FAST-OS: Petascale Single System Image

Presented by

Jeffrey S. Vetter (PI)

Nikhil Bhatia, Collin McCurdy, Phil Roth, Weikuan Yu

Future Technologies Group

Computer Science and Mathematics Division

SC07

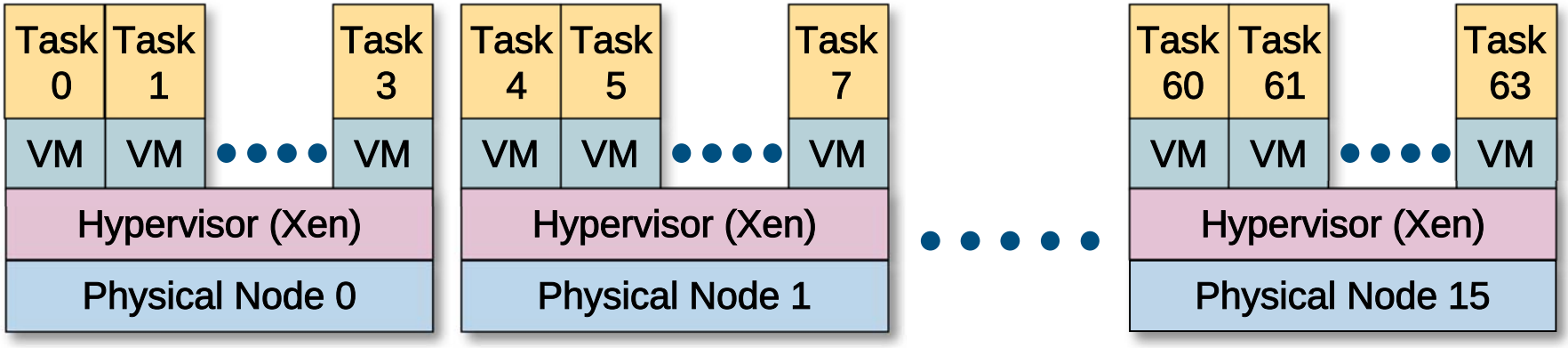
OAK RIDGE National Laboratory

PetaScale single system image

- What?
  - Make a collection of standard hardware look like one big machine, in as many ways as feasible
- Why?
  - Provide a single solution to address all forms of clustering
  - Simultaneously address availability, scalability, manageability, and usability
- Components
  - Virtual cluster using contemporary hypervisor technology
  - Performance and scalability of parallel shared root file system
  - Paging behaviors of applications, i.e., reducing TLB misses
  - Reducing “noise” from operating systems at scale

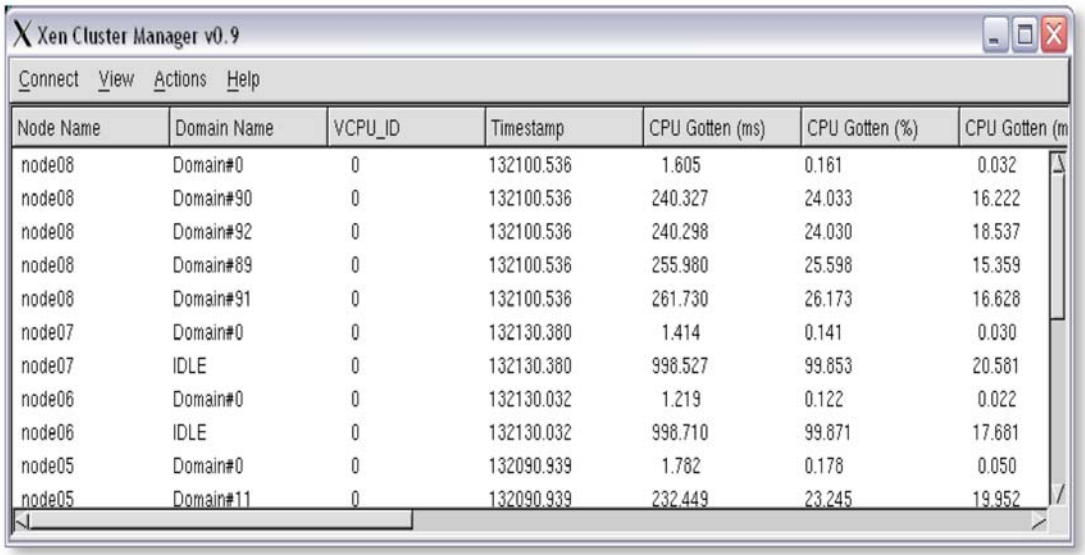
Virtual clusters using hypervisors

- Physical nodes run Hypervisor in the privileged mode.
- Virtual machines run on top of the hypervisor.
- Virtual machines are compute nodes hosting user-level distributed memory (e.g., MPI) tasks.
- Virtualization enables dynamic cluster management via live migration of compute nodes.
- Capabilities for dynamic load balancing and cluster management ensuring high availability.

