



TX-2500

An Interactive, On-Demand Rapid-Prototyping HPC System

Albert Reuther, William Arcand, Tim Currie, Andrew Funk,
Jeremy Kepner, Matthew Hubbell, Andrew McCabe, and Peter Michaleas

HPEC 2007

September 18-20, 2007

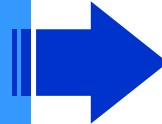


This work is sponsored by the Department of the Air Force under Air Force contract FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.



Outline

- **Introduction**



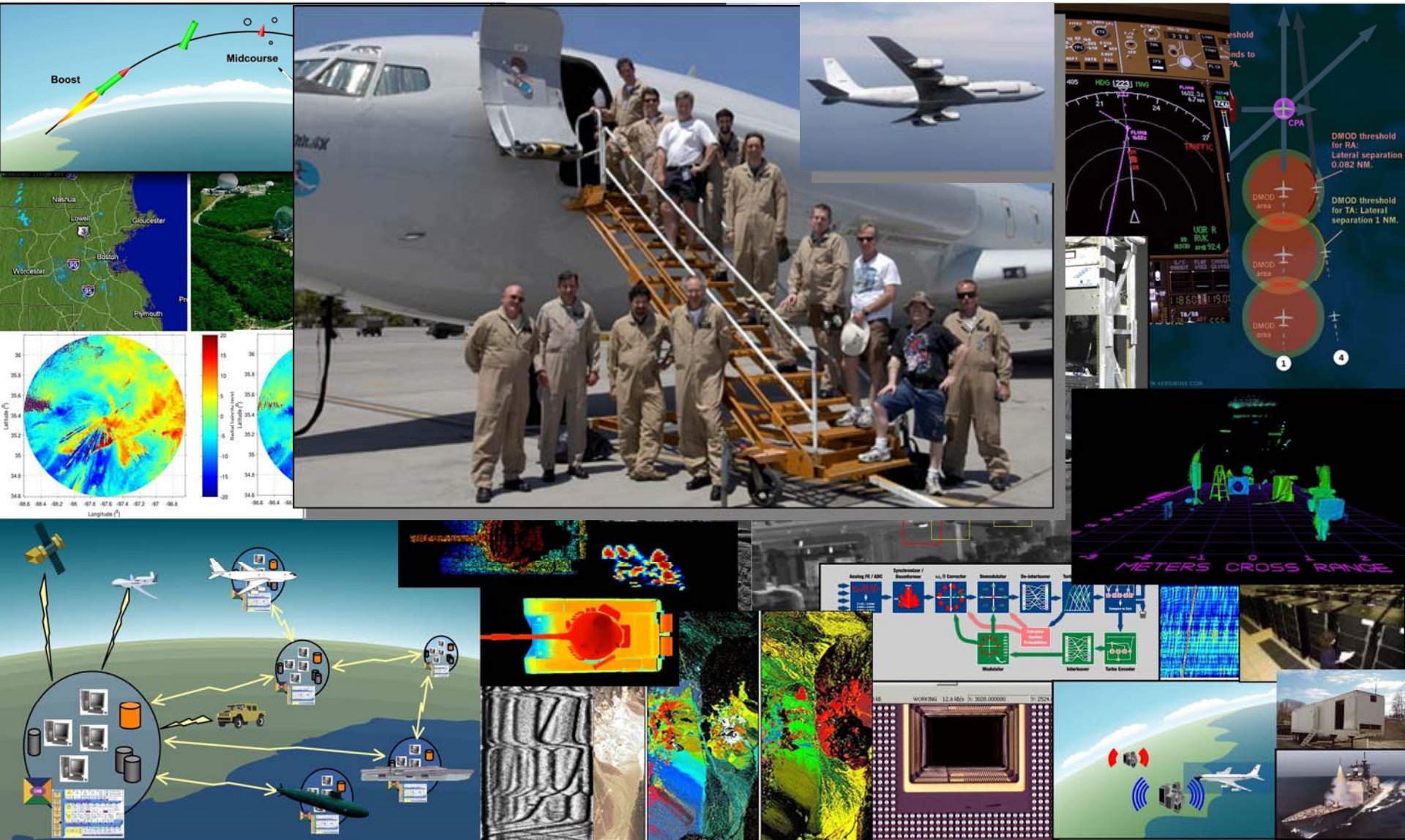
- *Motivation*
- *HPEC Software Design Workflow*

- Interactive, On-Demand

- High Performance

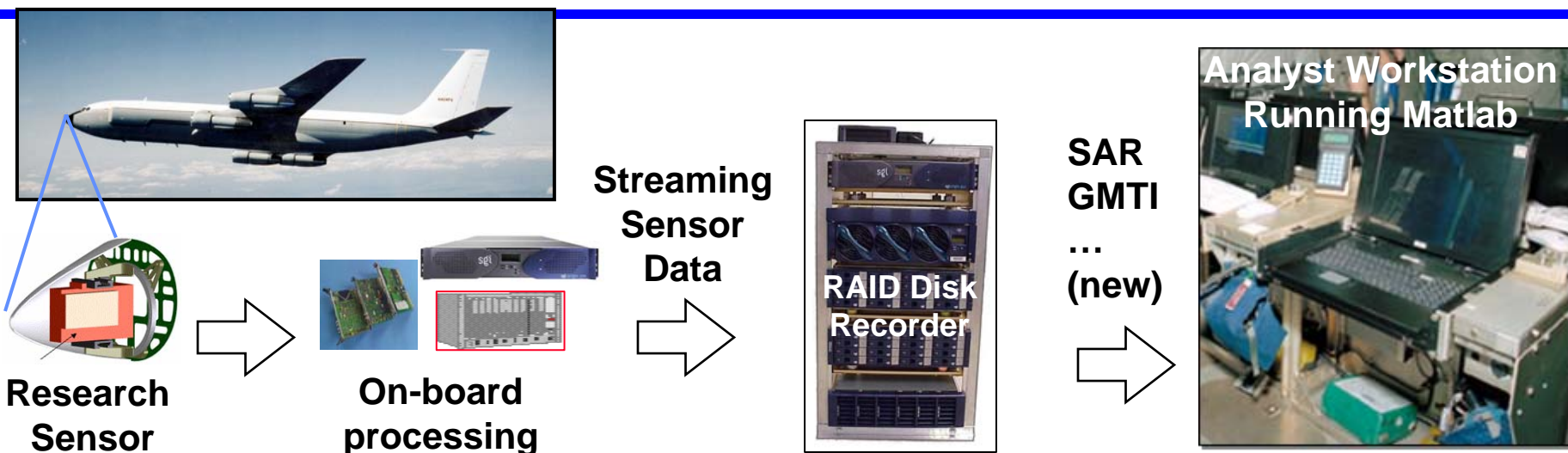
- Summary

LLGrid Applications

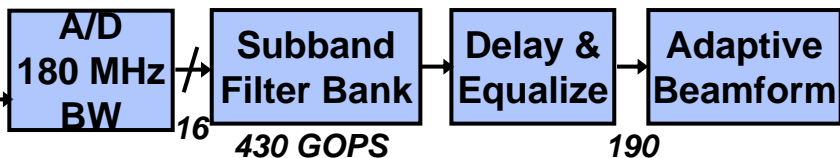




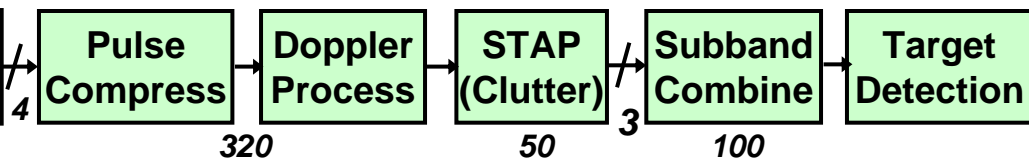
Example App: Prototype GMTI & SAR Signal Processing



