



Why Linux is a Bad Idea as a Compute Node OS (for Balanced Systems)

Ron Brightwell

Sandia National Labs

Scalable Computing Systems Department

bright@cs.sandia.gov



Target Architecture

- **Distributed memory, message passing**
- **Partition model of resources**
 - **Compute nodes**
 - **Small number of CPUs**
 - **Diskless**
 - **High performance network**
 - **Peak processor speed in MHz is near peak network bandwidth in MB/s**
 - **Service nodes**
 - **Disk I/O nodes**
 - **Network I/O nodes**



Target Applications

- **Resource constrained**
 - Can consume at least one resource (memory, memory bandwidth, processing, network, etc.)
 - *All* resources are precious
- **A single run may consume the entire system for days**
- **Primary concern is execution time**



Why Linux for Cplant™?

- **Free (speech & beer)**
- **Large developer community**
- **Kernel modules**
 - No need to reboot during development
 - Supports partition model
- **Supported on several platforms**
- **Familiarity with Linux**
 - Ported Linux 2.0 to ASCI/Red



Results

- **Cplant™ is now open source**
- **Large developer community is a wash**
 - Most developers not focused on HPC and scaling issues
 - Extreme Linux helped
 - Extreme Linux isn't very extreme
- **Modules**
 - Big help in developing the networking stack
- **Portals over any network device**
 - Myrinet
 - RTS/CTS to skbufs
 - Portals over IP
 - Portals over IP in kernel
- **Cplant™ runs on Alpha, x86, IA-64**
- **Linux changes too often to really be familiar**



Other Observations

- **Reliability**
 - Linux likely hasn't been the cause of any machine interrupts
 - But we can't really be sure
 - Main selling point of Linux for the server market
- **Application development environment more extensive**
 - Compilers, debuggers, tools
- **Lots of stuff we don't have to worry about**
 - Device drivers: Ethernet, Serial
 - BIOS's
 - Hardware bugs
- **Linux works OK for Cplant™ and commodity-based clusters**



Technical Issues

- **Predictability – avoid work unrelated to the computation**
 - Linux on Alpha takes 1000 interrupts per second - to keep time
 - Daemons: init, inetd, ipciod
 - Kernel threads: kswapd, kflushd, kupdate, kpiod
 - Seen as much as a 10x variability in execution time
 - Inappropriate resource management strategies
- **VM system**
 - Adverse impact on message passing
 - No physically contiguous memory
 - Must pin memory pages
 - Must maintain page tables for NIC
 - Fighting the page cache
 - How much memory is there?



Technical Issues (cont'd)

- **Requires a filesystem**
 - fork/exec model
 - Not appropriate for diskless compute nodes
- **Complexity**
 - We haven't done anything with Linux because it's not easy
 - Virtual node mode added to LWK by two relatively inexperienced kernel developers in 6 months



Social Issues

- **Kernel development moves fast**
 - Significant resources needed to keep up
- **Distributions and development environments also change frequently**
 - Tool vendors have trouble keeping up
- **Linus changed out the VM system in the middle of the 2.4 kernels!**
 - 2.4.9 – van Riel VM system
 - 2.4.10 – Arcangeli VM system
 - 150+ patches to the van Riel VM system
 - Linux fork?
- **Server vs. multimedia desktop**
 - Not HPC



Social Issues (cont'd)

- **Forced to take the good with the bad**
 - Want NFS v3, don't want OOM killer
- **Fairly fixed set of requirements**
 - Linux doesn't allow us to concentrate on those
- **Staying focused**
 - Linux community not addressing HPC issues
- **Linux is going to die soon anyway**
 - Ron Minnich says so



LWK NOP Trap Performance

- **Proc 0 mode**
 - 2.5052 μs
- **Proc 1 mode**
 - 1.6675 μs
- **Proc 3 mode**
 - 2.2968 μs
 - 2.7146 μs
- **Proc 0 mode with `-share`**
 - 4.4973 μs
 - 3.3056 μs



Lightweight Kernel?

- **Puma**

– section	size
– .text	83357
– .data	21856
– .bss	105440
– Total	210653

- **PCT**

– section	size
– .text	274993
– .data	50928
– .bss	642344
– Total	968265



Memory Use Breakdown

- Executable
 - $164833(.text) + 35340(.data) + 135816(.bss) = 335989$
- Proc 0 / Proc 1 mode
 - Text = 167936
 - Data = 172032
 - Stack = 2097152
 - Heap = 260046848
 - NX Heap = 262144
- Proc 3 mode
 - Text = 167936
 - Data = 172032
 - Stack = 2097152
 - Heap = 127926272
 - NX Heap = 262144



Source Lines Of Code (SLOC)*

- QK
 - 9078 CPU independent
 - 10061 x86-specific
 - 6088 i860-specific
- PCT - 12823
- yod - 13150
- Libraries
 - I/O and C – 16343
 - MPI
 - Device independent - 23849
 - Portals – 5234
- Linux (kernel, init, mm, include/linux) - 87453

*Generated using 'SLOCCount' by David A. Wheeler



It's not the size – it's what you do with it 😊



Rolf's Points

- **Linux is well-known**
 - So is Windows
- **Let other people do the work**
 - The other people don't care about HPC
- **Daemons don't really do anything**
 - Then why have them?
- **Use a real-time variant of Linux**
 - Real-time scares me - PROSE
- **Somebody needs to tweak Linux**
 - Tweaking is not easy



Summary

- **Linux works OK for Cplant™ and commodity clusters**
 - CPU performance is acceptable for cluster bytes-to-FLOPS ratio
- **Likely performance issues for large-scale platforms with a reasonable bytes-to-FLOPS ratio**
- **Community is a mixed blessing**