Quad-core Catamount and R&D in Multi-core Lightweight Kernels

Salishan Conference on High-Speed Computing
Gleneden Beach, Oregon
April 21-24, 2008

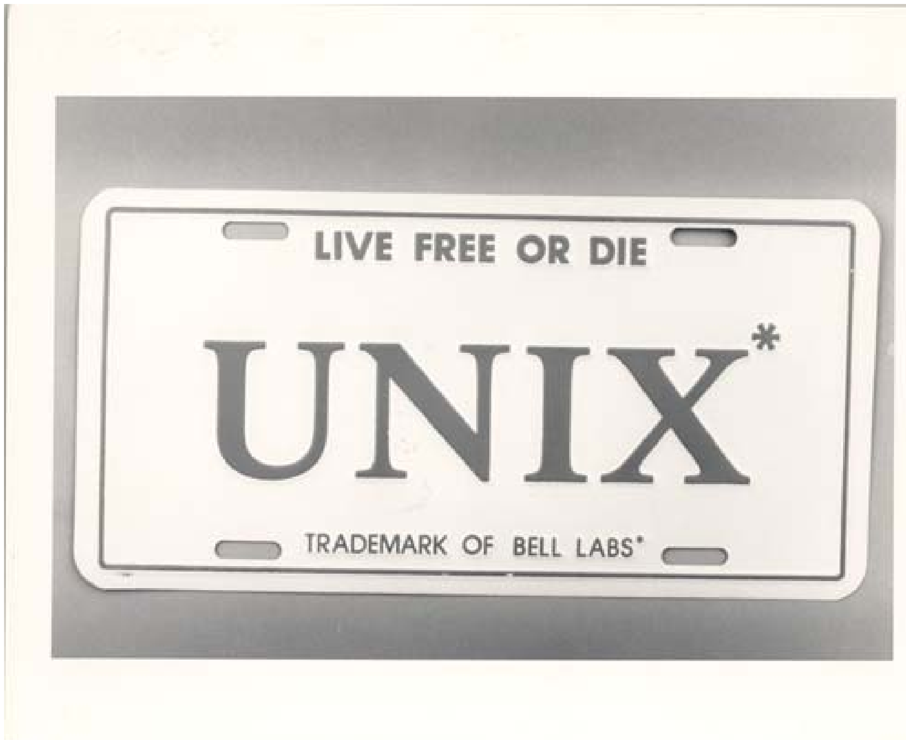
Kevin Pedretti
Senior Member of Technical Staff
Scalable System Software, Dept. 1423
ktpedre@sandia.gov

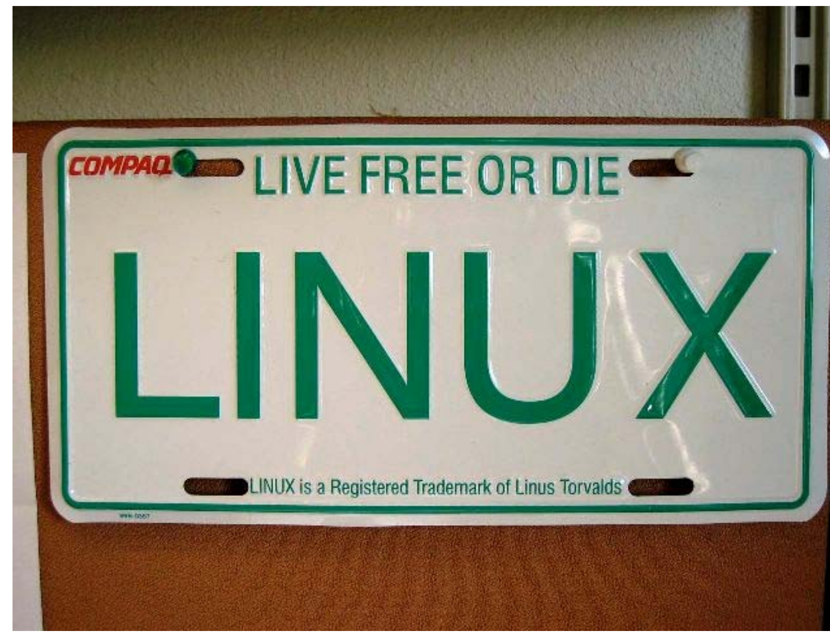
SAND Number: 2008-1725A

Outline

- **Introduction**
- **Quad-core Catamount LWK results**
- **Open-source LWK**
- **Research directions**
- **Conclusion**

Going on Four Decades of UNIX



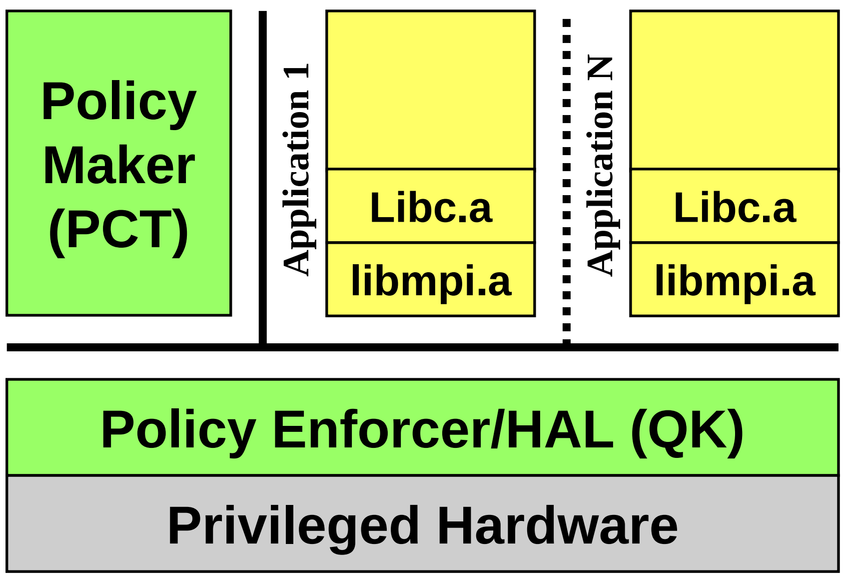


Operating System = Collection of software and APIs
Users care about environment, not implementation details
LWK is about getting details right for scalability

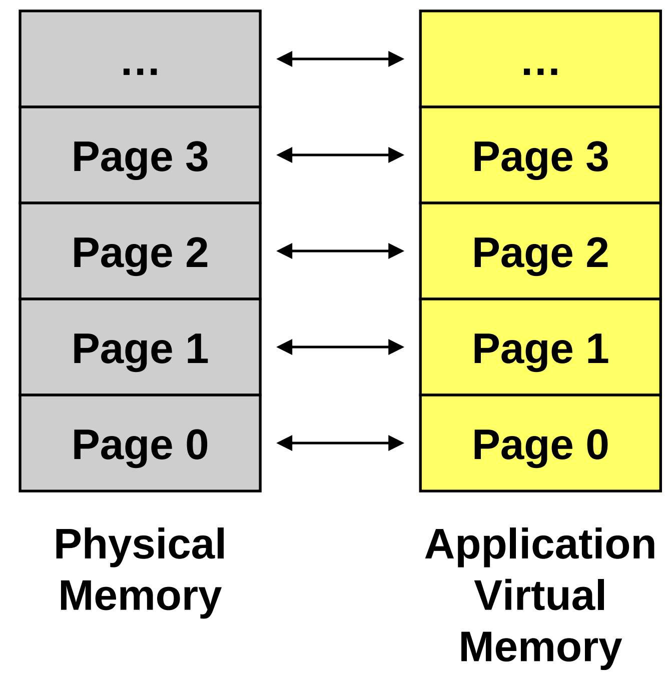
Sandia National Laboratories

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

Lightweight Kernel Timeline

Nov 2007 Top500 Top 10 System <u>Compute Processors:</u> 82% run a LWK

1990 – Sandia/UNM OS (SUNMOS), nCube-2

1991 – SUNMOS ported to Intel Paragon (1800 nodes)

1991 – Linux 0.02

1993 – SUNMOS enhanced, becomes Puma

- **First implementation of Portals communication architecture**

1994 – Linux 1.0

1995 – Puma ported to ASCI Red (4700 nodes)

- **Renamed Cougar, productized by Intel**

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)