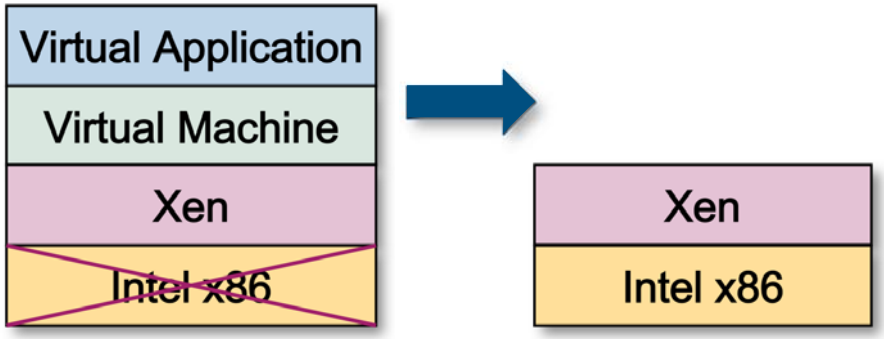


Virtual cluster management

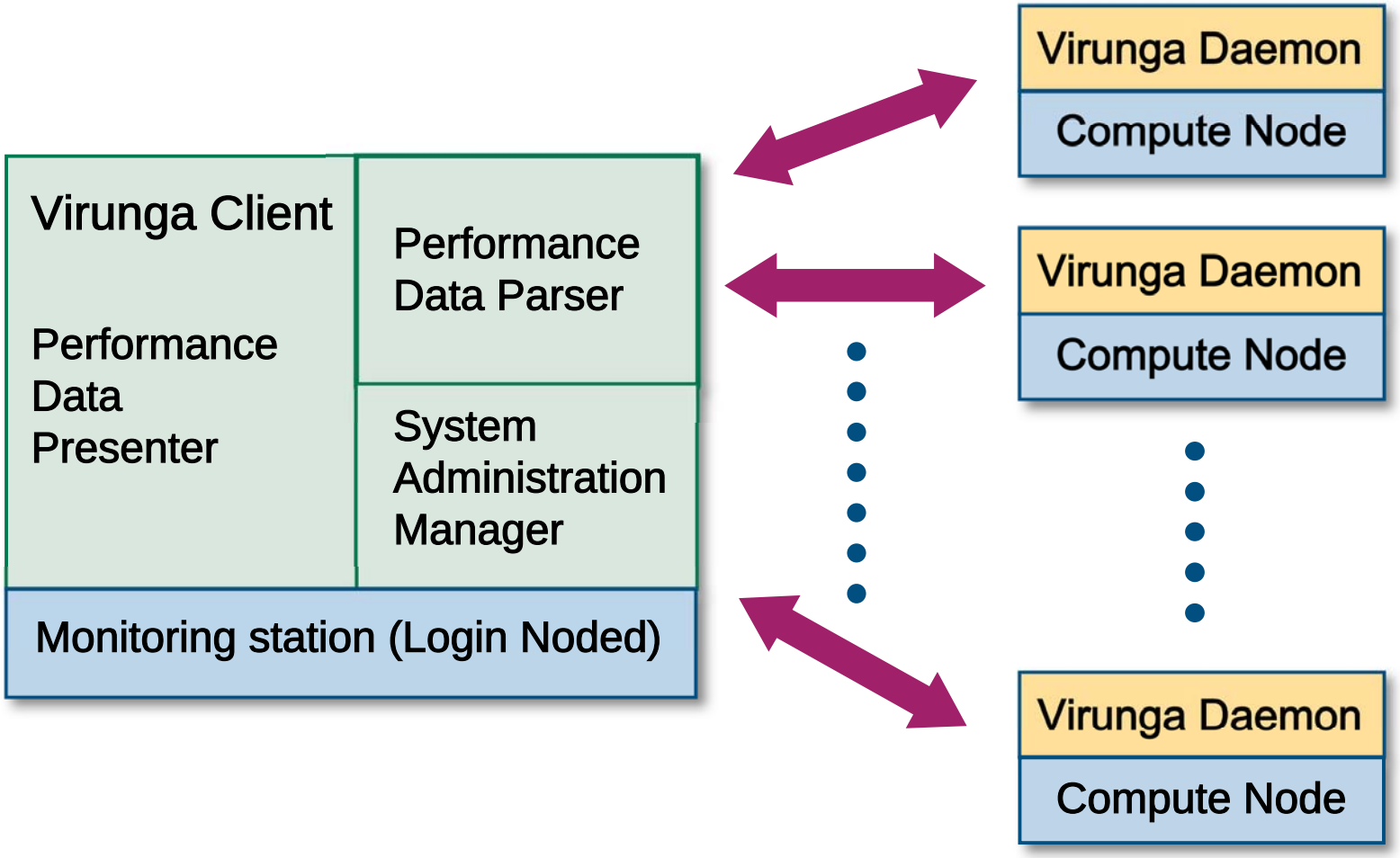
- Single system view of the cluster
- Cluster performance diagnosis
- Dynamic node addition and removal ensuring high availability
- Dynamic load balancing across the entire cluster
- Innovative load balancing schemes based on distributed algorithms



