On the Appropriateness of Commodity Operating Systems for Large-Scale, Balanced Computing Systems

Ron Brightwell Rolf Riesen
Sandia National Laboratories
Albuquerque, New Mexico
{rbbrigh,rolf}@sandia.gov

Arthur B. Maccabe
University of New Mexico
Albuquerque, New Mexico
maccabe@cs.unm.edu

Outline

- Background
- Target architecture and applications
- Experience with Linux
- Summary
- Future directions

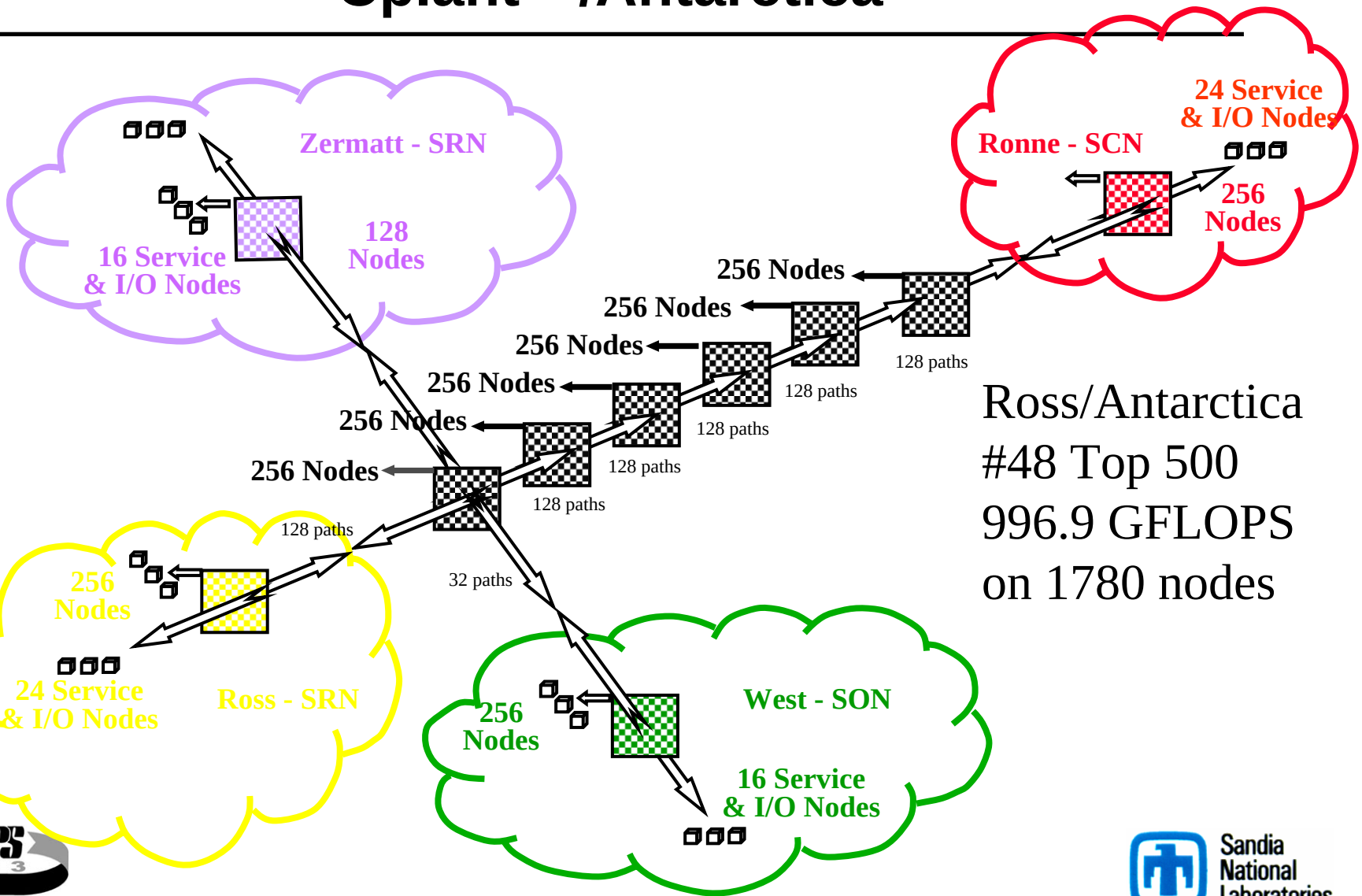
Sandia/UNM System Software Research

- Intel Paragon
  - 1,890 compute nodes
  - 3,680 i860 cpu's
  - 143/184 GFLOPS
  - 175 MB/sec network
- SUNMOS lightweight kernel
  - High performance compute node OS for distributed memory MPP's
  - Deliver as much performance as possible to apps
  - Small footprint
  - Started in January 1991 on the nCUBE-2 to explore new message passing schemes and high-performance I/O
  - Ported to Intel Paragon in Spring of 1993
- Intel ASCI Red
  - 4,576 compute nodes
  - 9,472 Pentium II cpu's
  - 2.38/3.21 TFLOPS
  - 400 MB/sec network
- Puma lightweight kernel
  - Multiprocess support
  - Modularized (kernel, PCT)
  - Developed on nCUBE-2 in 1993
  - Ported to Intel Paragon in 1995
  - Ported to Intel TFLOPS in 1996 (Cougar)
  - Portals 1.0
    - User/Kernel managed buffers
  - Portals 2.0
    - Avoid buffering and mem copies

Cplant™/Antarctica

- 1792+ Compaq DS10L Slates
  - 466MHz EV6, 256 MB RAM
- 590 Compaq XP1000s
  - 500 MHz EV6, 256 MB RAM
- Myrinet 33MHz 64bit LANai 7.x and 9.x
- Myrinet Mesh64 switches
- Classified, unclassified, open, and development network heads

Cplant™/Antarctica



Ross/Antarctica
#48 Top 500
996.9 GFLOPS
on 1780 nodes

Target Architecture

- Distributed memory, message passing systems
- Partition model of resources
  - Compute nodes
    - Small number of CPUs (<4)
    - Diskless
    - High performance network
  - Service nodes
  - Disk I/O nodes
  - Network I/O nodes
- Balanced
  - Ratio of peak processor speed to peak network bandwidth
  - Ratio of peak processor speed to peak memory bandwidth

Target Applications

- **Resource constrained**
  - **Can consume all of 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 application 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 2.0.13 to ASCI/Red nodes in 1997