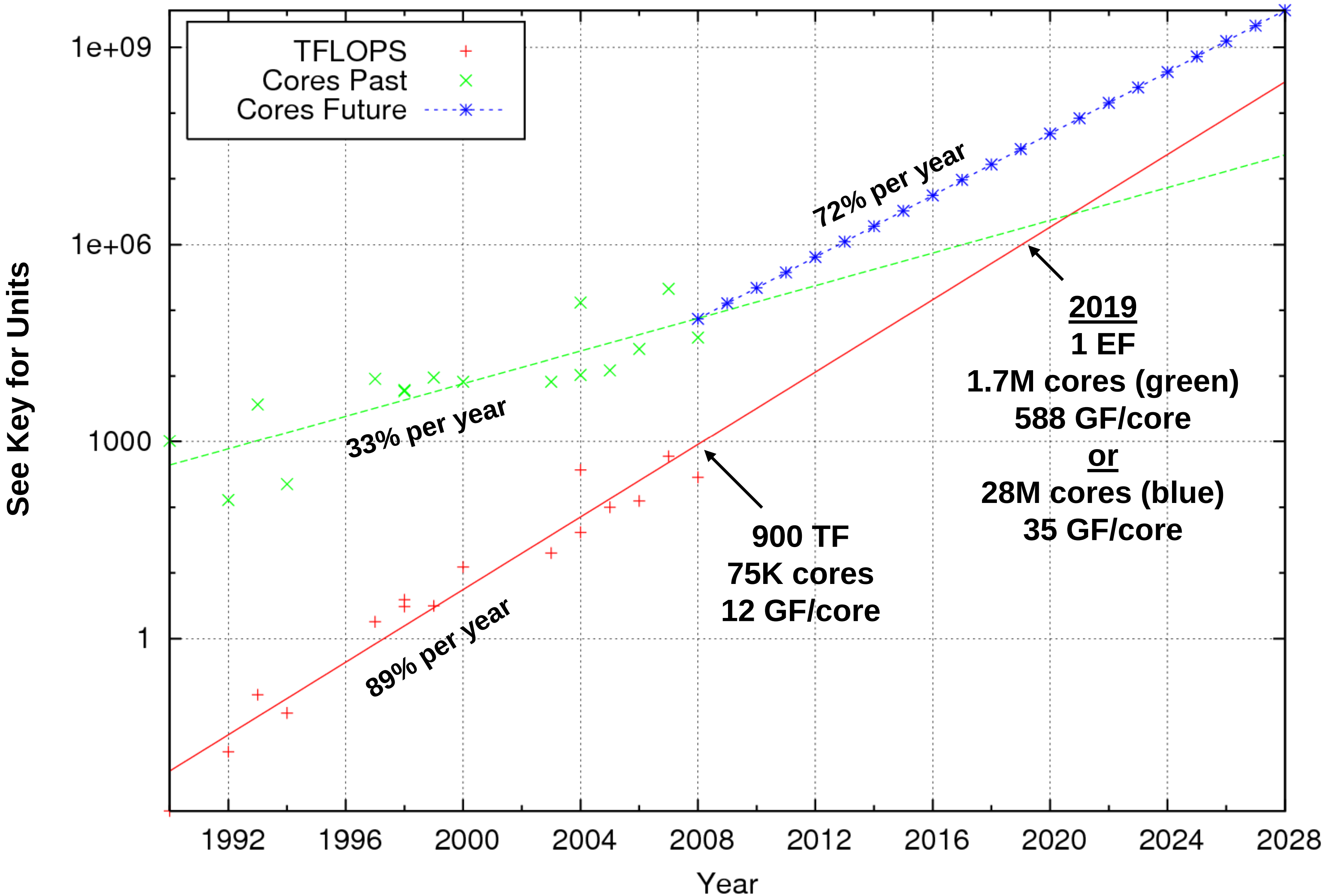
- **Renamed Catamount, productized by Cray**
- **Host and NIC-based Portals implementations**

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

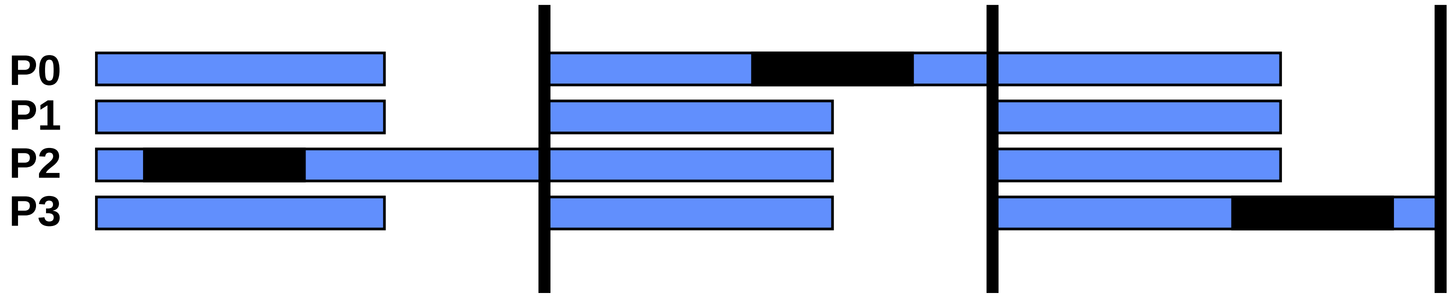
2005 – IBM & ETI develop LWK (C64) for Cyclops64 (160 cores/die)

Sandia National Laboratories

Challenge: Exponentially Increasing Parallelism



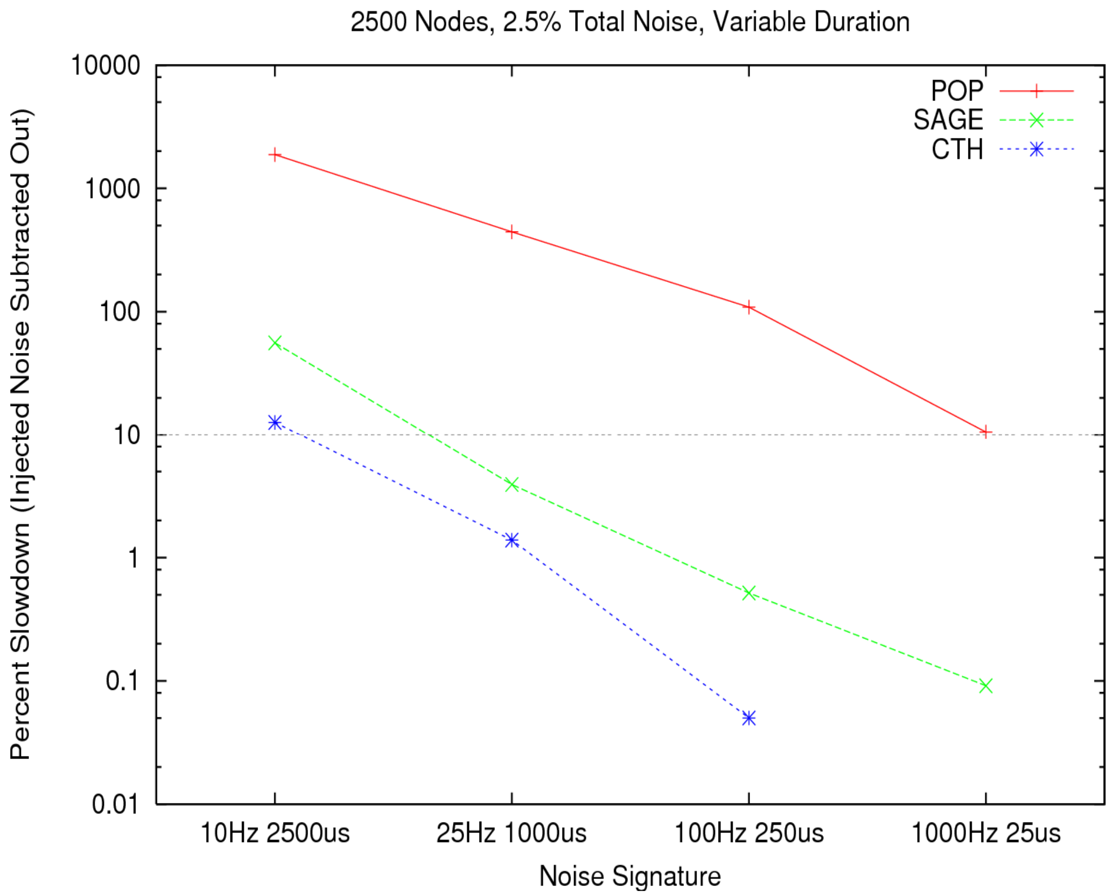
We Know OS Noise Matters



- 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



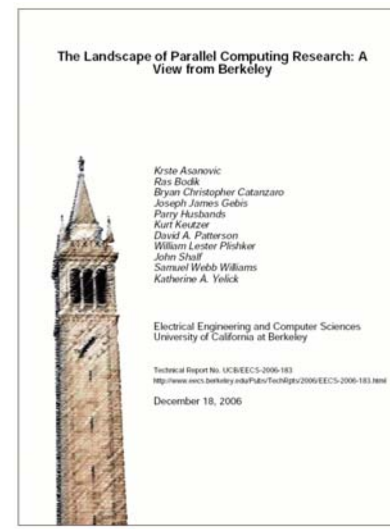
- **Result:**
  Noise duration is more important than frequency
- **OS should break up work into many small & short pieces**
- **Opposite of current efforts**
  - **Linux Dynaticks**
- **Cray CNL with 10 Hz timer had to revert back to 250 Hz due to OS noise duration issues**