Node Name	Domain Name	VCPU_ID	Timestamp	CPU Gotten (ms)	CPU Gotten (%)	CPU Gotten (m
node08	Domain#0	0	132100.536	1.605	0.161	0.032
node08	Domain#90	0	132100.536	240.327	24.033	16.222
node08	Domain#92	0	132100.536	240.298	24.030	18.537
node08	Domain#89	0	132100.536	255.980	25.598	15.359
node08	Domain#91	0	132100.536	261.730	26.173	16.628
node07	Domain#0	0	132130.380	1.414	0.141	0.030
node07	IDLE	0	132130.380	998.527	99.853	20.581
node06	Domain#0	0	132130.032	1.219	0.122	0.022
node06	IDLE	0	132130.032	998.710	99.871	17.681
node05	Domain#0	0	132090.939	1.782	0.178	0.050
node05	Domain#11	0	132090.939	232.449	23.245	19.952



Virunga—virtual cluster manager

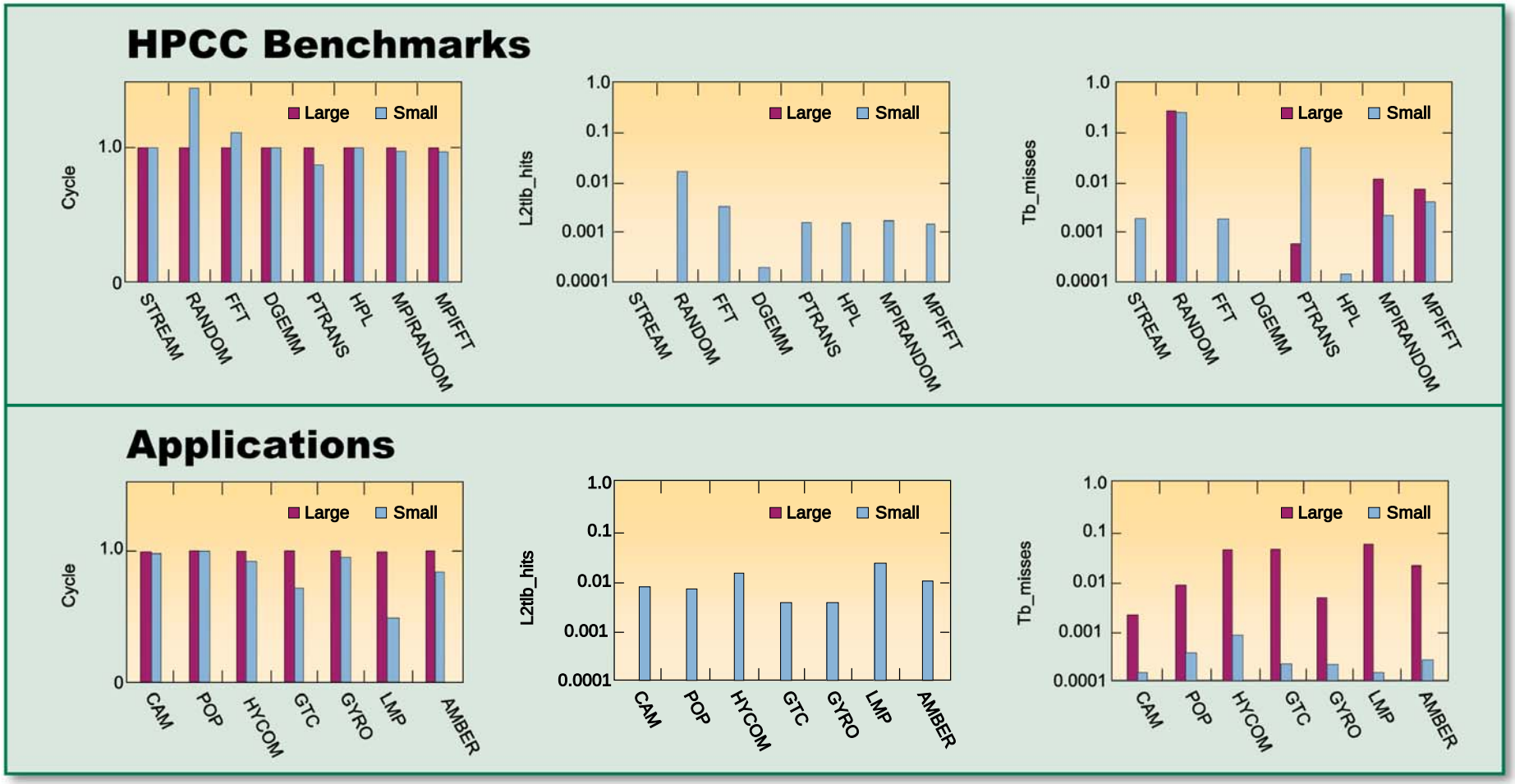


Paging behavior of DOE applications

Goal: Understand behavior of applications in large-scale systems composed of commodity processors

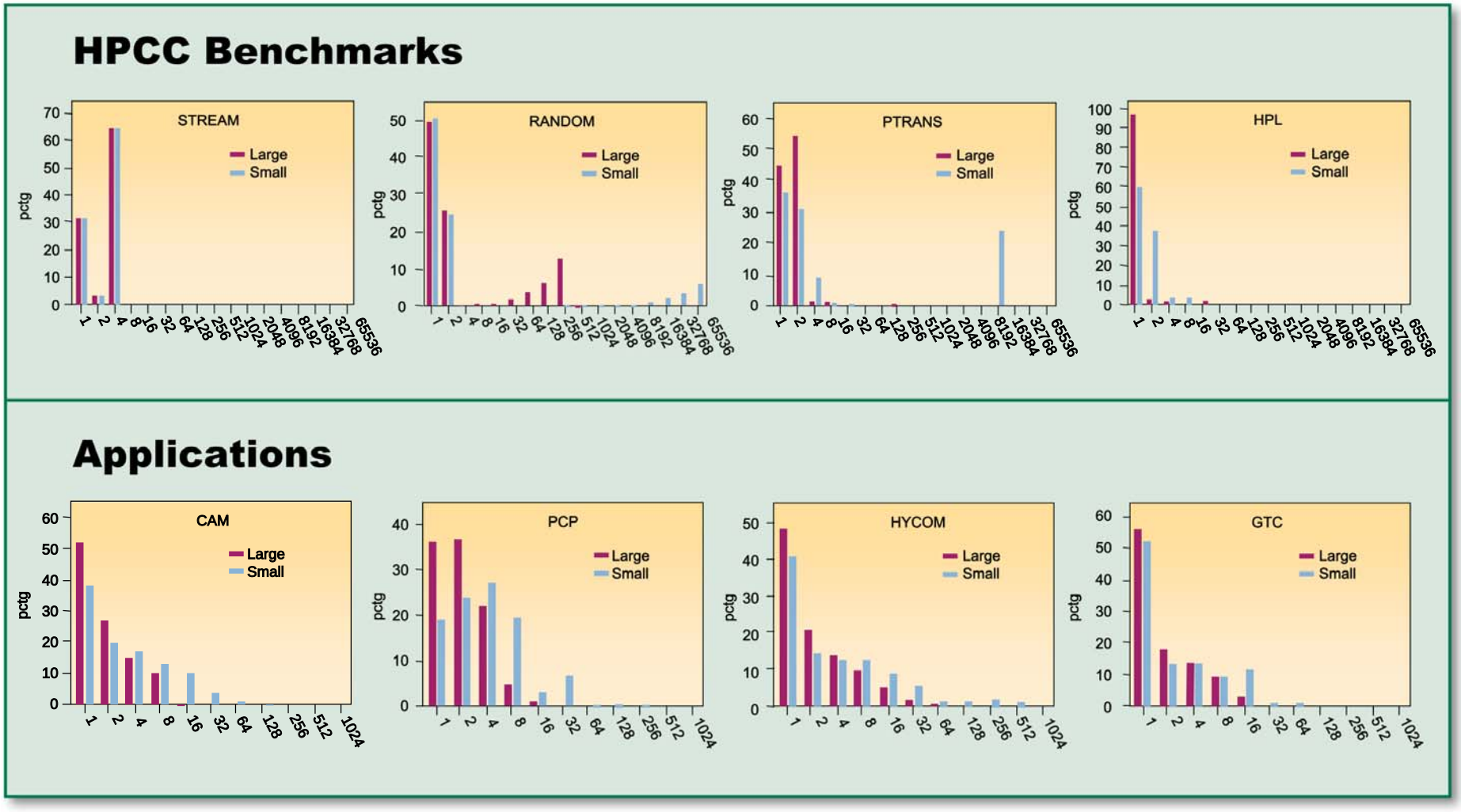
1. With performance counters and simulation:
   - Show that the HPCC benchmarks, meant to characterize the memory behavior of HPC applications, do not exhibit the same behavior in the presence of paging hardware as scientific applications of interest to the Office of Science
   - Offer insight into why that is the case
2. Use memory system simulation to determine whether large page performance will improve with the next generation of Opteron processors

