# Lightweight Operating Systems for Scalable Native and Virtualized Supercomputing

**April 20, 2009**

**ORNL Visit**

**Kevin Pedretti**
**Senior Member of Technical Staff**
**Scalable System Software, Dept. 1423**
**ktpedre@sandia.gov**

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

# Acknowledgments

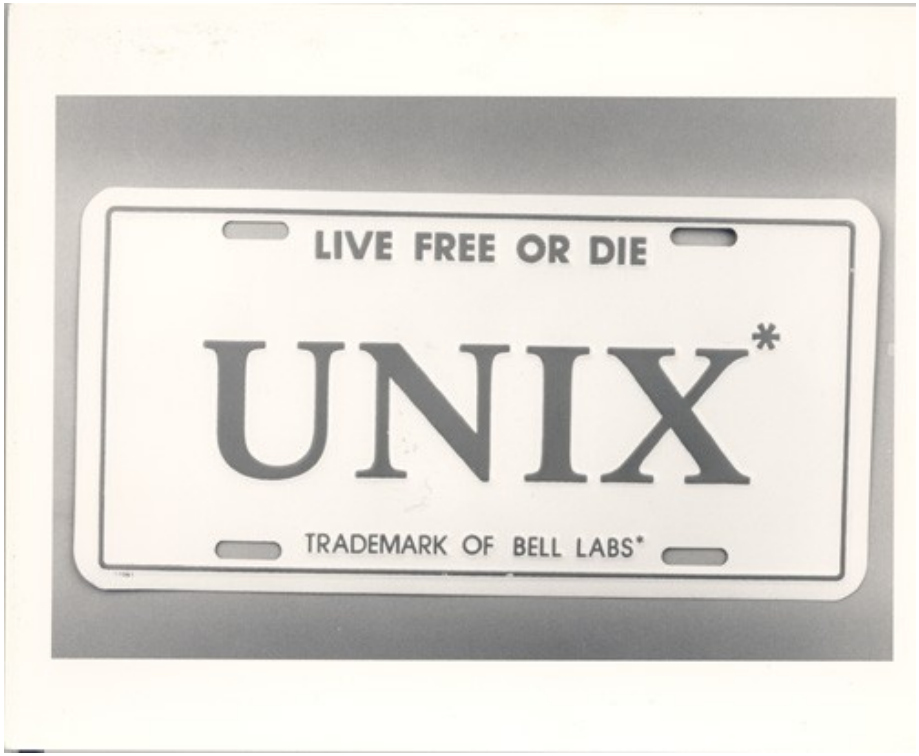

- Kitten Lightweight Kernel
  - Trammel Hudson, Mike Levenhagen, Kurt Ferreira
  - Funding: Sandia LDRD program
- Palacios Virtual Machine Monitor
  - Peter Dinda & Jack Lange (Northwestern Univ.)
  - Patrick Bridges (Univ. New Mexico)
- OS Noise Studies
  - Kurt Ferreira, Ron Brightwell
- Quad-core Catamount
  - Sue Kelly, John VanDyke, Courtenay Vaughan

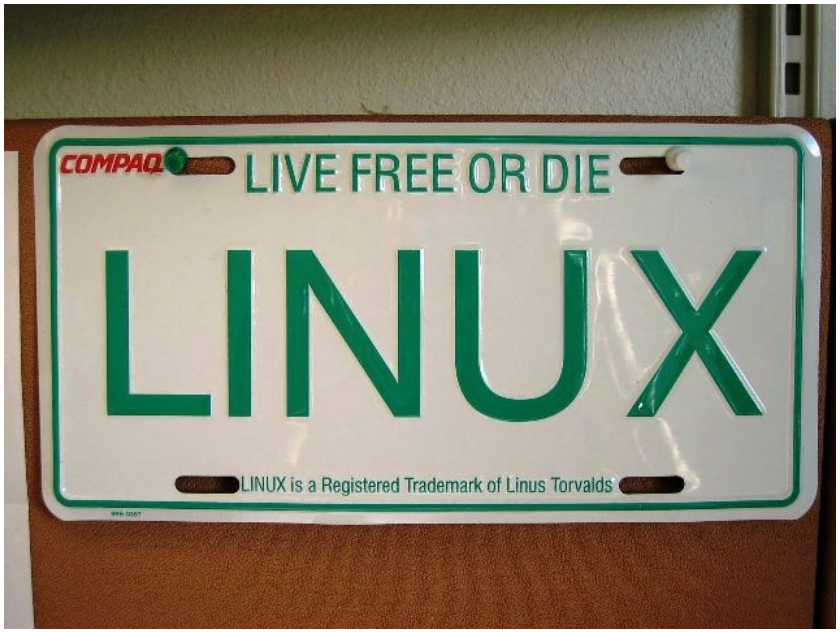

Sandia National Laboratories

# Outline

- Introduction
- Kitten Lightweight Kernel
- Palacios Virtual Machine Monitor
- Native vs. Guest OS results on Cray XT
- Conclusion

# Going on Four Decades of UNIX

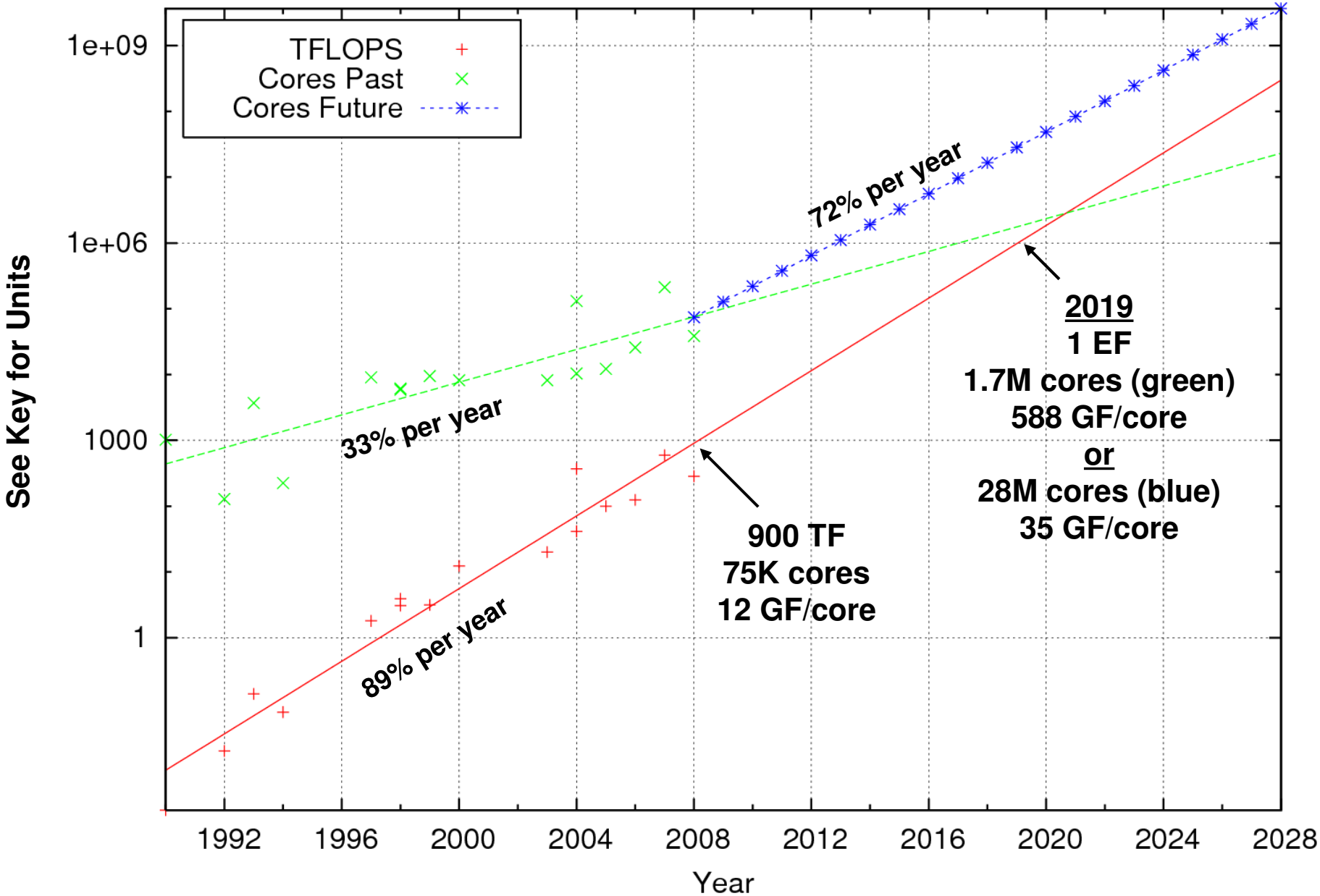



**Operating System = Collection of software and APIs**
**Users care about environment, not implementation details**
**LWK is about getting details right for scalability**