Real-time front-end processing



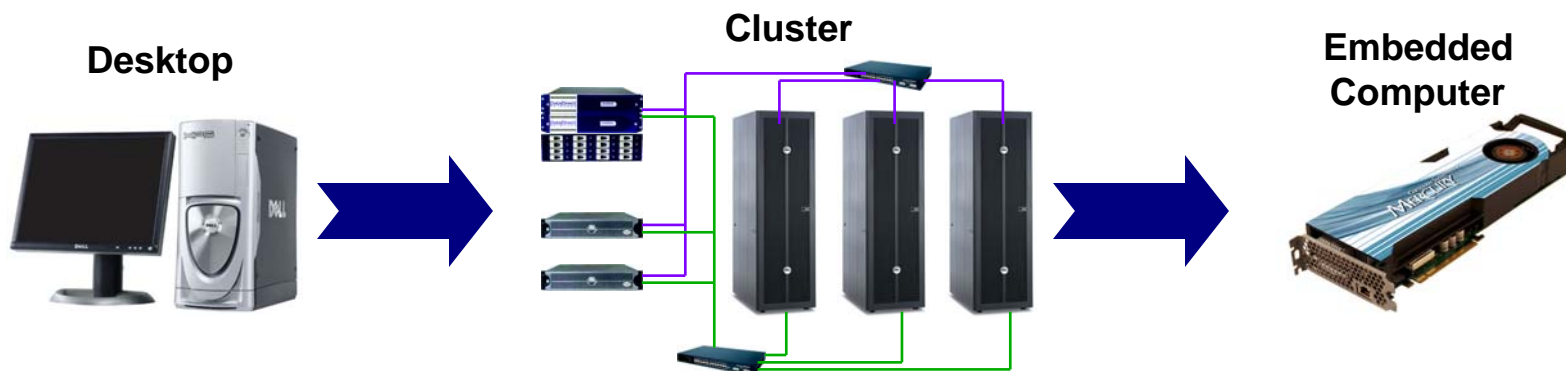
Non-real-time GMTI processing



- Airborne research sensor data collected
- Research analysts develop signal processing algorithms in MATLAB® using collected sensor data
- Individual runs can last hours or days on single workstation



HPEC Software Design Workflow



Algorithm Development

Develop serial code → Parallel code → Embedded code

Requirements

- Low barrier-to-entry
- Interactive
- Immediate execution
- High performance

Verification

Parallel code ↔ Embedded code

Data Analysis

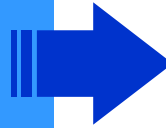
Parallel code ↔ Embedded code



Outline

- Introduction

- **Interactive, On-Demand**



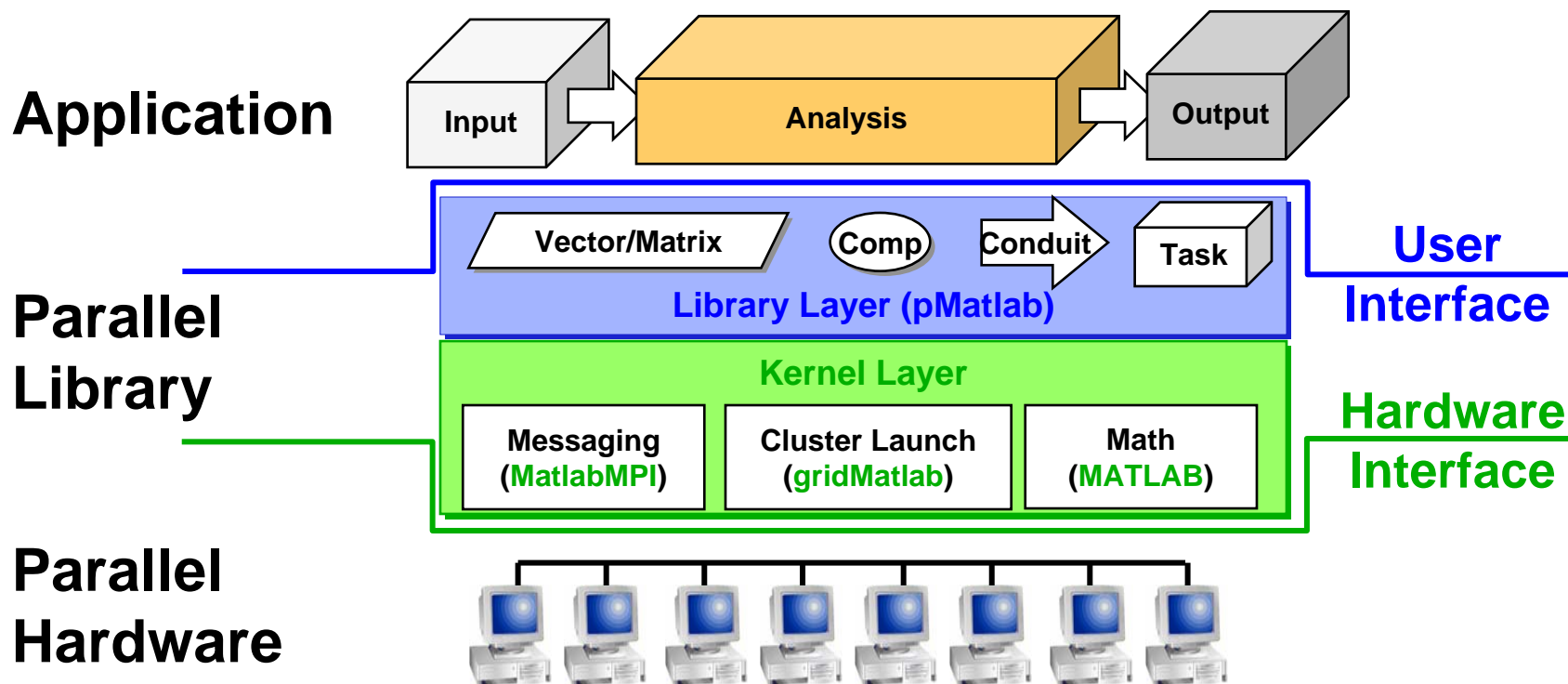
- *Technology*
- *Results*

- High Performance

- Summary



Parallel Matlab (pMatlab)



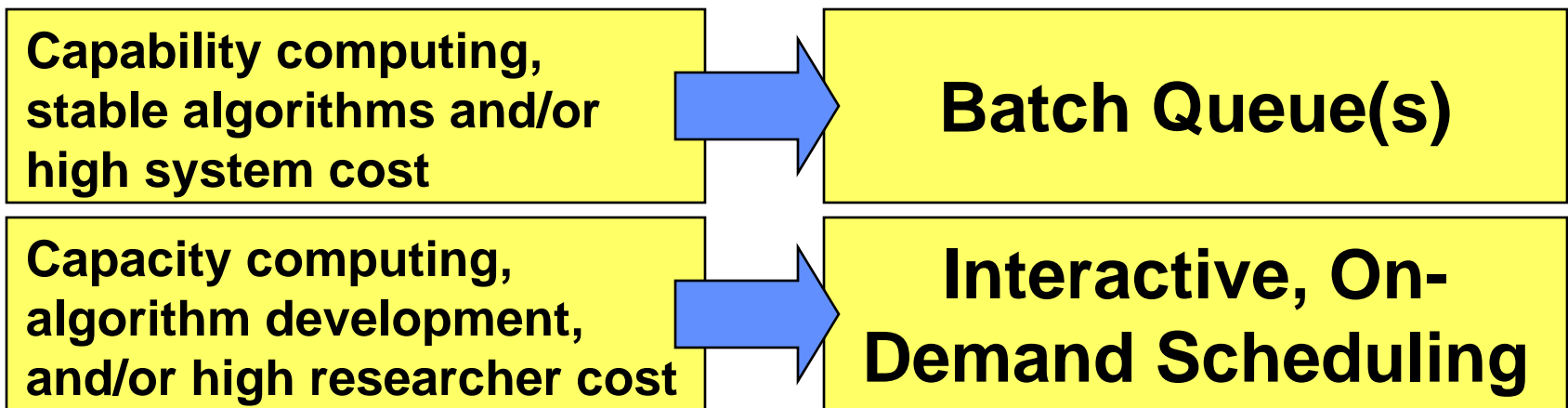
Layered Architecture for parallel computing

- Kernel layer does single-node math & parallel messaging
- Library layer provides a parallel data and computation toolbox to Matlab users