Experimental results (from performance counters)



- Performance: trends for large vs small nearly opposite
- TLBs: significantly different miss rates

Reuse distances (simulated)



Patterns are clearly significantly different…

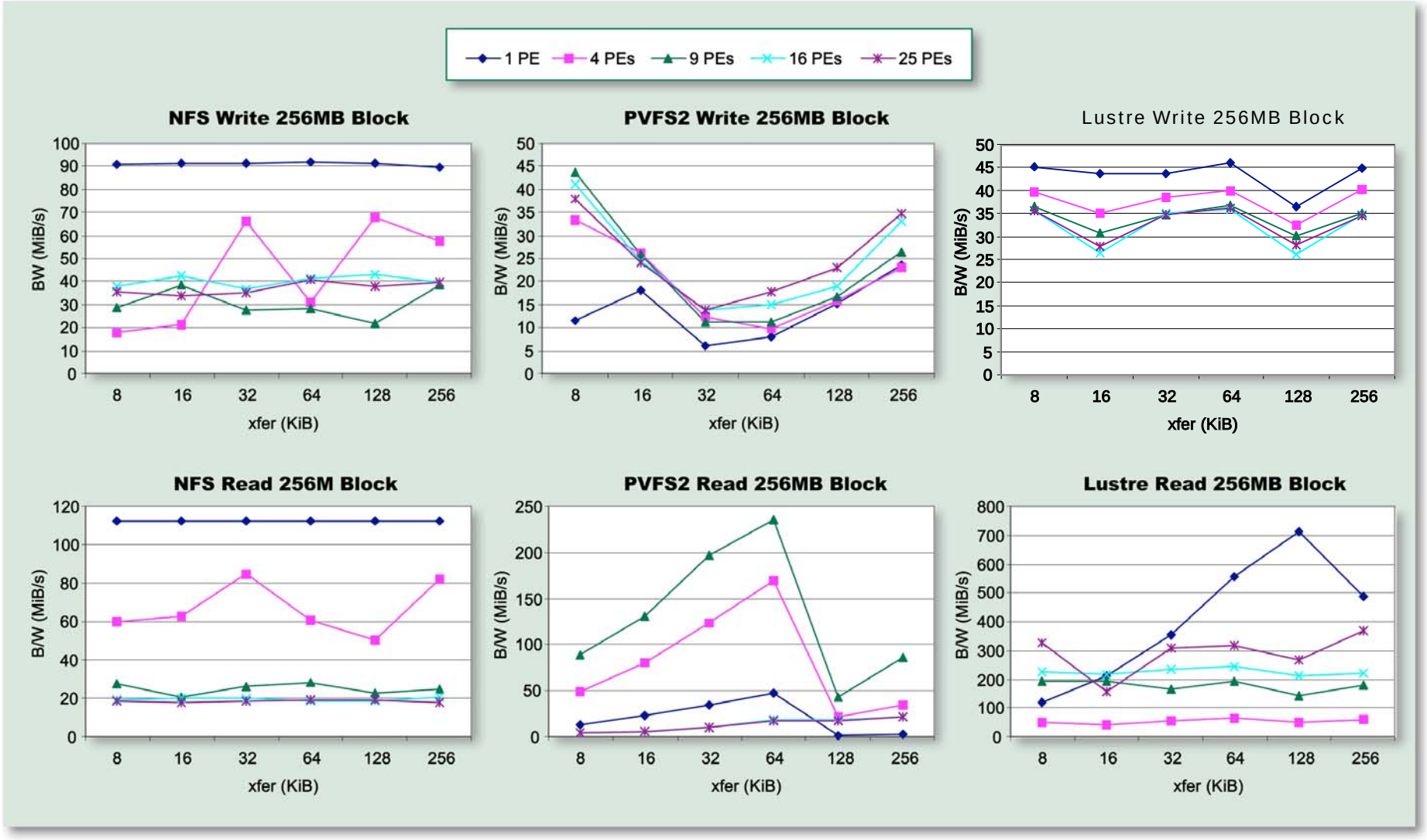
Paging conclusions

- HPCC benchmarks are *not* representative of paging behavior of typical DOE applications.
  - Applications access many more arrays concurrently.
- Simulation results (not shown) indicate the following:
  - New paging hardware in next-generation Opteron processors *will* improve large page performance.
    - Performance near that with paging turned off.
  - *However*, simulations also indicate that the mere presence of a TLB is likely degrading performance, whether paging hardware is on or off.