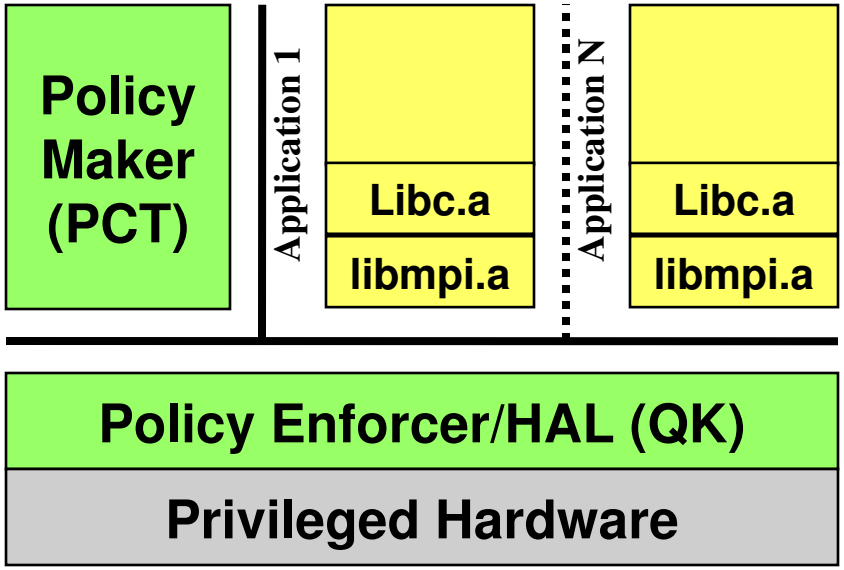
Sandia National Laboratories

# Challenge: Exponentially Increasing Parallelism

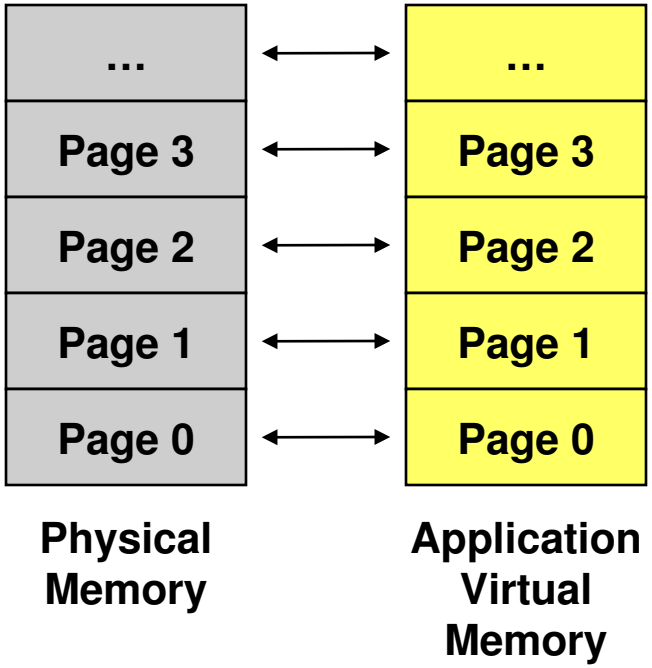


# LWK Overview

## Basic Architecture



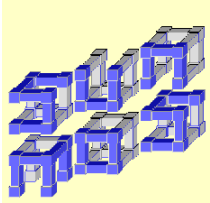

## Memory Management




- POSIX-like environment
- Inverted resource management
- Very low noise OS noise/jitter
- Straight-forward network stack (e.g., no pinning)
- Simplicity leads to reliability

Sandia National Laboratories

# Lightweight Kernel Timeline



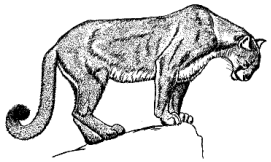
1991 – Sandia/UNM OS (SUNMOS), nCube-2

1991 – Linux 0.02

1993 – SUNMOS ported to Intel Paragon (1800 nodes)

1993 – SUNMOS experience used to design Puma

First implementation of Portals communication architecture



1994 – Linux 1.0

1995 – Puma ported to ASCI Red (4700 nodes)

Renamed Cougar, productized by Intel

Computational Plant

1997 – Stripped down Linux used on Cplant (2000 nodes)

Difficult to port Puma to COTS Alpha server

Included Portals API

2002 – Cougar ported to ASC Red Storm (13000 nodes)

Renamed Catamount, productized by Cray

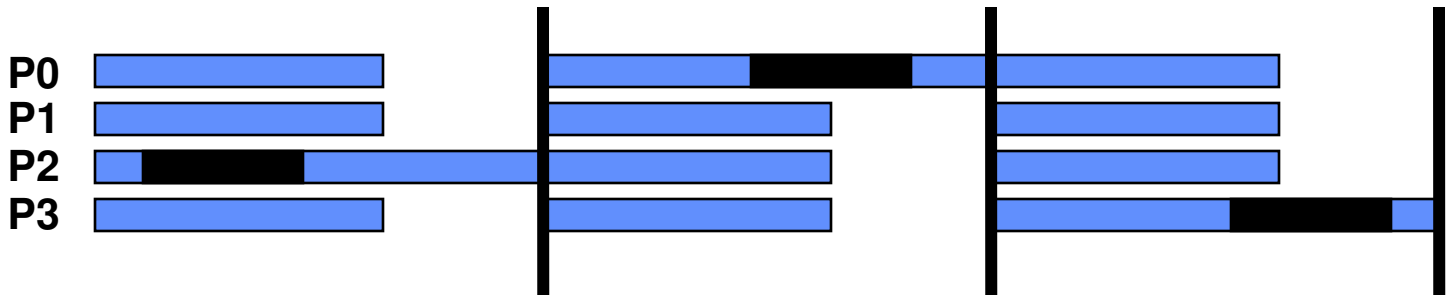
Host and NIC-based Portals implementations

2004 – IBM develops LWK (CNK) for BG/L/P (106000 nodes)

2005 – IBM & ETI develop LWK (C64) for Cyclops64 (160 cores/die)

Sandia National Laboratories

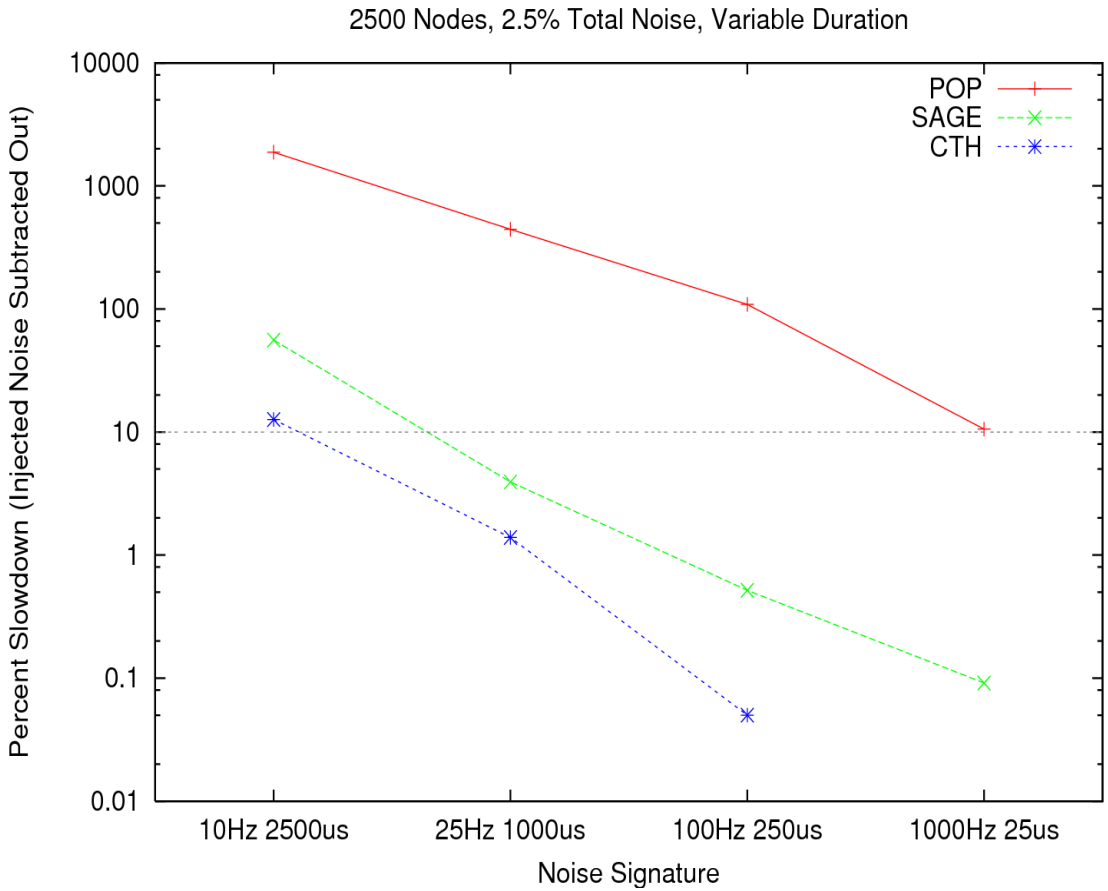
# We Know OS Noise Matters




- Impact of noise increases with scale (basic probability)
- Multi-core increases load on OS
- Idle noise measurements distort reality
  - Not asking OS to do anything
  - Micro-benchmark != real application

See "The Case of the Missing Supercomputer Performance", Petrini, et al.