    - No network though
- Port of Cougar infeasible for schedule

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 (see Linux Magazine)
  - Other markets starting to help (eg. databases)
- Modules
  - Big help in developing the networking stack
- Portals over any network device
  - Myrinet
  - 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
    - Problems when we tried to play with this
  - 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 (usable) physically contiguous memory mechanism
  - Must explicitly 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 where filesystem is all at user-level
- Complexity
  - We haven't done anything substantial with Linux because it's not easy (and moves too fast)
  - Virtual node mode added to Cougar by two relatively inexperienced kernel developers in six months

Social Issues

- Kernel development moves too fast
  - Significant resources needed to keep up and maintain a production system
- Distributions and development environments also change frequently
  - Tool vendors have trouble keeping up (ask Etnus)
  - Last two bugs on Cplant™ were with glibc from RedHat
- 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
- Server vs. multimedia desktop
  - Neither one is 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
  - No real market drivers

Trends Are Helping Linux

Machine	Memory per Node	TLB Entries	CPU Speed	Network
Paragon	16 MB	4	50 MHz	200 MB/s
ASCI Red	256 MB	64	333 MHz	400 MB/s
Cplant™	1 GB	128?	466 MHz	100 MB/s

Summary

- Linux works fine for Cplant™ and commodity clusters
  - CPU performance is acceptable for cluster balance factors
- Likely performance issues for large-scale platforms with a reasonable balance ratios
- Community is a mixed blessing
- Linux will likely catch up, but we have large-scale systems now

Future Directions

- Currently performing a direct comparison between Cougar and Linux on ASCI Red hardware
  - Finally did a network driver for ASCI Red network
  - Should allow us to have a better understanding of Linux performance and scalability on a balanced machine
- Working on an approach for a lightweight kernel that leverages Linux for hardware support