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

Outline

- Introduction
- Quad-core Catamount LWK results
- Open-source LWK
- Research directions
- Conclusion

Sandia National Laboratories

Quad-core Catamount

- **Risk mitigation for ORNL Jaguar System**
  - Plan of record: CNL + ALPS
  - Backup plan: Quad-core Catamount
- **Funded by DOE Office of Science and ORNL**
  - PI: Sue Kelly;
    John VanDyke, Courtenay Vaughan
  - Project complete, fully functional
  - Will be used for Red Storm quad-core upgrade:
    38400 cores, 284 TFLOPS
- **Results discussed:**
  - Large-scale dual-core CNL vs. Catamount
  - Small-scale quad-core performance

Sandia National Laboratories

Large-scale Dual-core CNL vs. Catamount

	CNL 2.0.03+	Catamount 2.0.05+	CNL vs. Catamount
	PGI 6.1.6	PGI 6.1.3	% CNL worse
GTC			
1024 XT3 only	595.6	584.0	2.0
4096 XT3 only	614.6	593.8	3.5
20000 XT3/XT4	786.5	778.9	1.0
VH1			
1024 XT3 only	22.7	20.9	8.6
4096 XT3 only	137.1	117.4	16.8
20000 XT3/XT4	1186.0	981.7	20.8
POP			
4800 XT3 only	90.6	77.6	16.8
20000 XT3/XT4	98.8	75.2	31.4

Testing performed June 16-17, 2007 at ORNL

- Apps important to ORNL
- Time ran out before LSMS and S3D problems diagnosed
- Catamount apps did not link with IOBUF library

Small-scale Quad-core CNL vs. Catamount

Application	# MPI Ranks	Cores per Node	CNL (time units, lower better)	Catamount (time units, lower better)	(CNL/Catamount - 1) * 100%
GTC	16	4	664.9	670.6	-0.8
S3D	16	4	1949.1	1948.9	0.0
POP	16	4	153.8	151.9	1.3
LSMS	16	4	290.1	276.8	4.8
SPPM	16	4	847.8	845.0	0.3
UMT	16	4	8.4	7.9	6.4
PRONTO	16	4	241.5	222.0	8.8
SAGE	16	4	267.8	234.9	14.0
CTH	16	4	15.1	13.0	16.6
PARTISN	16	4	43.2	35.7	21.0

Disclaimer: Some test problems were small

Testing performed April, 2008 at Sandia

- Four nodes, 2.2 GHz quad-core, rev. B2
- UNICOS 2.0.44
- 4 KB pages CNL, 2 MB Catamount
- VH1 wouldn't run on CNL

Sandia National Laboratories

Catamount Quad-core Cores Effectively Used

Application	Utilization of each Core	Cores Effectively Used
PARTISN	40%	1.60
CTH	71%	2.84
SAGE	74%	2.95
PRONTO	79%	3.18
UMT2K	91%	3.62

Disclaimer: UMT2K problem was possibly small, others reasonable

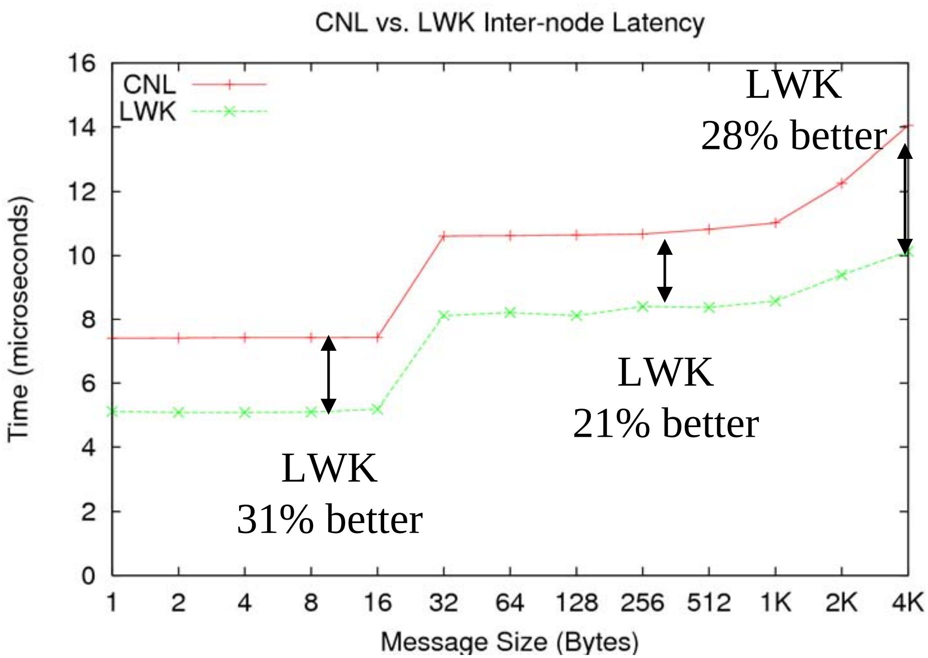
Calculation:

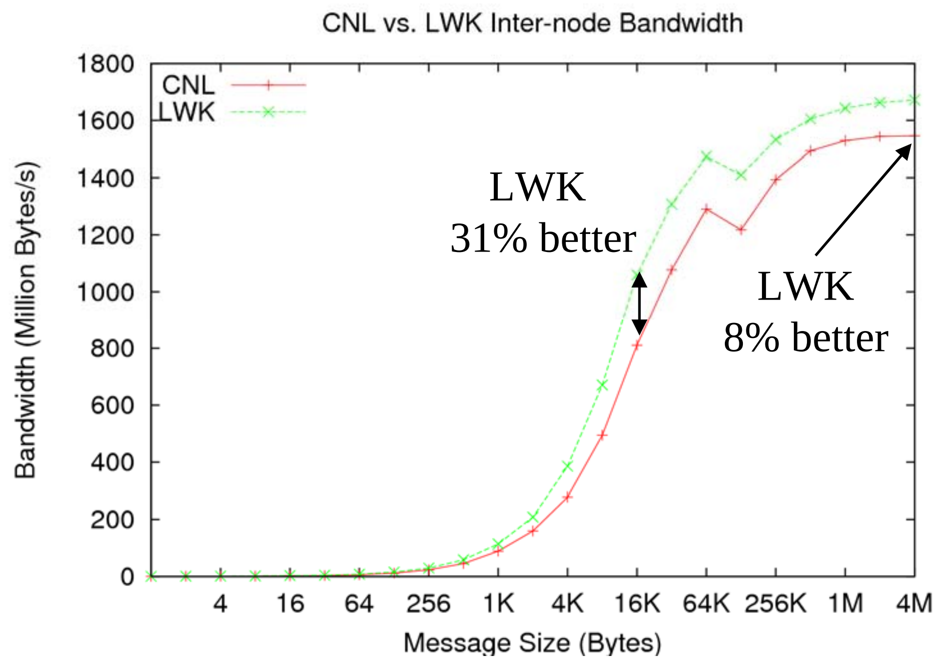
- 4 core runs, either 1 core per node (S) or 4 cores per node (Q)
- Assume S takes 1 hr. and Q takes .85 hours
- Assume S using 100% of core
- Q is effectively using .85 * 4 = 3.4 of each core