- More research into implications of paging, and of commodity processors in general, on performance of scientific applications is required.

Parallel root file system

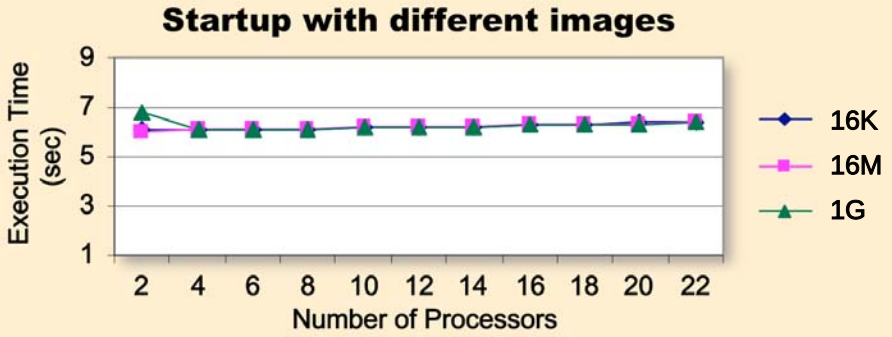
- Goals of study
  - Use a parallel file system for implementation of shared root environment
  - Evaluate performance of parallel root file system
  - Evaluate the benefits of high-speed interconnects
  - Understand root I/O access pattern and potential scaling limits
- Current status
  - RootFS implemented using NFS, PVFS-2, Lustre, and GFS
  - RootFS distributed using ramdisk via etherboot
    - Modified mkinitrd program locally
    - Modified to init scripts to mount root at boot time
  - Evaluation with parallel benchmarks (IOR, b_eff_io, NPB I/O)
  - Evaluation with emulated loads of I/O accesses for RootFS
  - Evaluation of high-speed interconnects for Lustre-based RootFS