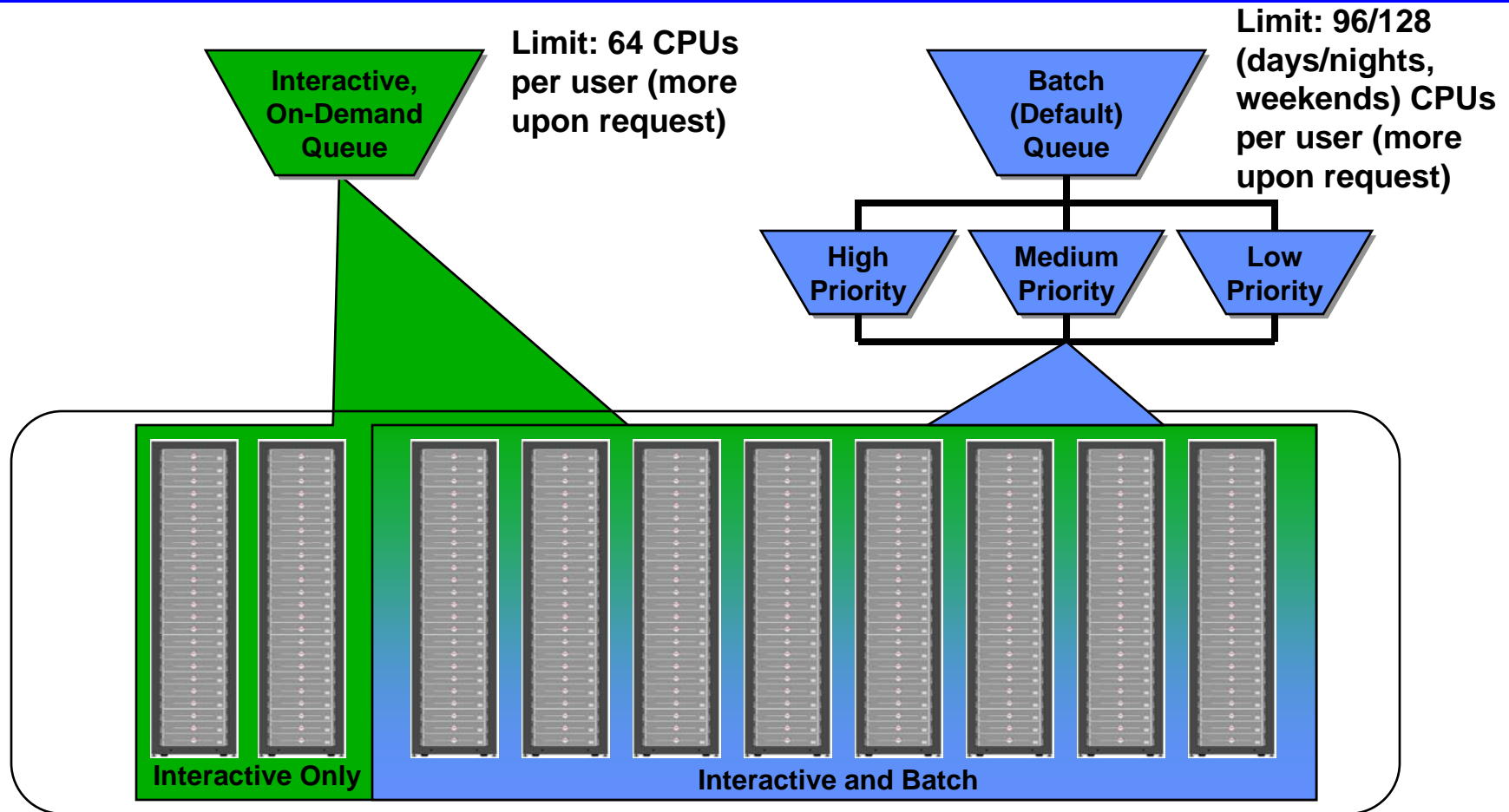
Batch vs. Interactive, On-Demand Scheduling Tradeoffs

- **What is the goal of the system?**
 - Capability computing
 - Capacity computing
- **What types of algorithms and code will system run?**
 - Stable algorithms
 - Algorithm development
- **What's more expensive?**
 - The system and it's maintenance
 - The researchers' time





LLGrid User Queues

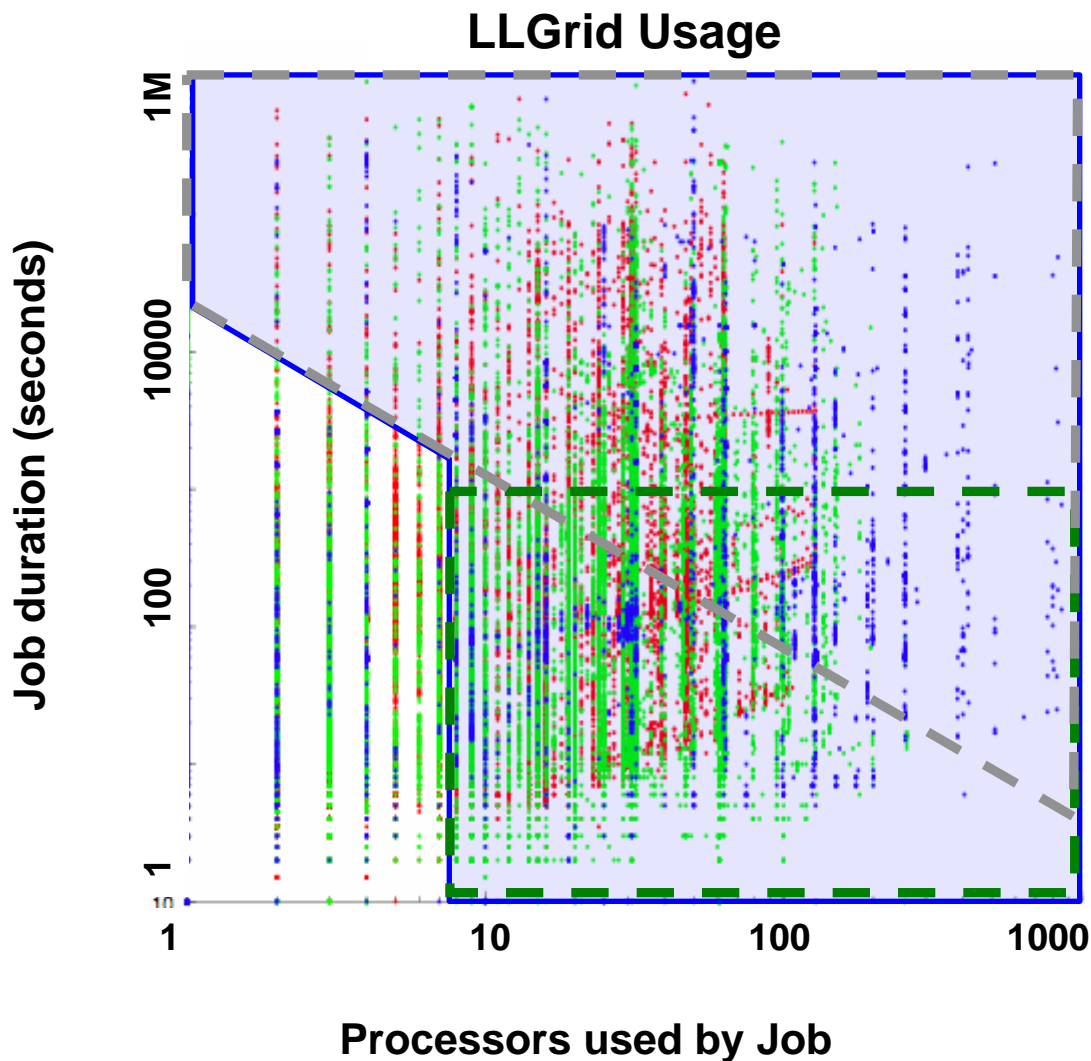


- Not using scheduler's interactive features (`lsrun`, `lsgun`, `bsub -I`)
- Certain CPUs for interactive, on-demand jobs only
- CPU allotments will change when upgrading to larger system



LLGrid Usage

December 2003 – August 2007



Statistics

- 280 + 864 CPUs
- 226 Users
- 234,658 Jobs
- 130,160 CPU Days

>8 CPU hours - Infeasible on Desktop

>8 CPUs - Requires On-Demand Parallel Computing

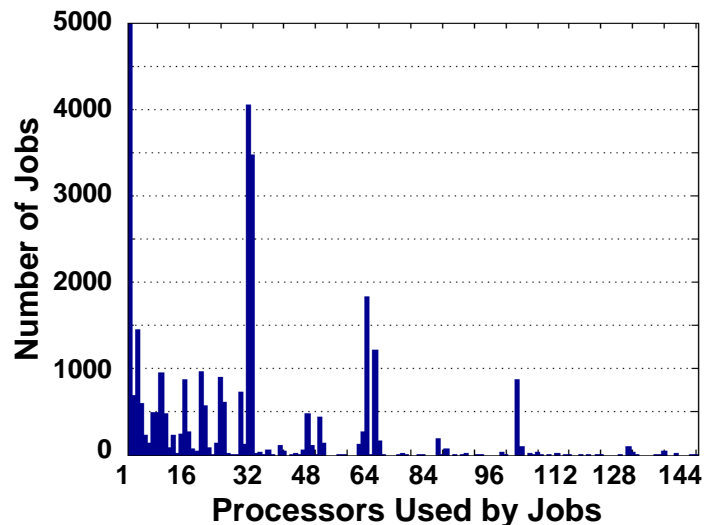
Jobs Legend:

- alphaGrid
- betaGrid
- TX-2500

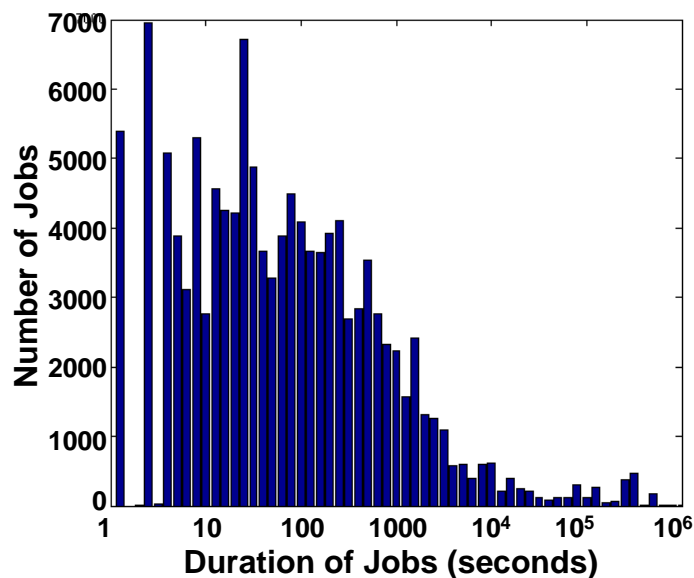


Usage Statistics

December-03 – August-07



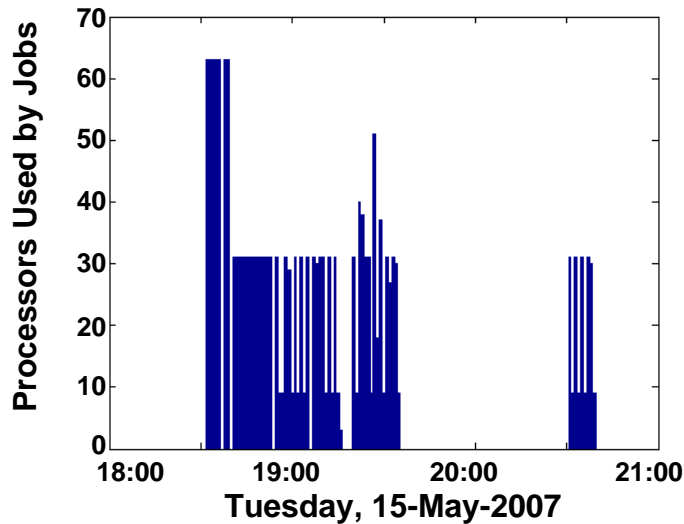
Total jobs run	234,658
Median CPUs per job	11
Mean CPUs per job	20
Maximum CPUs per job	856
Total CPU time	130,160d 15h
Median job duration	35s
Mean job duration	40m 41s
Max job duration	18d 7h 6m



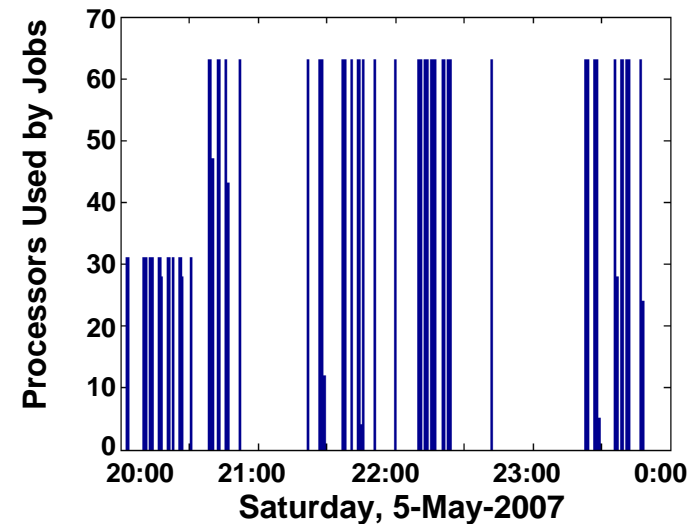
- Most jobs use less than 32 CPUs
- Many jobs with 2, 8, 24, 30, 32, 64, and 100 CPUs
- Very low median job duration
- Modest mean job duration
- Some jobs still take hours on 16 or 32 CPUs



Individuals' Usage Examples



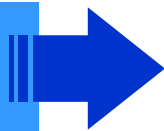
- **Developing non-linear equalization ASIC**
- **Post-run processing from overnight run**
- **Debug runs during day**
- **Prepare for long overnight runs**



- **Simulating laser propagation through atmosphere**
- **Simulation results direct subsequent algorithm development and parameters**
- **Many engineering iterations during course of day**

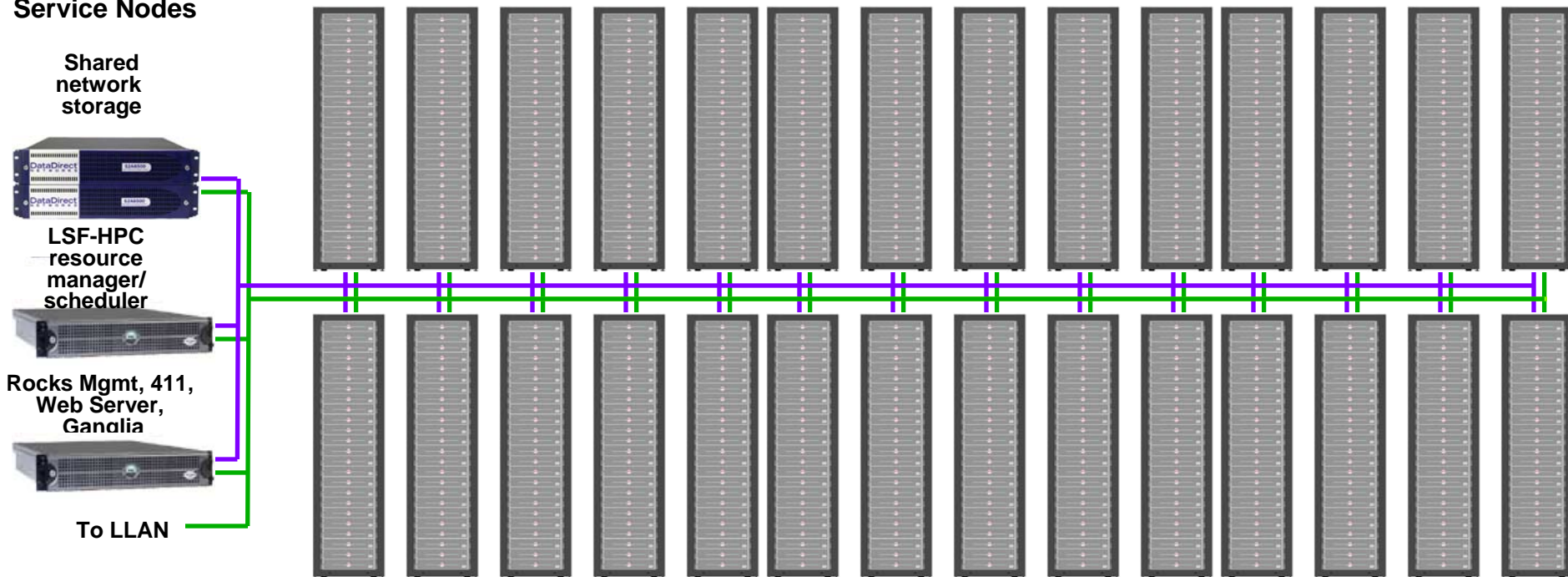


Outline

- Introduction
- Interactive, On-Demand
- **High Performance** 
 - *Technology*
 - *Results*
- Summary