# Red Storm Noise Injection Experiments



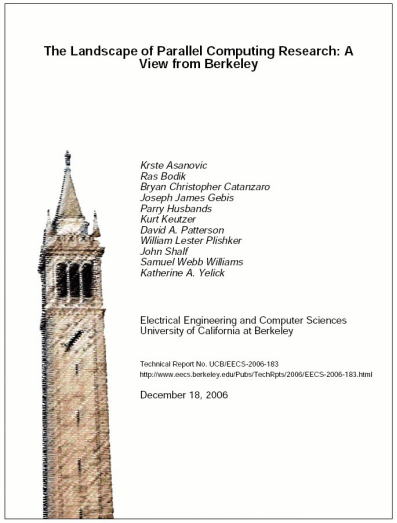

- **Result:**
  **Noise duration is more important than frequency**
- **OS should break up work into many small & short pieces**
- **Opposite of current efforts**
  - **Linux Dynaticks**
- **Cray CNL with 10 Hz timer had to revert back to 250 Hz due to OS noise duration issues**

From Kurt Ferreira's Masters Thesis

Sandia National Laboratories

# Drivers for LWK Compute Node OS

- Practical advantages
  - Low OS noise
  - Performance – tuned for scalability
  - Determinism – inverted resource management
  - Reliability
- Research advantages
  - Small and simple
  - Freedom to innovate (see “Berkeley View”)
    - Multi-core
    - Virtualization
  - Focused on capability systems
- Can’t separate OS from node-level architecture



The Landscape of Parallel Computing Research: A View from Berkeley

*Krste Asanovic*
*Ras Bodik*
*Bryan Christopher Catanzaro*
*Joseph James Gebis*
*Parry Husbands*
*Kurt Keutzer*
*David A. Patterson*
*William Lester Plishker*
*John Shalf*
*Samuel Webb Williams*
*Katherine A. Yelick*

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2006-183
http://www.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-183.html

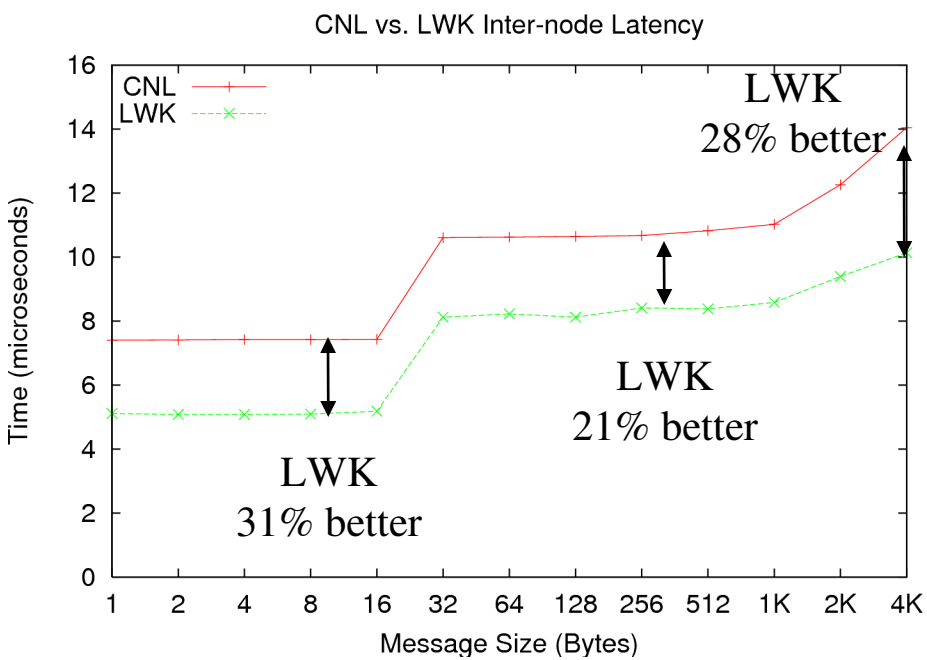
December 18, 2006

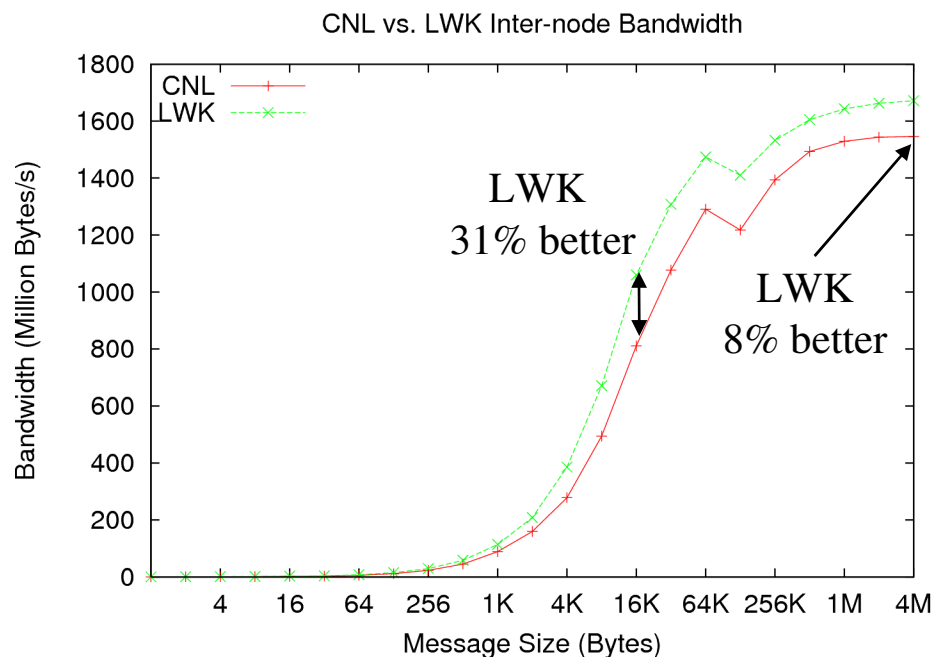
**Much simpler to create LWK than mainstream OS**

Sandia National Laboratories

# Architecture and System Software are Tightly Coupled

- **LWK's static, contiguous memory layout simplifies network stack**
  - **No pinning/unpinning overhead**
  - **Send address/length to SeaStar NIC**

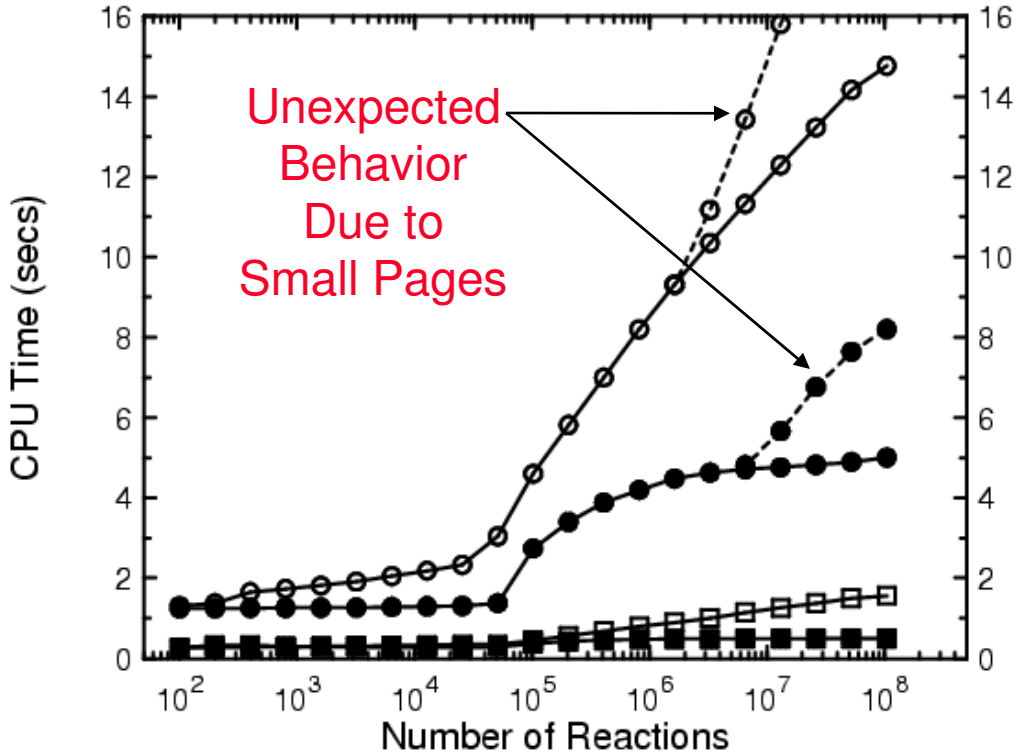



Host-based Network Stack (Generic Portals)
Testing Performed April 2008 at Sandia, UNICOS 2.0.44