Sandia National Laboratories

Quad-Core Catamount Network Stack Performance

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

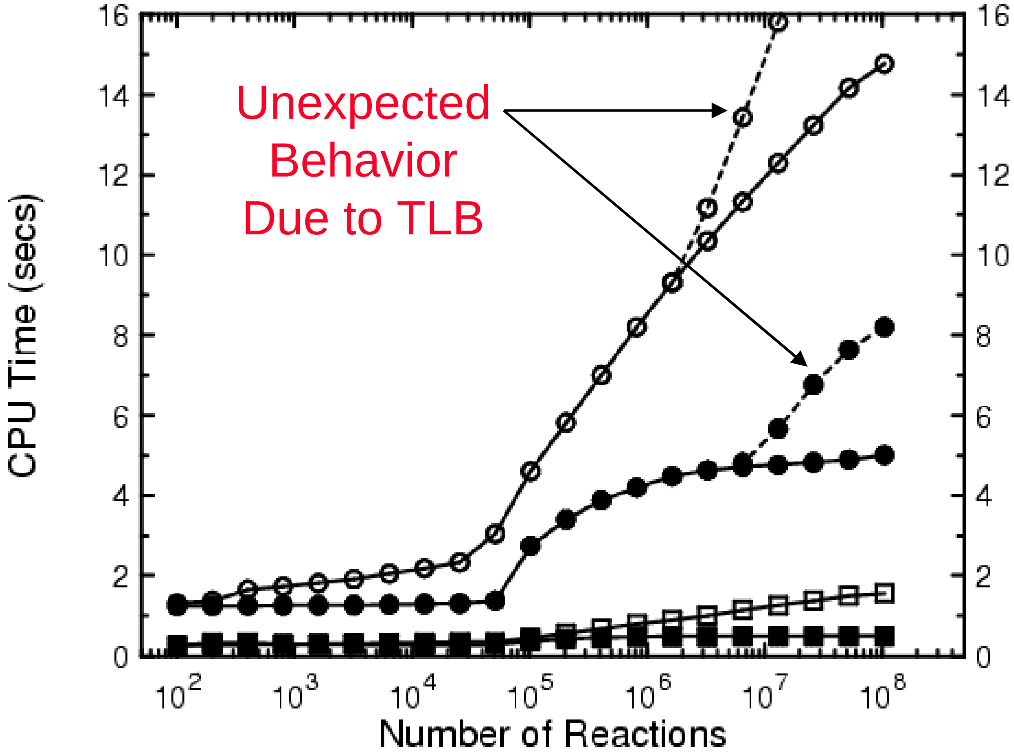




Host-based Network Stack (Generic Portals)
Testing Performed April 2008 at Sandia, UNICOS 2.0.44

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)

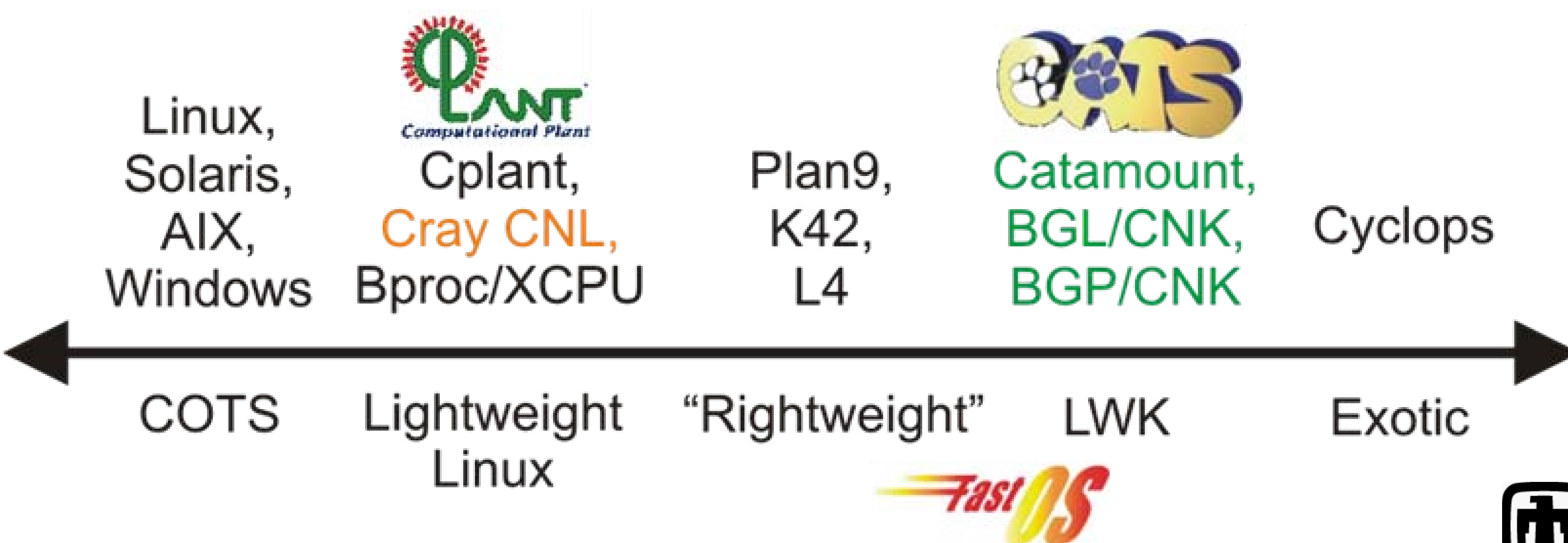
Sandia National Laboratories

Outline

- Introduction
- Quad-core Catamount LWK results
- Open-source LWK
- Research directions
- Conclusion

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**



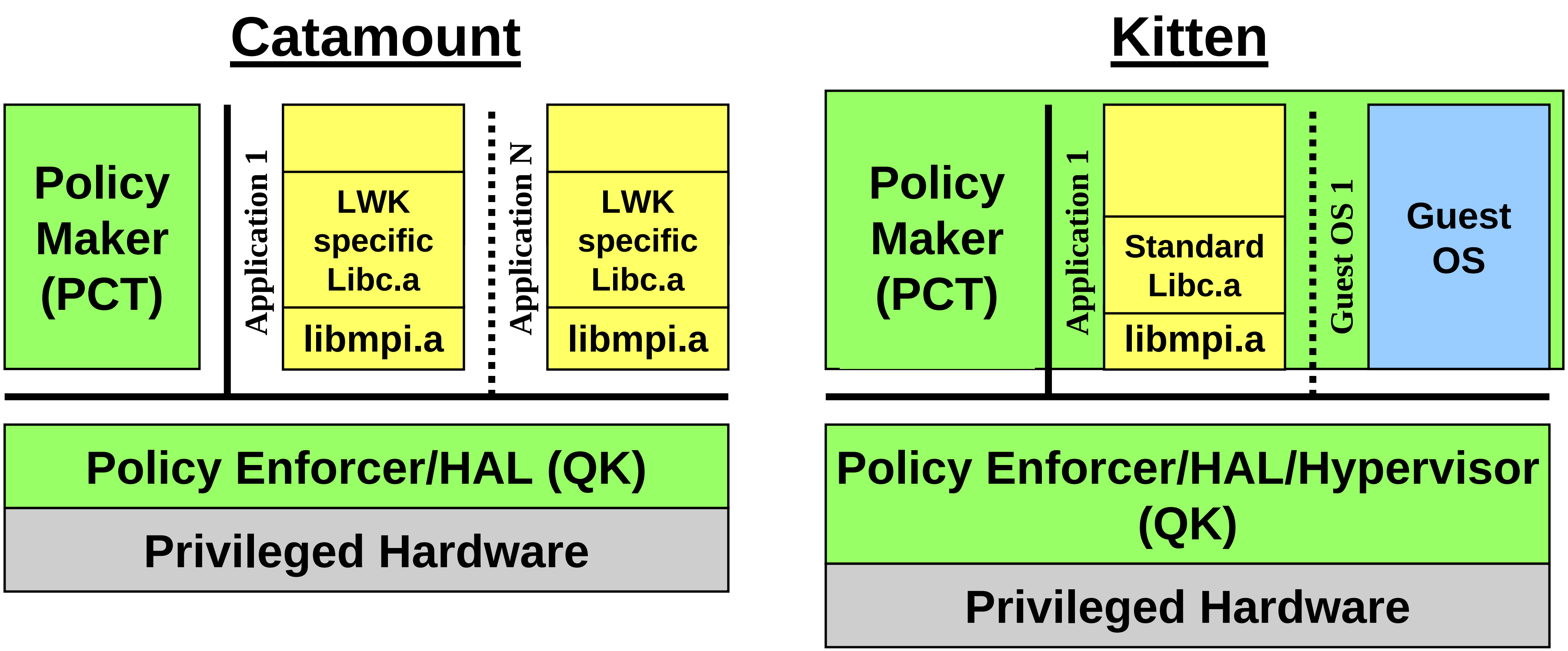
Sandia National Laboratories

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
  - Fully tick-less operation, but short duration OS work
- **Aim for drop-in replacement for CNL**
- **Open platform more attractive to collaborators**
  - Northwestern and UNM adding their V3VEE lightweight hypervisor to Kitten (NSF funded)
  - Potential for wider impact

