls the logarithmic launch of a parallel job
  – Proxies standard I/O, plus other I/O, if necessary
  – Manages the parallel job throughout its run

• Yod is an evolution of the xnc (eXecute Network Computer) program used to launch jobs on the nCube: \((x+1)(n+1)(c+1) = yod\)

Dual Core Support for Catamount

• Motivation for Virtual Node (VN) on Catamount
  – Virtual Node Mode was a very successful late addition to Cougar on ASCI Red
  – Doubles the number of available nodes
  – Significantly increases compute power for many applications

• AMD has a dual-core Opteron that simply plugs into an XT3 node
Catamount Dual Core Design

• Follow Cougar and ASCI Red
  • Application perspective
    – Twice as many nodes
    – Half the memory
  • System perspective
    – One copy of QK (only a subset of the code runs on CPU-1)
    – One PCT
    – Network access done by CPU-0 QK only
    – Network requests from CPU-1 are proxied to CPU-0
• Network perspective
  – One Node Identifier
  – Two process Indices
Dual Core CPU Responsibility Assignments

Dual Core Opteron

CPU-0
- QK
- APP-0
- PCT

CPU-1
- QK subset
- APP-1

Seastar Network Interface Chip

[Image of Dual Core Opteron diagram with CPU-0 and CPU-1 assignments]
Catamount’s libc is pruned version of glibc

- No threads support
- No off-node communication other than via Portals, such as pipes, sockets, rpc's or Internet Protocols
- No dynamic process creation; for example: no exec(), fork(), popen(), or system()
- No dynamic loading of executable code
- Limited signals support
- No /proc or ptrace
- No mmap. A skeleton function is supplied, but returns −1.
- No profil()
- Limited ioctl
- No getpwd family of calls
- No functions requirement any form of db (e.g. ndb). For example, there is no support for the uid, gid family of queries that based on the ndb.
- No terminal control
- No functions that require UNIX-style daemons
- Custom catamount malloc is used by default
Libsysio routes I/O calls to the appropriate file system handler.
Libcatamount

- RPC mechanism to communicate with yod for stdio and system call offload
- Custom malloc tuned for large allocations
- Pre-main initialization
- Interface routines for PCT and QK services
Libportals

- Message passing API
- Separate software package
- Required by Catamount
- http://www.sourceforge.net/packages/sandiaportals
Multi-Partition Job Support is new with Catamount

- Support for parallel applications that span Catamount and Linux
  - Yod using load file option (-F)
  - Requires a PCT to run on Linux
  - Requires different executables
  - Creates one MPI_COMM_WORLD
Future Plans

• Studying whether catamount virtual node design is viable for four-core support
• Utilize a portals protocol offload engine in the Network Interface Chip (NIC)