Sandia National Laboratories

# TLB Gets in Way of Algorithm Research




Dashed Line =
Small pages

Solid Line =
Large pages
(Dual-core Opteron)

Open Shapes =
Existing Logarithmic Algorithm
(Gibson/Bruck)

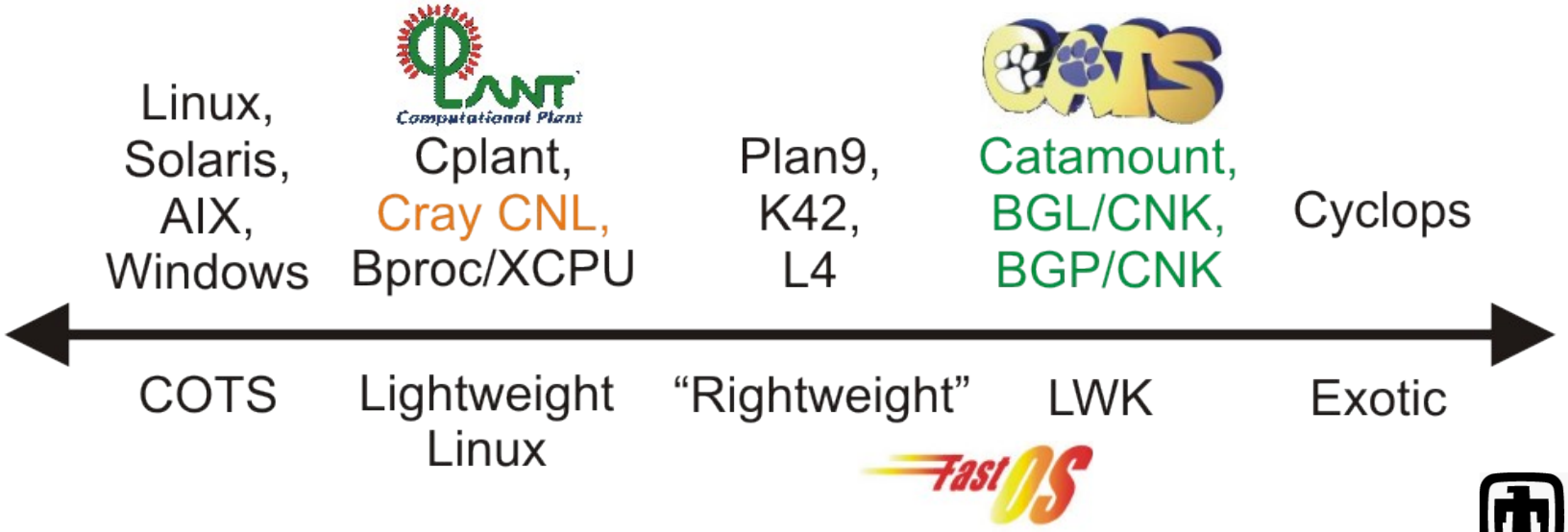
Solid Shapes =
New Constant-Time Algorithm
(Slepoy, Thompson, Plimpton)

**TLB misses increased with large pages,**
**but time to service miss decreased dramatically (10x).**
**Page table fits in L1! (vs. 2MB per GB with small pages)**

Sandia National Laboratories

# Project Kitten

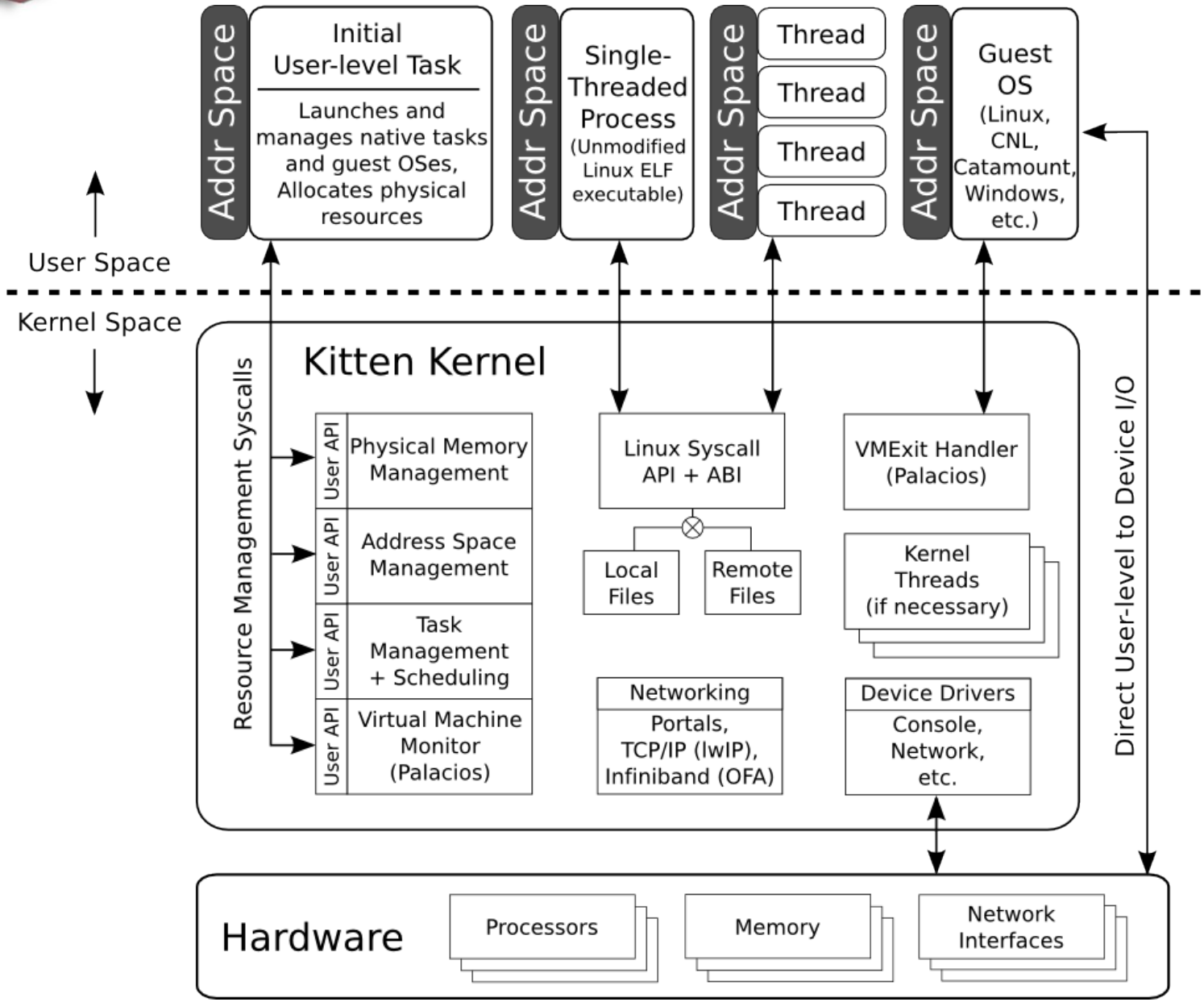
- **Creating modern open-source LWK platform**
  - Multi-core becoming MPP on a chip, requires innovation
  - Leverage hardware virtualization for flexibility
- **Retain scalability and determinism of Catamount**
- **Better match user and vendor expectations**
- **Available from http://software.sandia.gov/trac/kitten**