Sandia National Laboratories

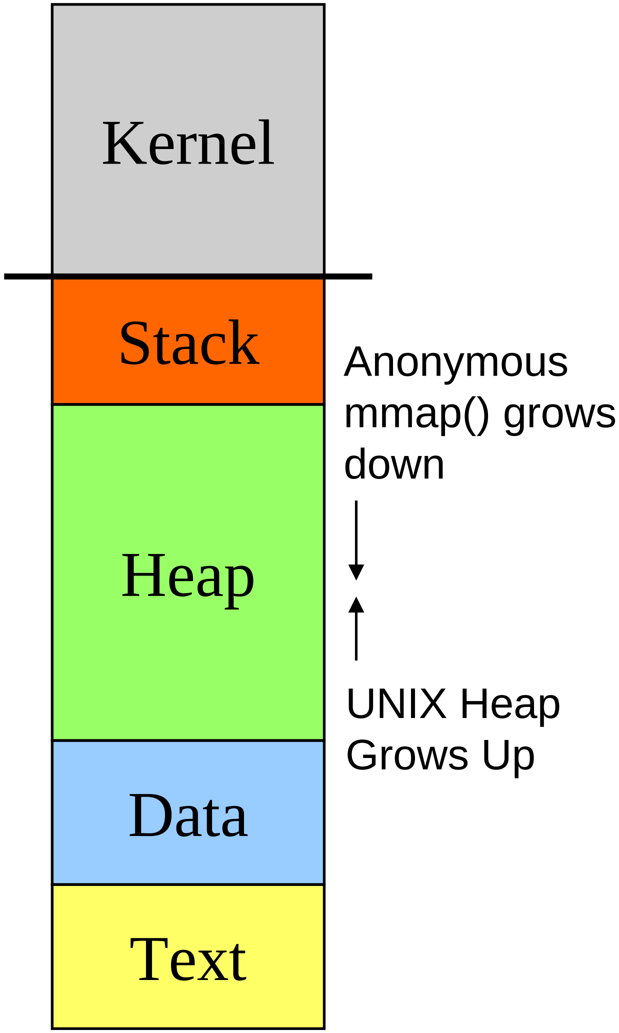
LWK Architecture



Major changes:

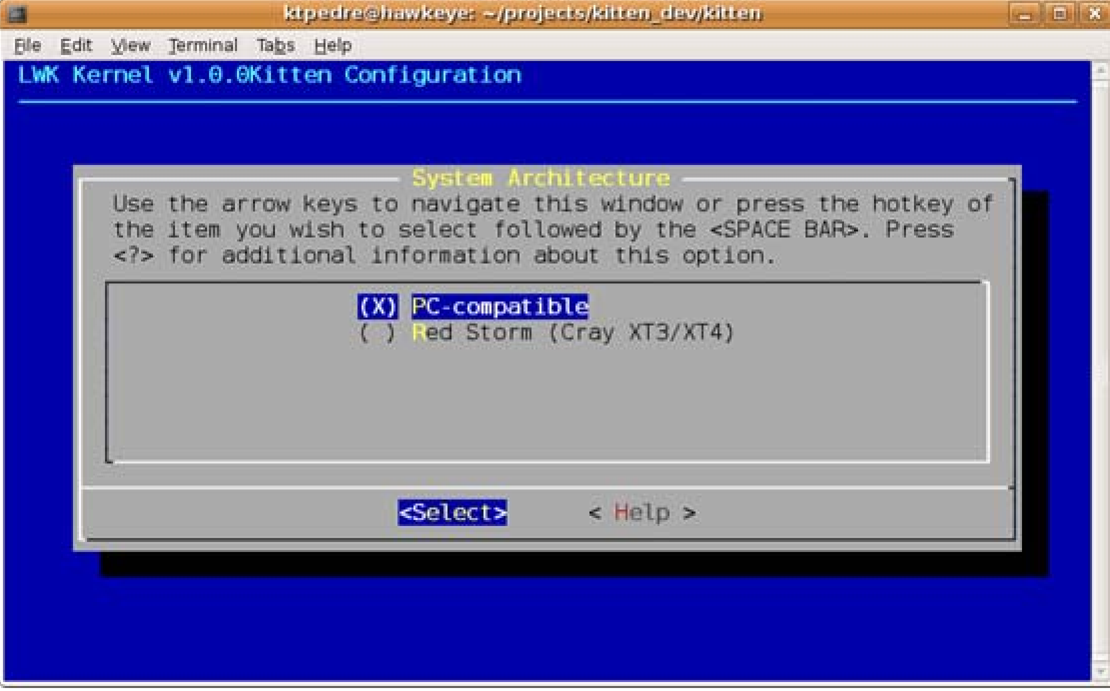
- QK includes hypervisor functionality
- QK provides Linux ABI interface, relay to PCT
- PCT provides function shipping, rather than special libc.a

Status



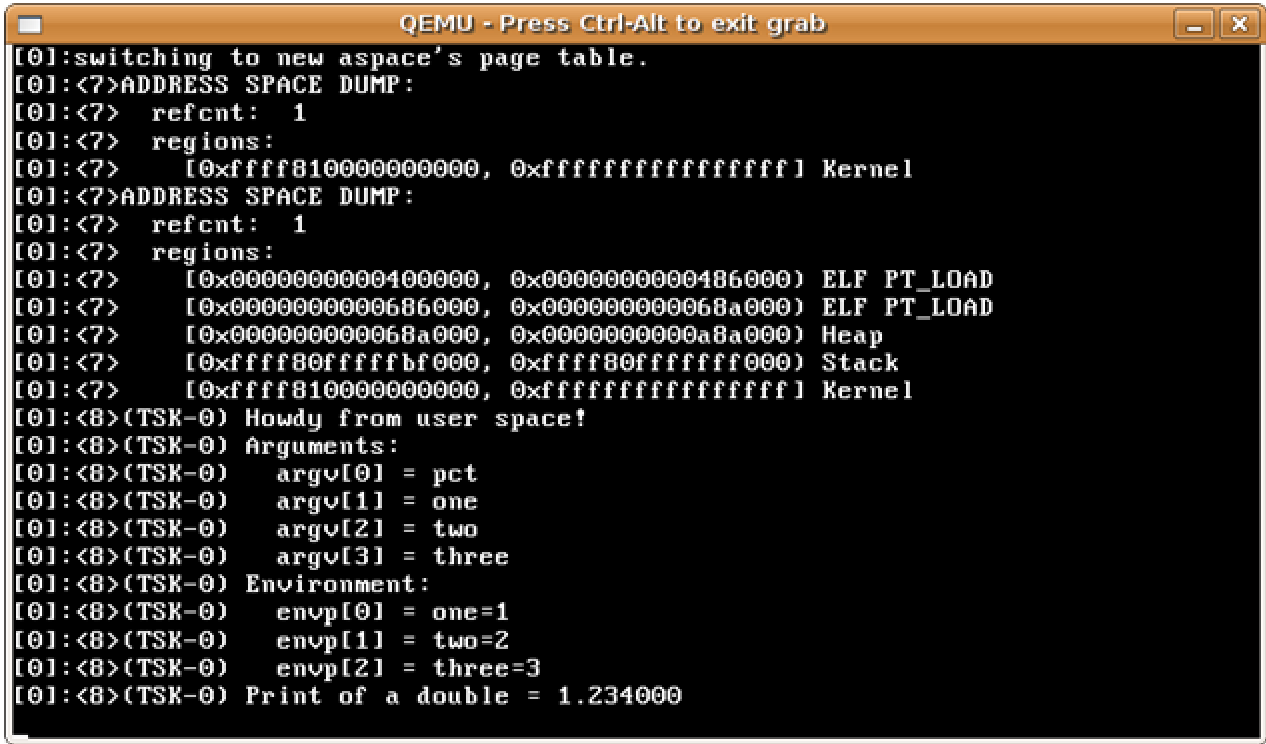
- **X86-64 support**
- **Linux ABI**
  - **Basic system calls**
  - **Initial user-stack setup**
  - **Thread Local Storage (TLS)**
  - **Virtual system calls**
- **Boots on Red Storm**
  - **Drop-in CNL replacement**
  - **Console I/O**
  - **Portals network stack**
- **Initrd treated as PCT (ELF image)**
- **Runs STREAM compiled with standard Linux toolchain**
- **DOE approved for open source release (GPL)**