TX-2500

Service Nodes



432 **DELL** PowerEdge 2850



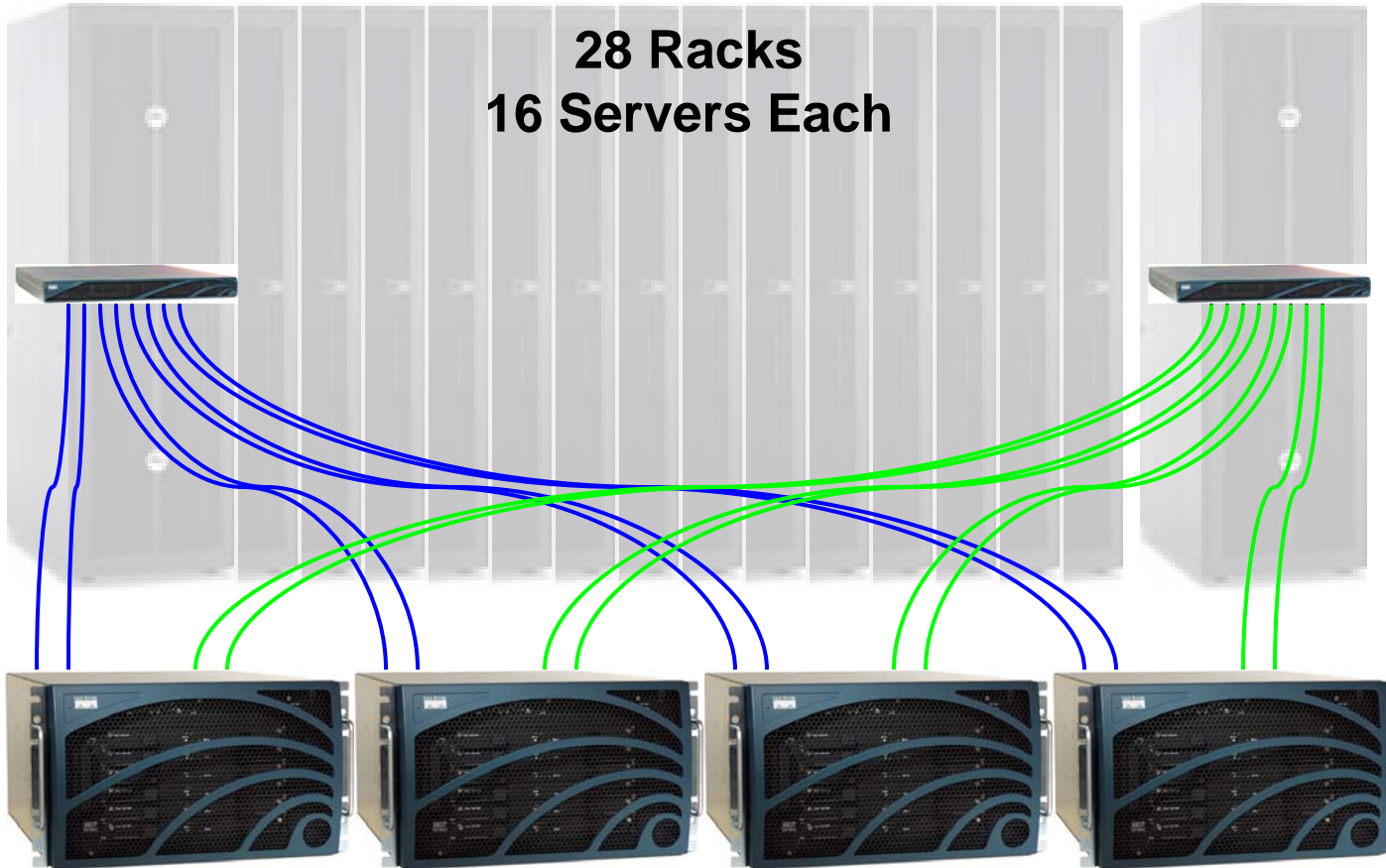
Dual 3.2 GHz EM64-T Xeon
(Irwindale CPUs on Lindenhurst
Chipset)
8 GB RAM memory
Two Gig-E Intel interfaces
InfiniBand interface
Six 300-GB disk drives

- 432+3 Nodes
- 864+6 CPUs
- 3.4 TB RAM
- **0.78 PB of Disk**
- 28 Racks



InfiniBand Topology

**28 Racks
16 Servers Each**



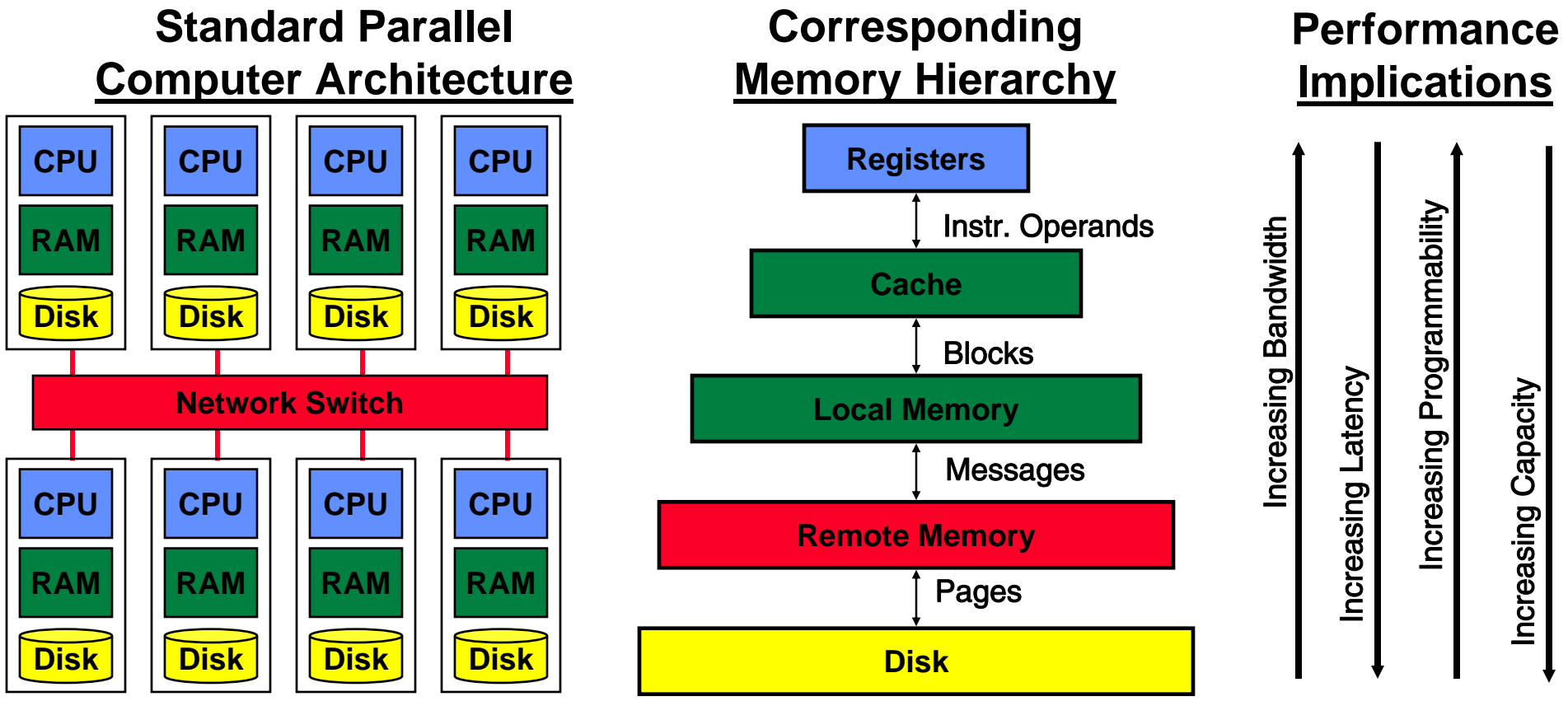
**28 Cisco SFS 7000P
Rack Switches
(24 InfiniBand 10-
Gbps 4X ports each)**

**Four Cisco SFS 7008P
Core Switches (96
InfiniBand 10-Gbps 4X
ports each)**

- **Two uplinks from each rack switch to each core switch**
- **Provides redundancy and minimizes contention**