# Leverage Linux and Open Source

- **Repurpose basic functionality from Linux Kernel**
  - Hardware bootstrap
  - Basic OS kernel primitives
- **Innovate in key areas**
  - Memory management (Catamount-like)
  - Network stack
  - SMARTMAP
  - Fully tick-less operation, but short duration OS work
- **Aim for drop-in replacement for CNL**
- **Open platform more attractive to collaborators**
  - Collaborating with Northwestern Univ. and Univ. New Mexico on lightweight virtualization for HPC, http://v3vee.org/
  - Potential for wider impact

# Kitten Architecture



# Current Status

- Initial release (December 2008)
  - Single node, multi-core
  - Available from http://software.sandia.gov/trac/kitten
- Development trunk
  - Support for Glibc NPTL and GCC OpenMP via Linux ABI compatible clone(), futex(), ...
  - Palacios virtual machine monitor support (planning parallel Kitten and Palacios releases for May 1)
  - Kernel threads and local files for device drivers
- Private development trees
  - Catamount user-level for multi-node (yod, PCT, Catamount Glibc port, Libsysio, etc.)
  - Ported Open Fabrics Alliance IB stack

Sandia National Laboratories

# Virtualization Support

- Kitten optionally links with Palacios
  - Palacios developed by Jack Lange and Peter Dinda at Northwestern
  - Allows user-level Kitten applications to launch unmodified guest ISO images or disk images
  - Standard PC environment exposed to guest, even on Cray XT
  - Guests booted: Puppy Linux 3.0 (32-bit), Finnix 92.0 (64-bit), Compute Node Linux, Catamount
- "Lightweight Virtualization"
  - Physically contiguous memory allocated to guest
  - Pass-through devices (memory + interrupts)
  - Low noise, no timers or deferred work
  - Space-sharing rather than time-sharing

Sandia National Laboratories

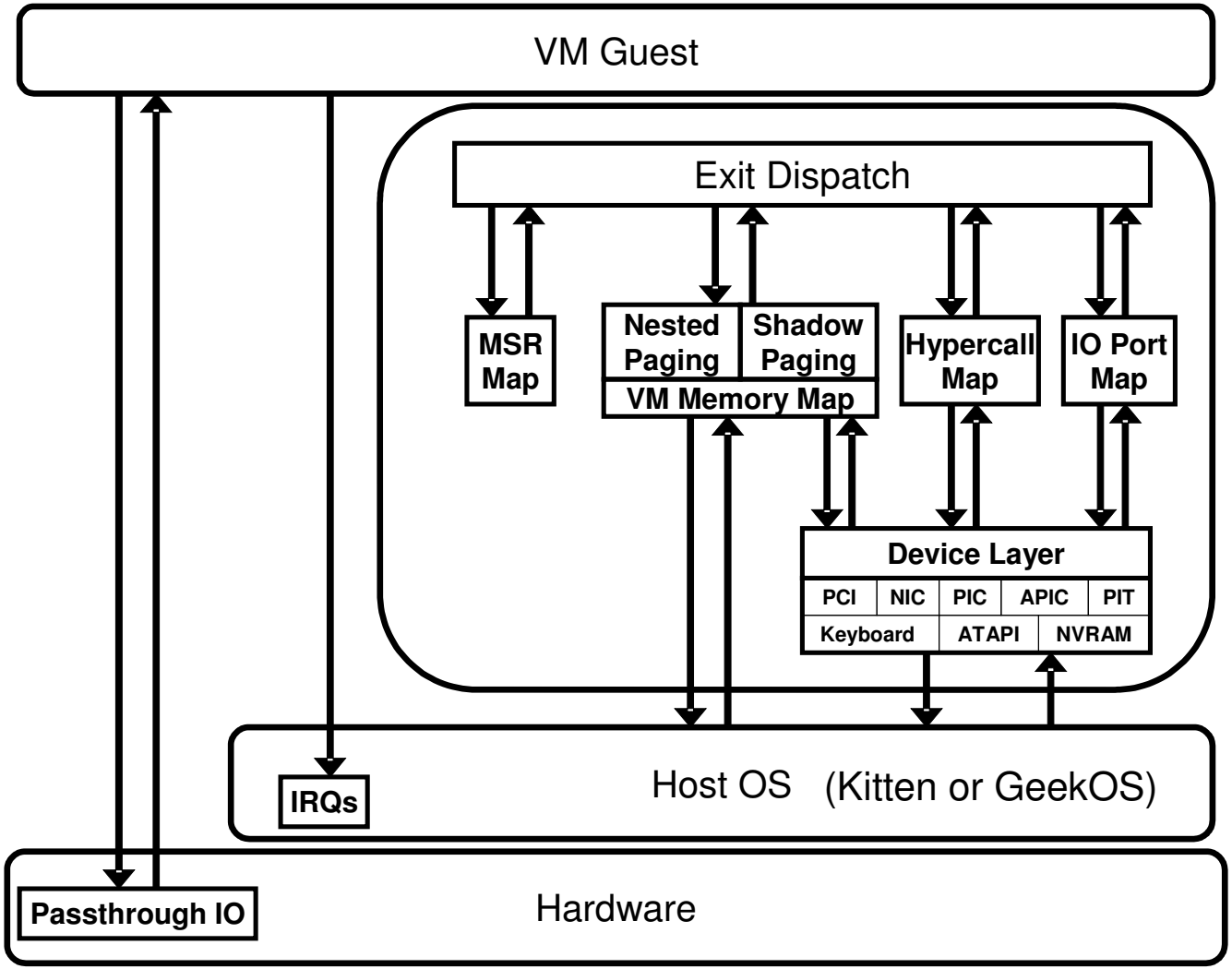
# Motivations for Virtualization in HPC

- Provide full-featured OS functionality in a lightweight kernel
  - Custom tailor OS to application (ConfigOS, JeOS)
  - Possibly augment guest OS's capabilities
- Improve resiliency
  - Node migration, full-system checkpointing
  - Enhanced debug capabilities
- Dynamic assignment of compute node roles
  - Individual jobs determine I/O node to compute node balance
  - No rebooting required
- Run-time system replacement
  - Capability run-time poor match for high-throughput serial workloads

Sandia National Laboratories

# Palacios Architecture

(credit: Jack Lange, Northwestern University)



# Shadow vs. Nested Paging: No Clear Winner

**Shadow Paging, O(N) mem accesses per TLB miss**

Palacios managed page tables used by the CPU

Page Faults

Page tables the guest OS thinks it is using

**Nested Paging, O(N^2) mem accesses per TLB miss**

Palacios managed guest phys to host phys page tables

CPU MMU

Guest OS managed guest virt to guest phys page tables

# Lines of Code in Kitten and Palacios

| Component | Lines of Code | |
|---|---|---|
| | sloccount . | wc *.c *.h *.s |
| **Kitten** | | |
| Kitten Core (C) | 17,995 | 29,540 |
| Kitten x86_64 Arch Code (C+Assembly) | 14,604 | 22,190 |
| Misc. Contrib Code (Kbuild + lwIP) | 27,973 | 39,593 |
| Palacios Glue Module (C) | 286 | 455 |
| Total | 60,858 | 91,778 |
| **Palacios** | | |
| Palacios Core (C+Assembly) | 15,084 | 24,710 |
| Palacios Virtual Devices (C) | 8,708 | 13,406 |
| XED Interface (C+Assembly) | 4,320 | 7,712 |
| Total | 28,112 | 45,828 |
| Grand Total | 88,970 | 137,606 |