Sandia National Laboratories




```
make bzImage
make isoimage
```

```
kvm -cdrom image.iso
```



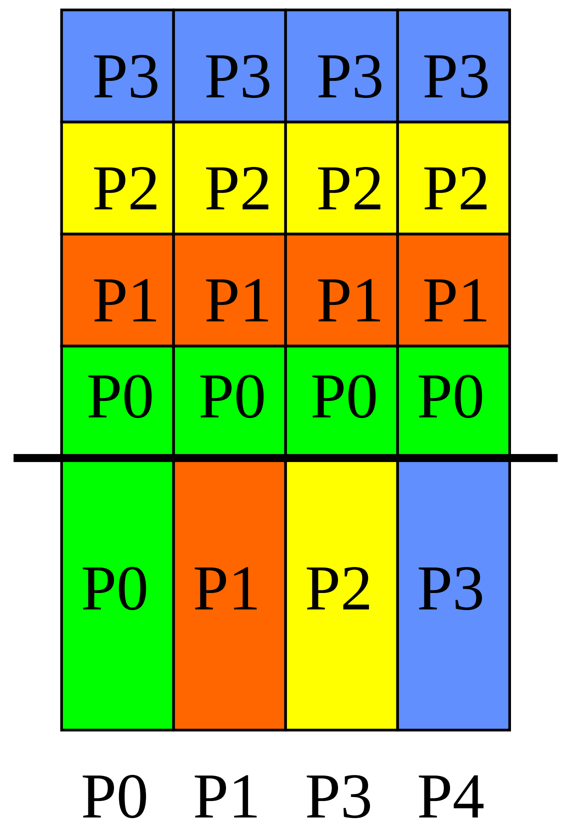
Outline

- Introduction
- Quad-core Catamount LWK results
- Open-source LWK
- Research directions
- Conclusion

SMARTMAP: Simple Mapping of Address Region Tables for Multi-core Aware Programming

Ron Brightwell, Trammell Hudson, Kevin Pedretti

- Leverages LWK memory management model
- Allows all of the processes on a multi-core processor to access each others' memory directly
  - User-space to user-space
  - No serialization through the OS
  - Access to remote address by flipping a bit
- Each process still has a separate virtual address space
- Allows MPI to minimize memory-to-memory copies on node
  - No copying for non-contiguous MPI datatypes
  - More efficient collective operations
    - Reductions can operate directly on user buffer



Complexity of a Lightweight OS

LWK Code

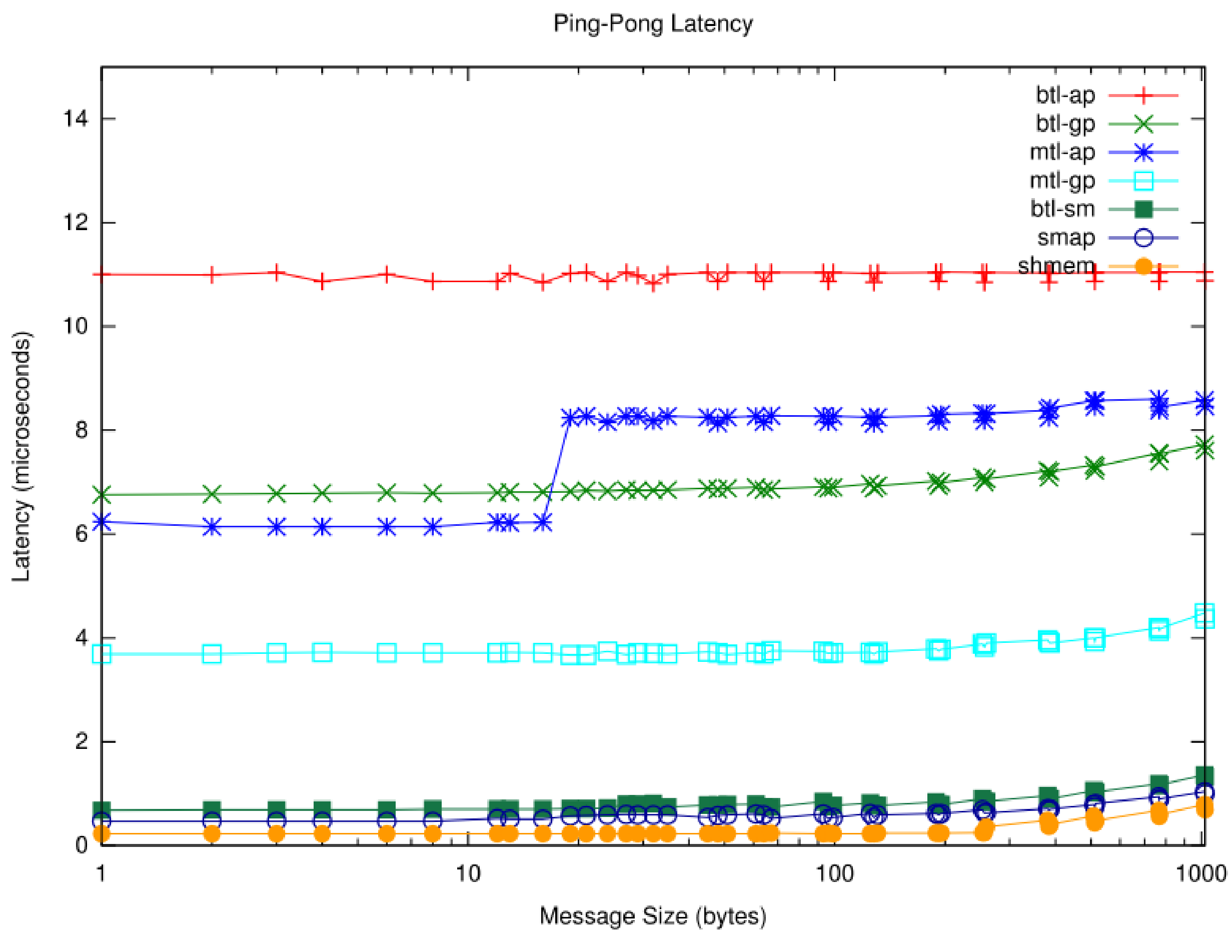
```
static void
initialize_shared_memory( void )
{
	extern page_table_t *pml4_table_cpu[];
	int cpu;
	for ( cpu=0; cpu < MAX_NUM_CPUS; cpu++ )
	{
		page_table_t *pml4 = pml4_table_cpu[ cpu ];
		if ( !pml4 )
			continue;
		pcb_t * kpcb = cur_kpcb_ptr[cpu];
		if ( !kpcb )
			continue;
		page_table_entry_t dirbase = (
			phys_addr( kpcb->kpcb_dirbase )
			| PDE_P
			| PDE_W
			| PDE_U
		);
		int other;
		for ( other=0; other < MAX_NUM_CPUS; other++ )
		{
			page_table_t *other_pml4 = pml4_table_cpu[other];
			if ( !other_pml4 )
				continue;
			other_pml4[ cpu+1 ] = dirbase;
		}
	}
}
```

User Code

```
static inline void *
remote_address(
	unsigned core,
	volatile void * vaddr)
{
	uintptr_t addr = (uintptr_t) vaddr;
	addr |= ((uintptr_t) (core+1)) << 39;
	return (void*) addr;
}
```