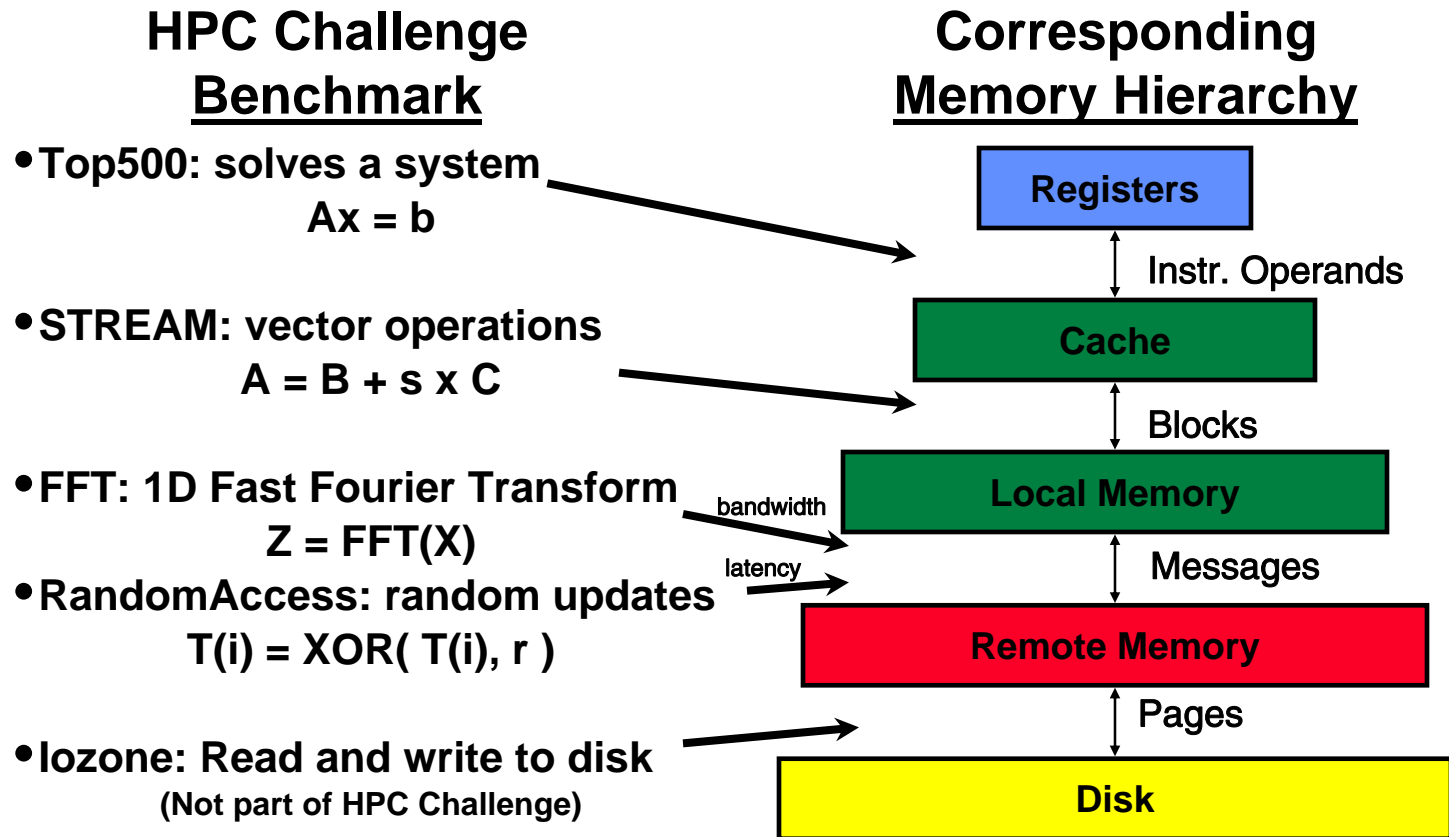
Parallel Computing Architecture Issues



- Standard architecture produces a “steep” multi-layered memory hierarchy
 - Programmer must manage this hierarchy to get good performance
- Need to measure each level to determine system performance

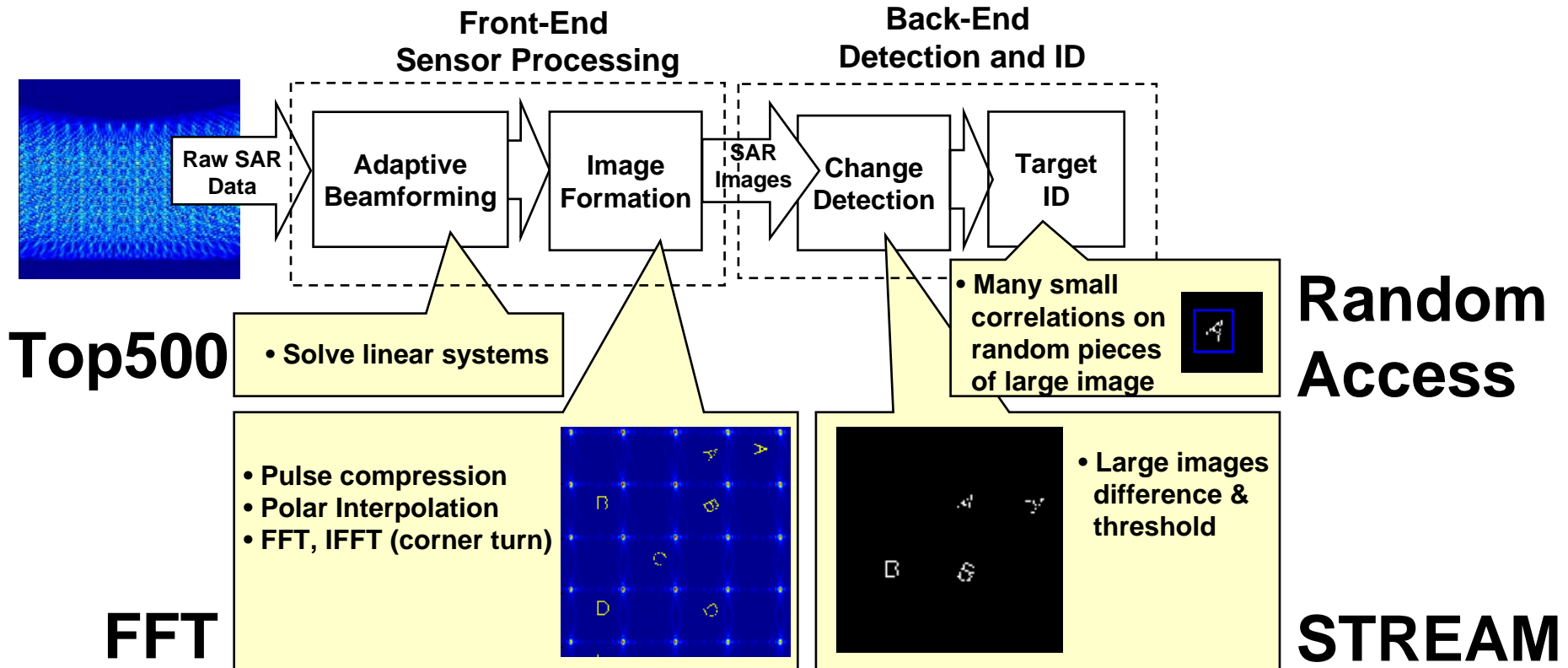


HPC Challenge Benchmarks



- HPC Challenge with lozone measures this hierarchy
- Benchmarks performance of architecture