Sandia National Laboratories

# Kitten+Palacios on Cray XT

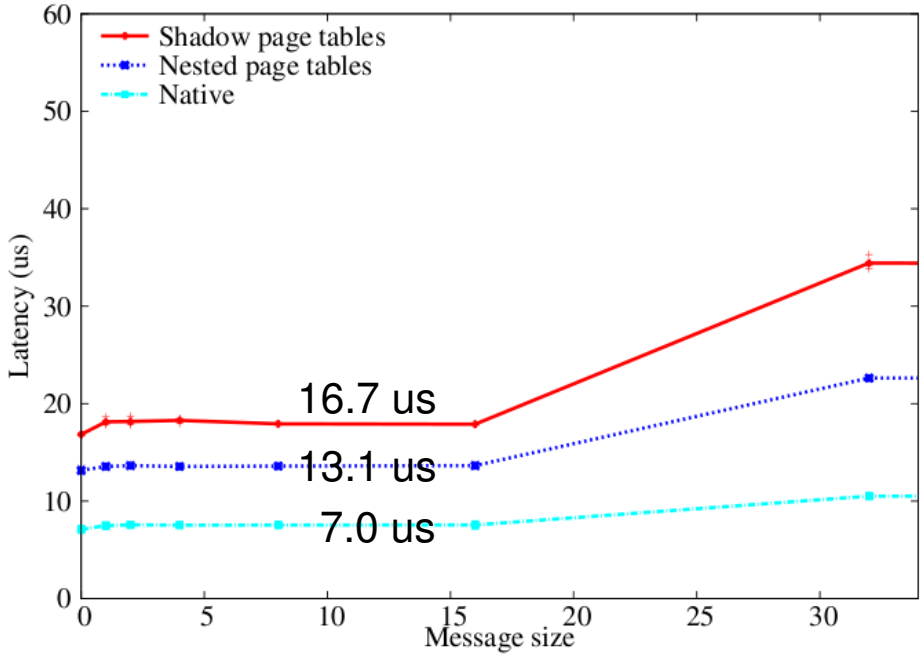
- Kitten boots as drop-in replacement for CNL
  - Kitten kernel vmlwk.bin -> vmlinux
  - Kitten initial task ELF binary -> initramfs
  - Kernel command-line args passed via parameters file
- Guest OS ISO image embedded in Kitten initial task
  - Kitten boots, starts user-level initial task, initial task "boots" the embedded guest OS
  - Both CNL and Catamount ported to the standard PC environment that Palacios exposes
- SeaStar direct-mapped through to guest
  - SeaStar 2 MB device window direct mapped to guest physical memory
  - SeaStar interrupts delivered to Kitten, Kitten forwards to Palacios, Palacios injects into guest

Sandia National Laboratories

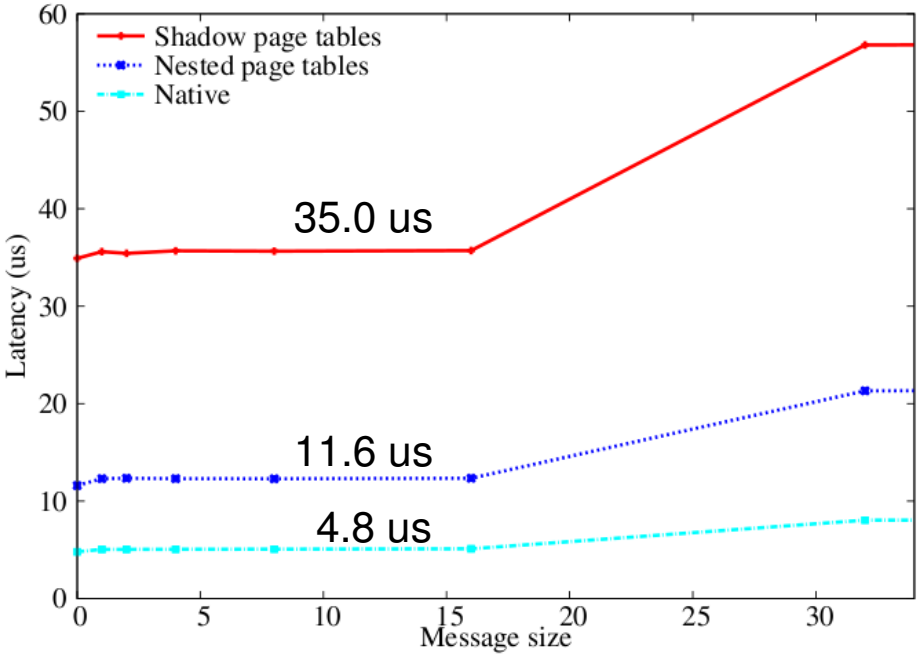
# Native vs. Guest
# CNL and Catamount Tests

- Testing performed on rsqual XT4 system at Sandia
  - Single cabinet, 48 2.2 GHz quad-core nodes
  - Developers have reboot capability
- Benchmarks:
  - Intel Messaging Benchmarks (IMB, formerly Pallas)
  - HPCCG "Mini-application"
    - Sparse CG solver
    - 100 x 100 x 100 problem, ~400 MB per node
  - CTH Application
    - Shock physics, important Sandia application
    - Shaped charge test problem (no AMR)
    - Weakly scaled

# IMB PingPong Latency: Nested Paging has Lowest Overhead




Compute Node Linux

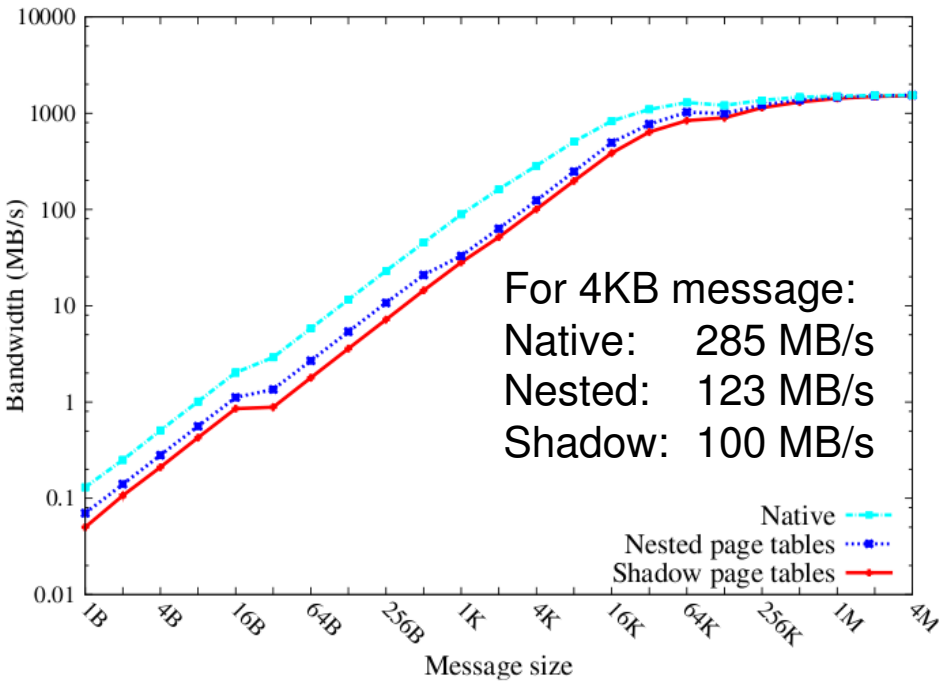



Catamount

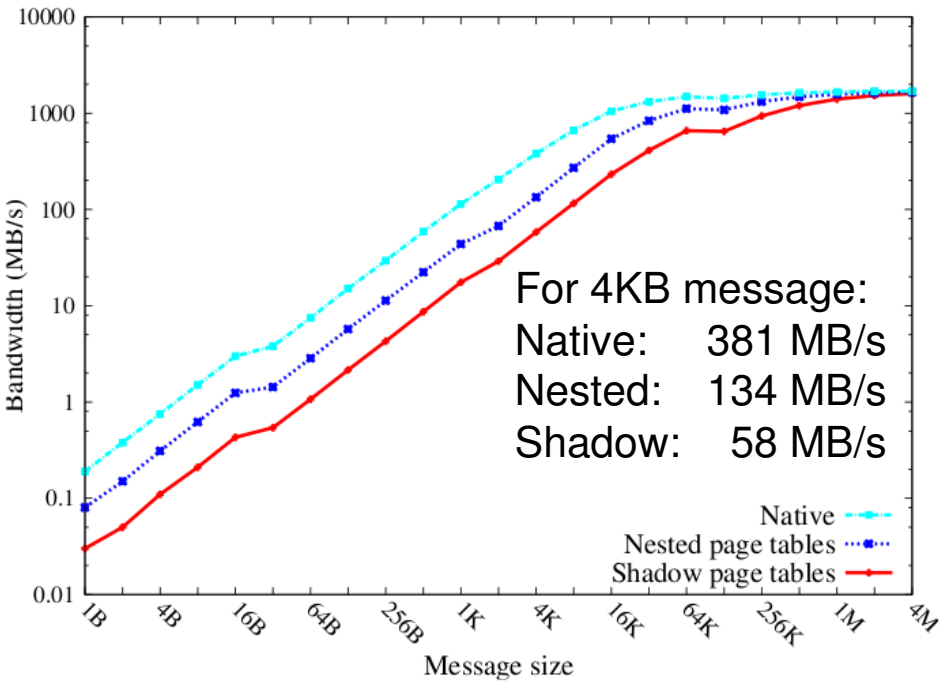
Still investigating cause of poor performance of shadow paging on Catamount. Likely due to overhead/bug in emulating guest 2 MB pages for pass-through memory-mapped devices.