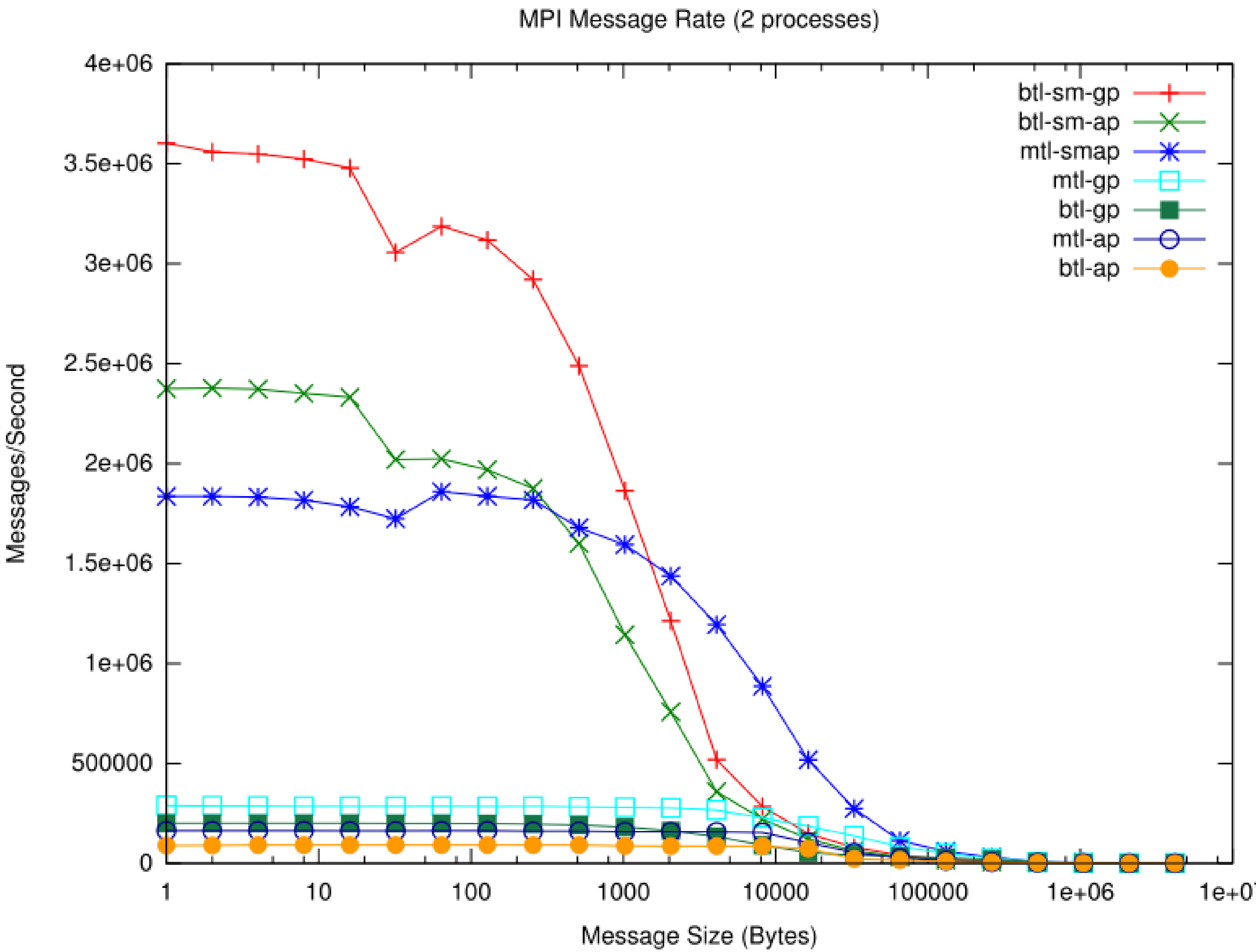

Sandia National Laboratories

PingPong Latency

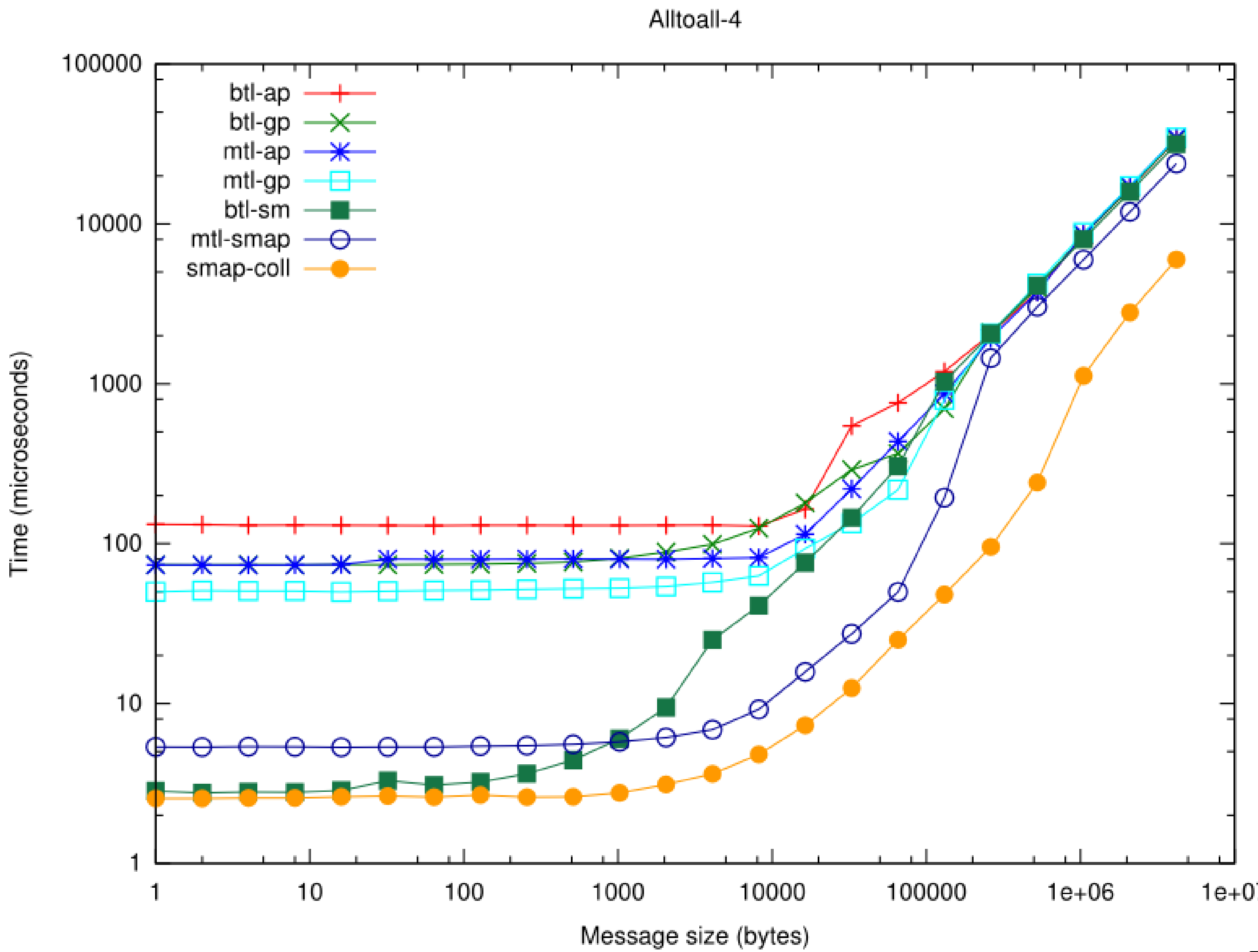


- **2.2 GHz Quad-core AMD Opteron**
- **Catamount N-Way (CNW) 2.0.41**
- **PGI 7.1.4**
- **GNU 3.3.3**
- **Open MPI subversion head**

Sandia National Laboratories



MPI Message Rate (2 processes)
Messages/Second
Message Size (Bytes)
4e+06
3.5e+06
3e+06
2.5e+06
2e+06
1.5e+06
1e+06
500000
0
1
10
100
1000
10000
100000
1e+06
1e+07
btl-sm-gp
btl-sm-ap
mtl-smap
mtl-gp
btl-gp
mtl-ap
btl-ap