Example SAR Application

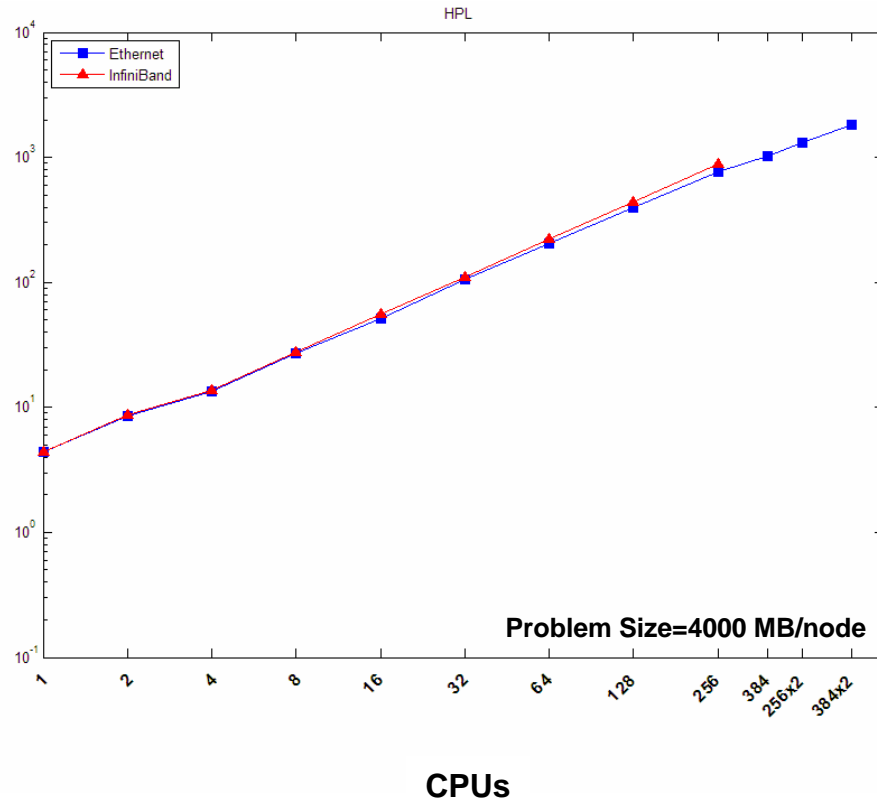


- HPC Challenge benchmarks are similar to pieces of real apps
- Real applications are an average of many different operations
- How do we correlate HPC Challenge with application performance?

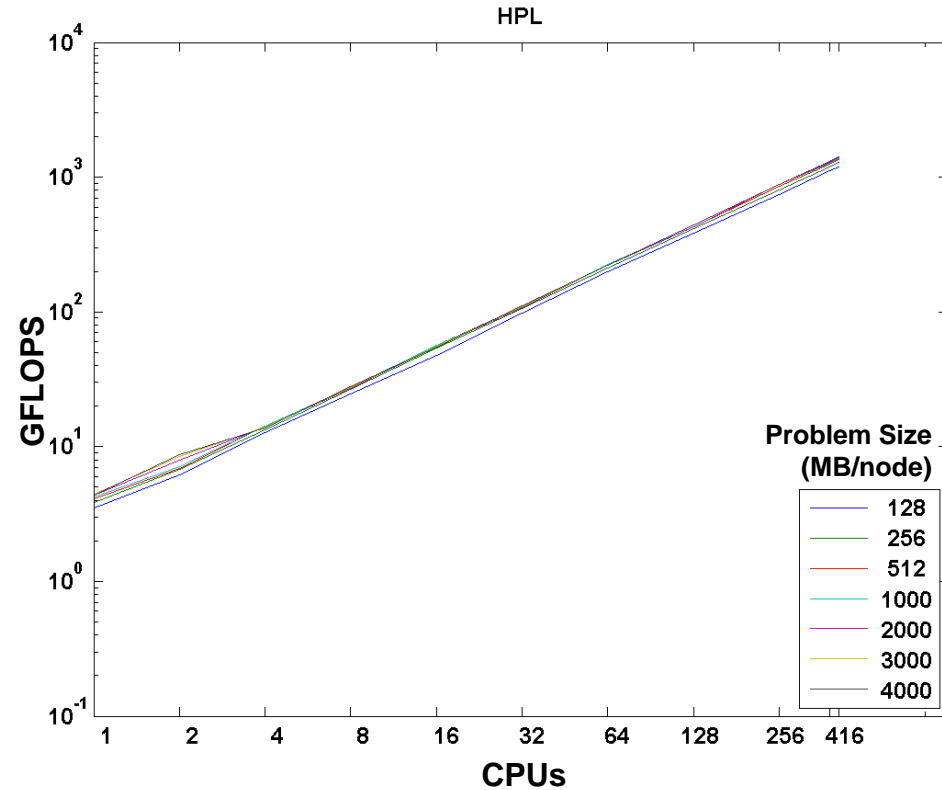


Flops: Top500

Ethernet vs. InfiniBand



Problem Size Comparison - InfiniBand

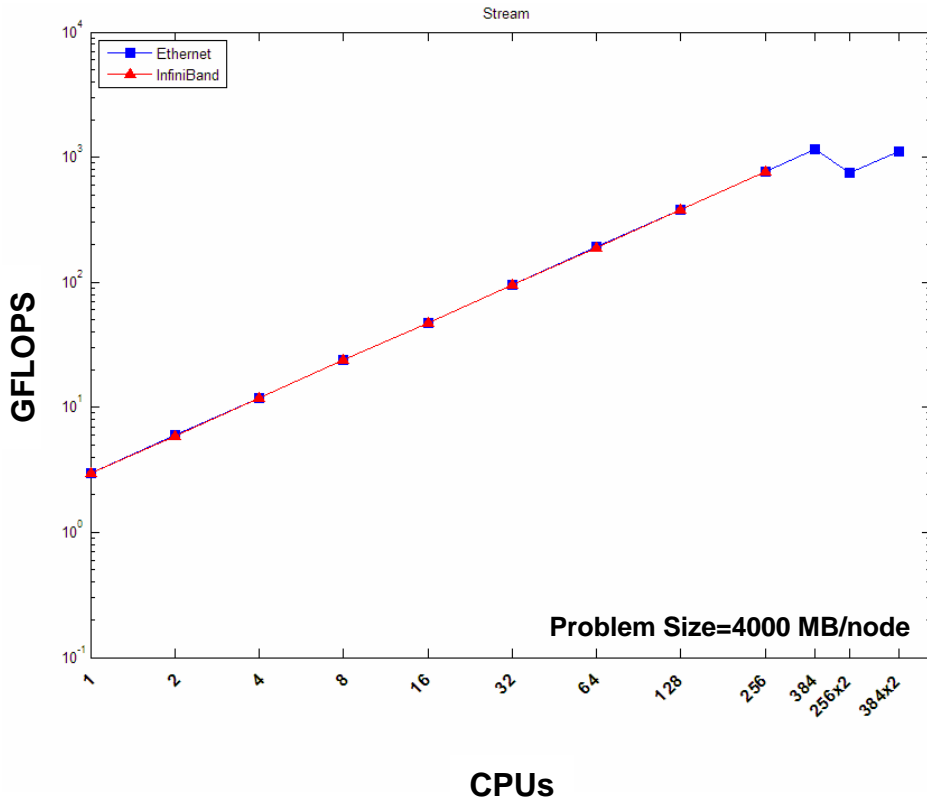


- Top500 measures register-to-CPU comms (1.42 TFlops peak)
- Top500 does not stress the network subsystem