Sandia National Laboratories

# IMB PingPong Bandwidth: All Cases Converge to Same Peak Bandwidth



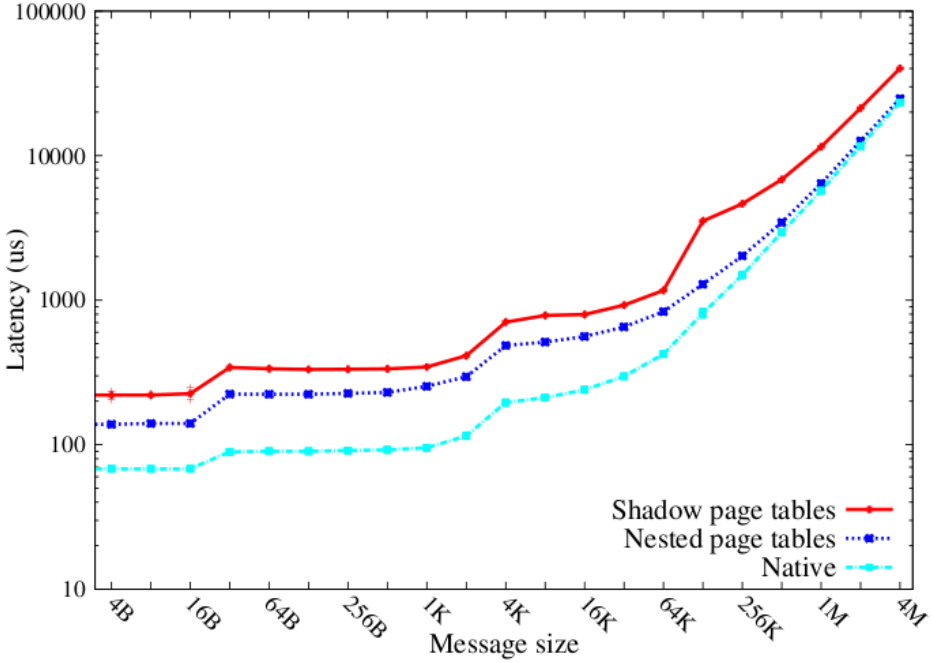

Compute Node Linux



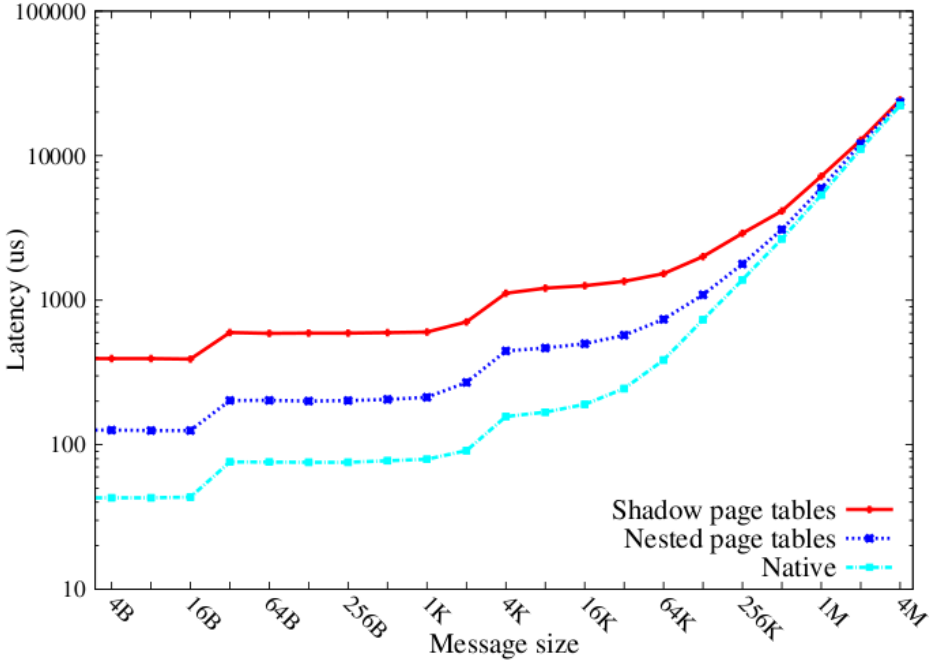

Catamount

Sandia National Laboratories

# 48-Node IMB Allreduce Latency: Nested Paging Wins, Most Converge at Large Message Sizes



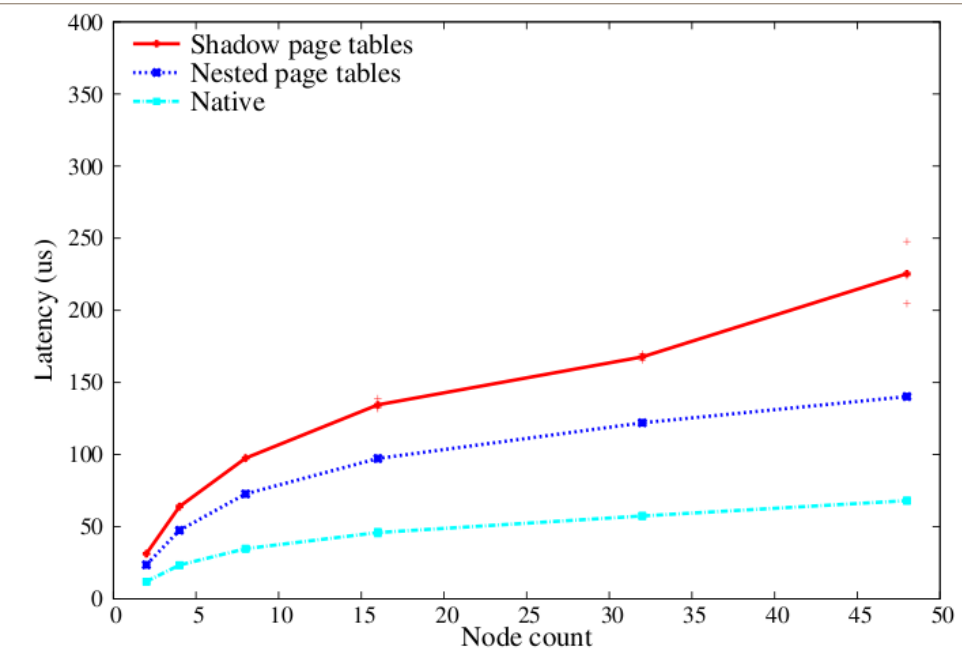

Compute Node Linux



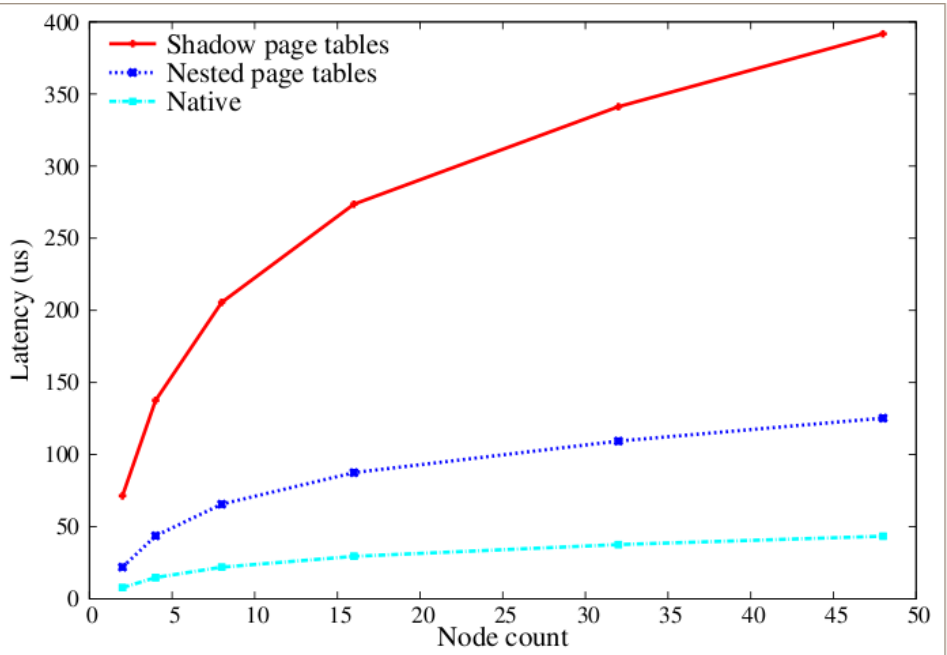

Catamount

Sandia National Laboratories

# 16-byte IMB Allreduce Scaling: Native and Nested Paging Scale Similarly




Compute Node Linux

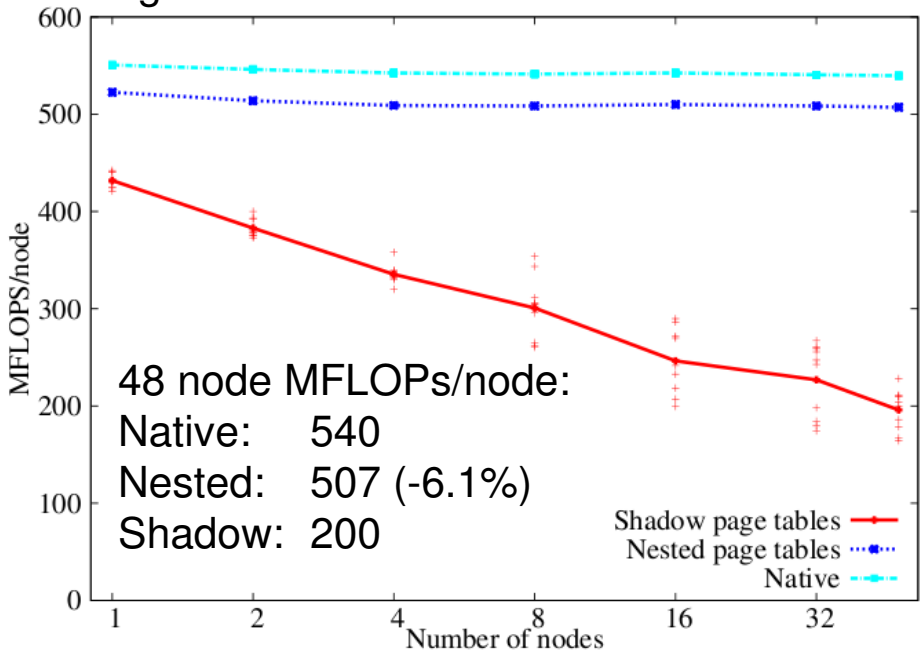



Catamount

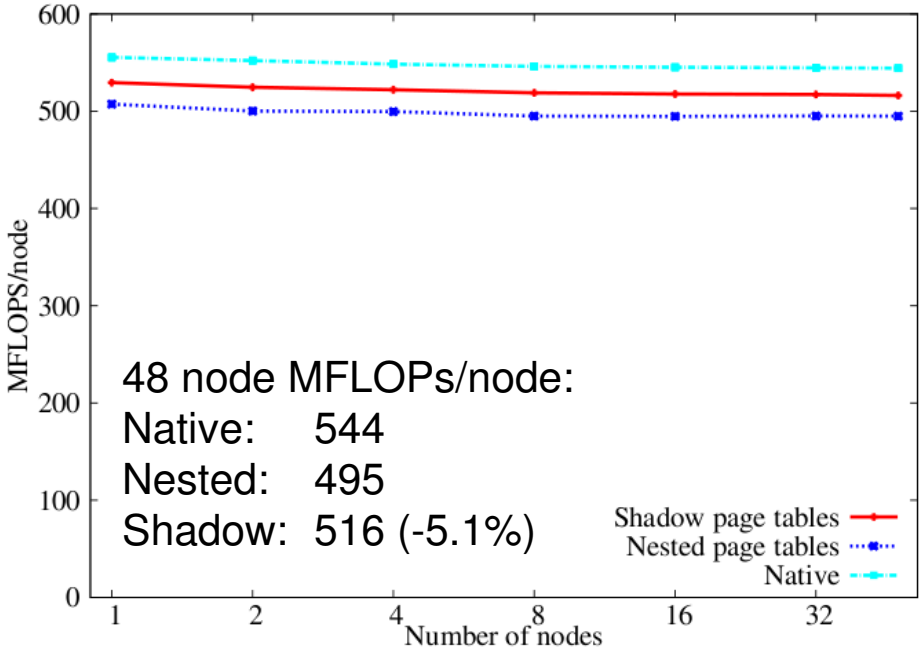
Sandia National Laboratories

# HPCCG Scaling:
# 5-6% Virtualization Overhead
# Shadow faster than Nested on Catamount

Higher is Better




Compute Node Linux




Catamount

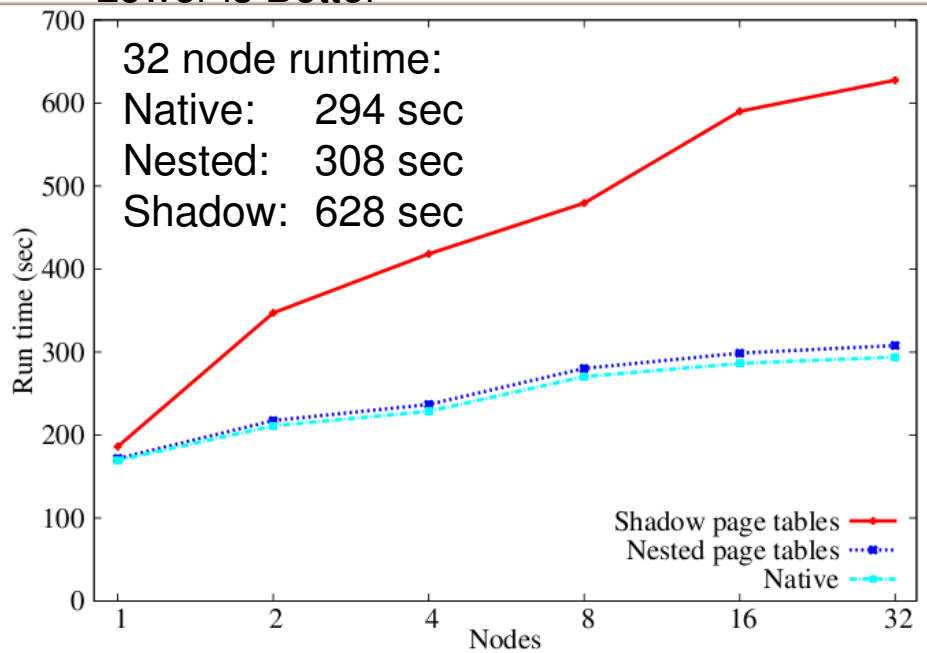