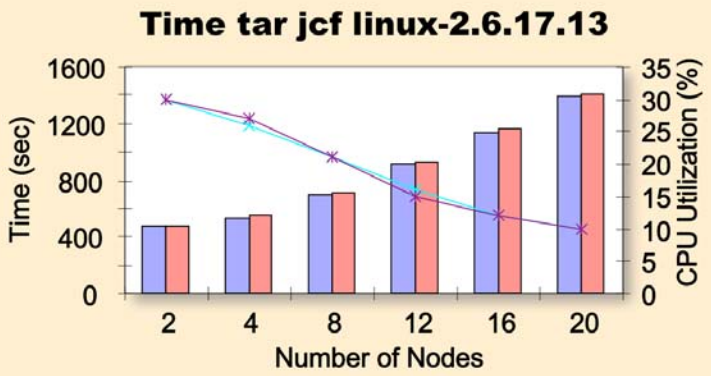
Example results: Parallel benchmarks IOR read/write throughput

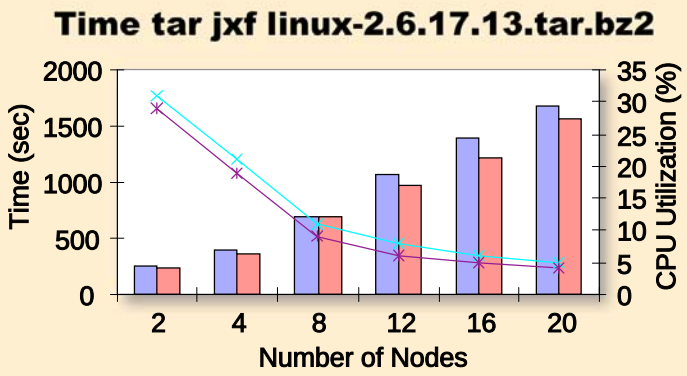


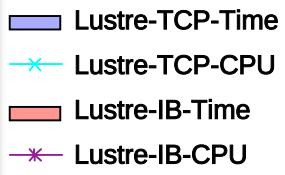
OAK RIDGE National Laboratory

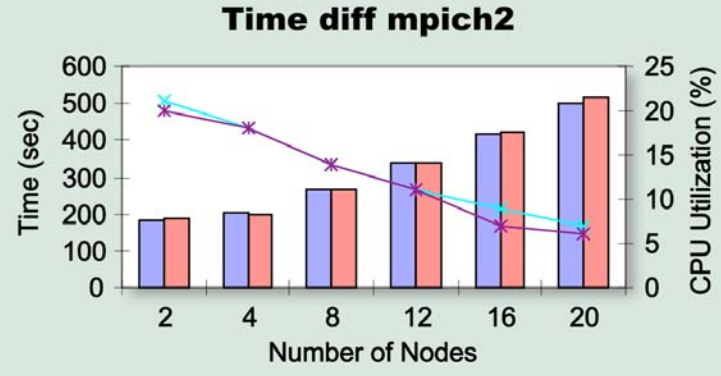
Performance with synthetic I/O accesses

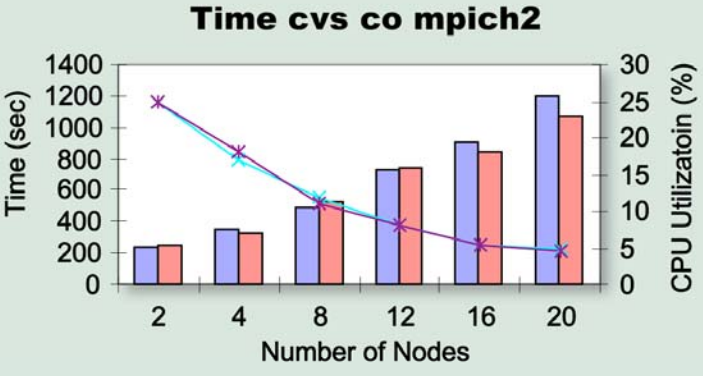


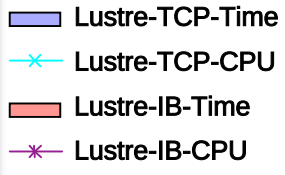












Contacts

Jeffrey S. Vetter

Principle Investigator
Future Technologies Group
Computer Science and Mathematics Division
(865) 356-1649
vetter@ornl.gov

Nikhil Bhatia

(865) 241-1535
bhatia@ornl.gov

Phil Roth

(865) 241-1543
rothpc@ornl.gov

Collin McCurdy

(865) 241-6433
cmccurdy@ornl.gov

Weikuan Yu

(865) 574-7990
wyu@ornl.gov