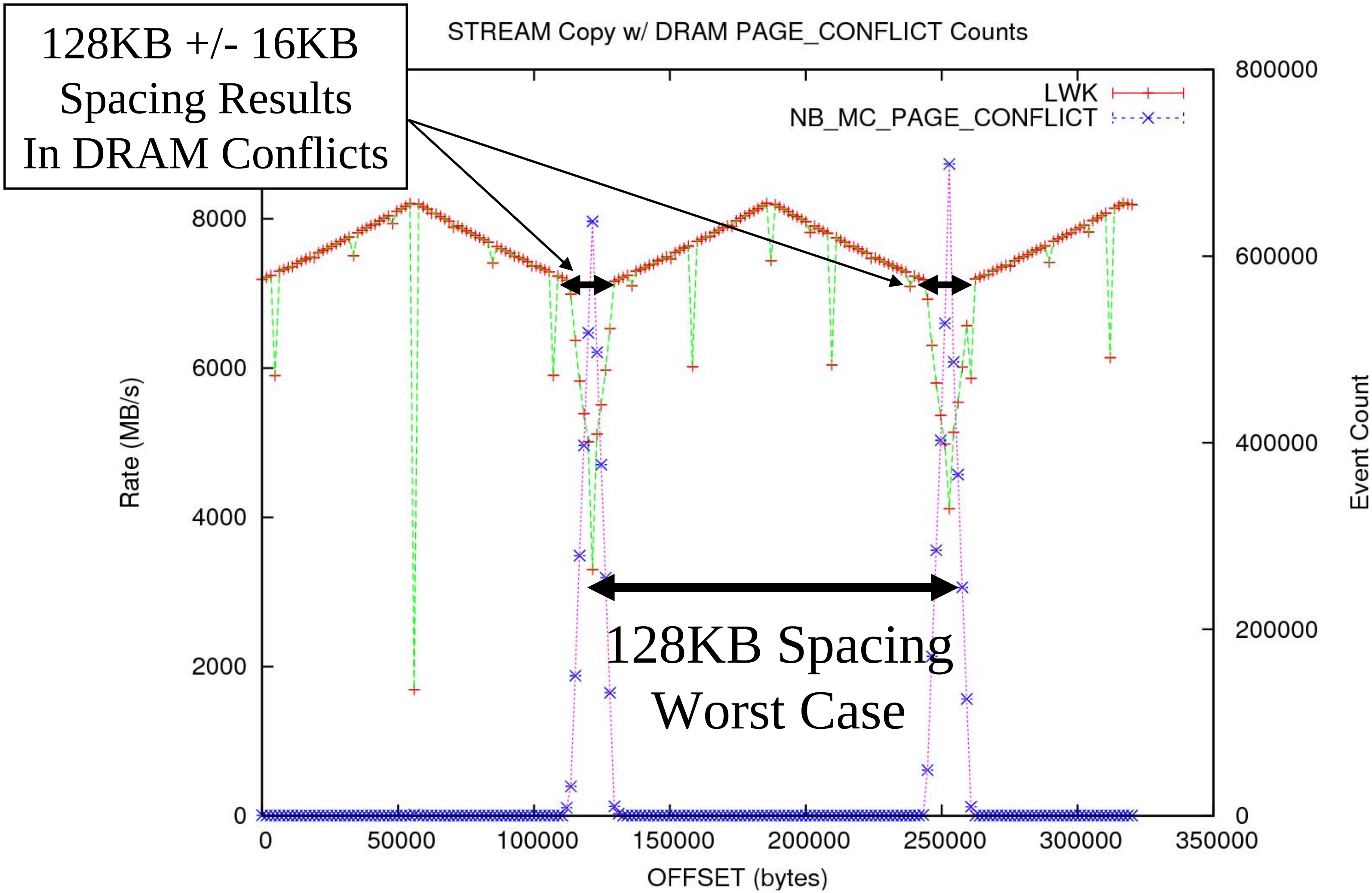


Alltoall-4
Time (microseconds)
Message size (bytes)
btl-ap
btl-gp
mtl-ap
mtl-gp
btl-sm
mtl-smap
smap-coll
100000
10000
1000
100
10
1
10
100
1000
10000
100000
1e+06
1e+07

Future Work

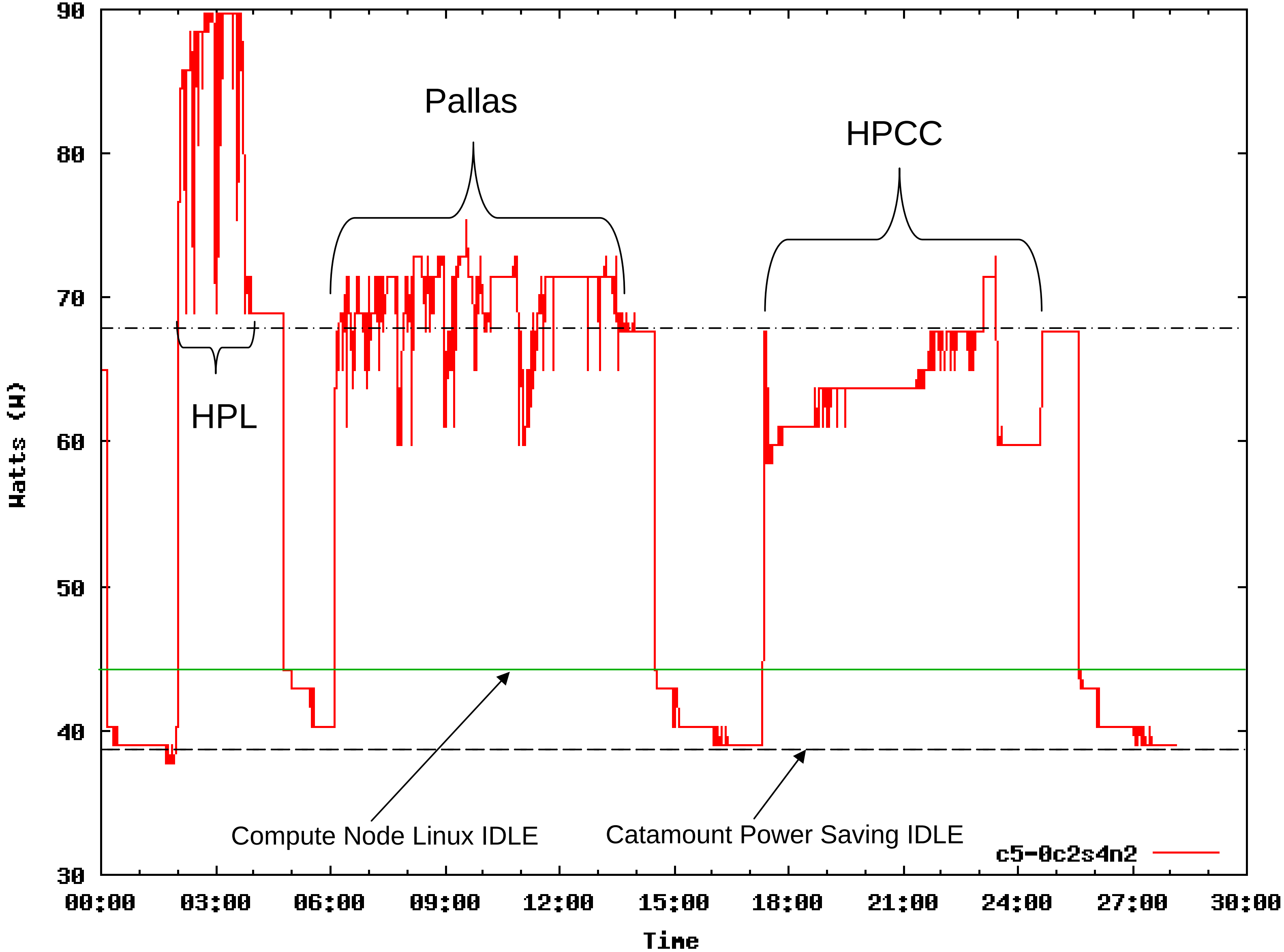
- Lots of MPI work
- Expose node/network topology through MPI communicators
  - MPI_COMM_NODE
  - MPI_COMM_NETWORK
- Explore ways for applications to use directly
  - Compiler (BEC)?
  - Libraries (LibSM)?

Mitigating DRAM Bank Conflicts



Sandia National Laboratories

Application Power Signatures



Conclusion

- Sandia focusing on needs of capability systems
- Quad-core Catamount ready for action
  - Risk mitigation for ORNL Jaguar
  - Will be used for Red Storm upgrade:
    38400 cores, 284 TFLOPS
- Kitten LWK in development
  - Open source
  - Multi-core and hardware virtualization
- Leveraging LWK for system software research

Acknowledgements

- Quad-core Catamount
  - Office of Science and ORNL
  - Sue Kelly, John VanDyke, Courtenay Vaughan, Jim Tomkins
- Kitten LWK
  - Kurt Ferreira, Trammell Hudson, Sue Kelly, Michael Levenhagen, John VanDyke
- SMARTMAP
  - Ron Brightwell, Trammell Hudson
- DRAM Bank Conflicts
  - Kurt Ferreira, Courtenay Vaughan
- Power Signatures
  - Jim Laros

Sandia National Laboratories