Poor performance of shadow paging on CNL due to context switching. Could be avoided by adding page table caching to Palacios.

Catamount is essentially doing no context switching, benefiting shadow paging (2n vs. n^2 page table depth issue discussed earlier)

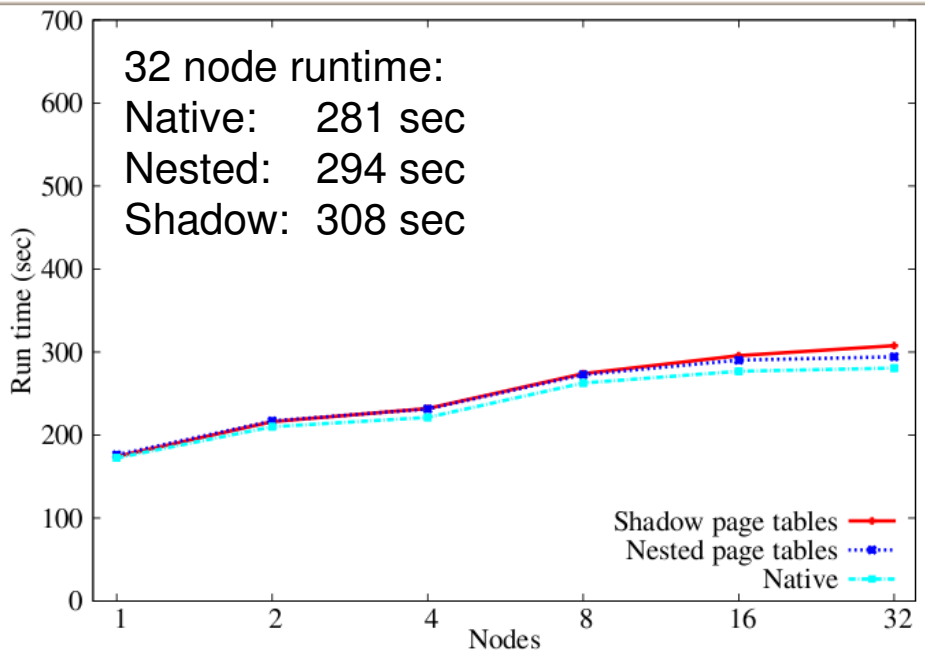
Sandia National Laboratories

# CTH Scaling:
# < 5% Virtualization Overhead
# Nested faster than Shadow on Catamount

Lower is Better




Compute Node Linux




Catamount

Poor performance of shadow paging on CNL due to context switching.
Could be avoided by adding page table caching to Palacios.

Sandia National Laboratories

# Conclusion

- Kitten LWK is in active development
  - Runs on Cray XT and standard PC hardware
  - Guest OS support when combined with Palacios
  - Available now, open-source
- Virtualization experiments on Cray XT indicate ~5% performance overhead for CTH application
  - Would like to do larger scale testing
  - Accelerated portals may further reduce overhead