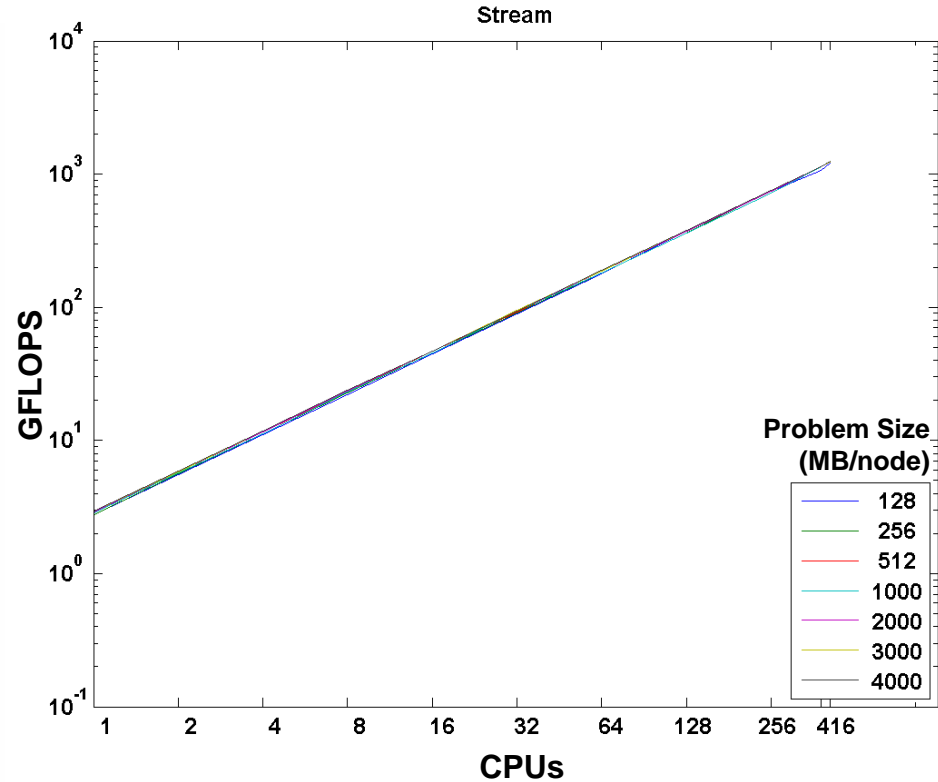


Memory Bandwidth: STREAM

Ethernet vs. InfiniBand



Problem Size Comparison - InfiniBand



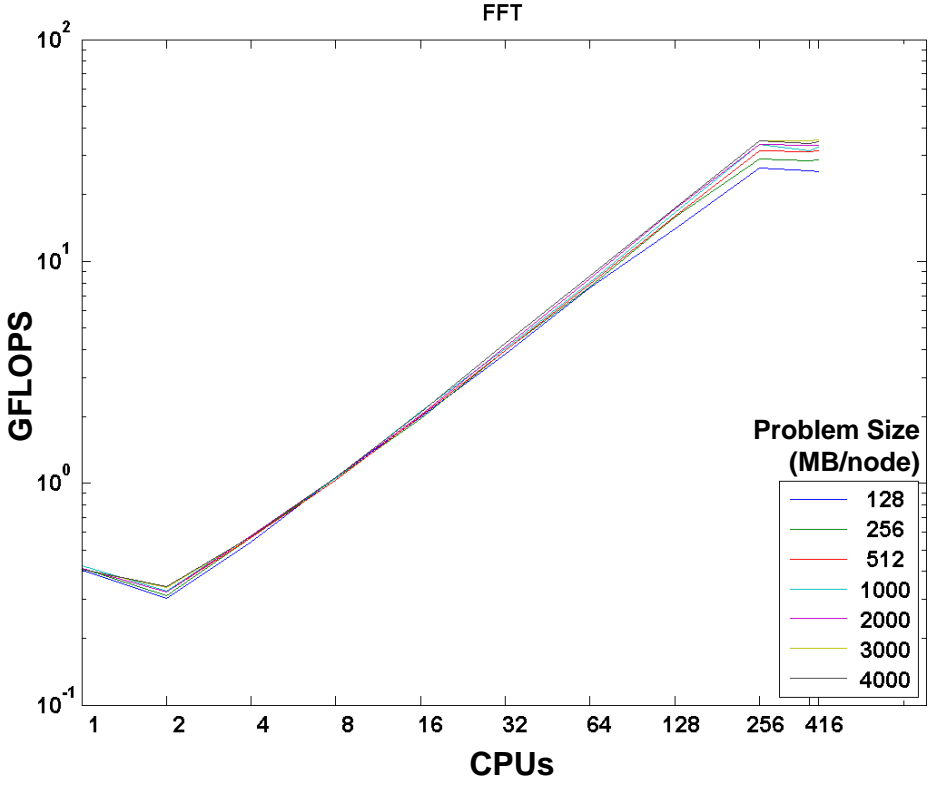
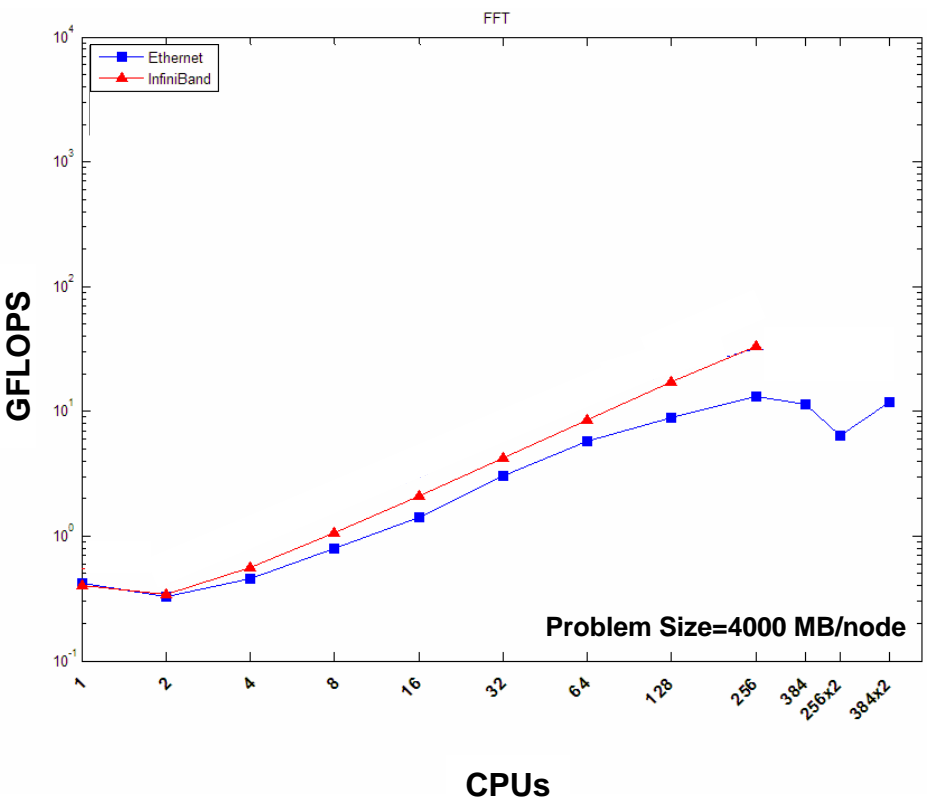
- Memory to CPU bandwidth tested (~2.7 GB/s per node)
- Unaffected by memory footprint of data set



Network Bandwidth: FFT

Ethernet vs. InfiniBand

Problem Size Comparison - InfiniBand

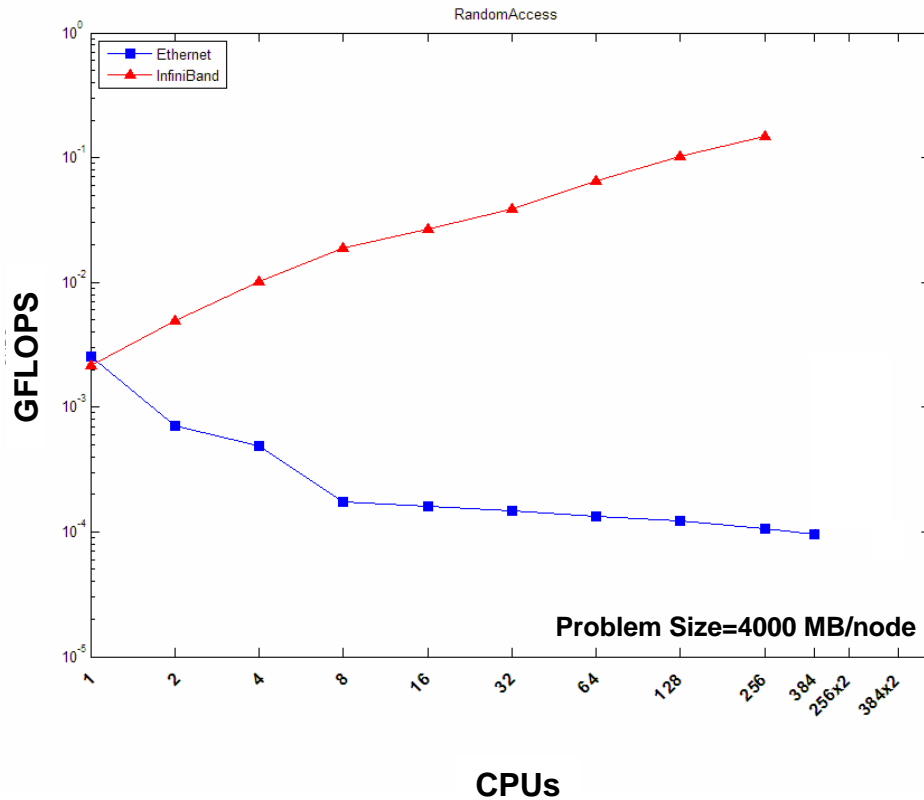


- Infiniband makes gains due to higher bandwidth
- Ethernet plateaus around 256 CPUs

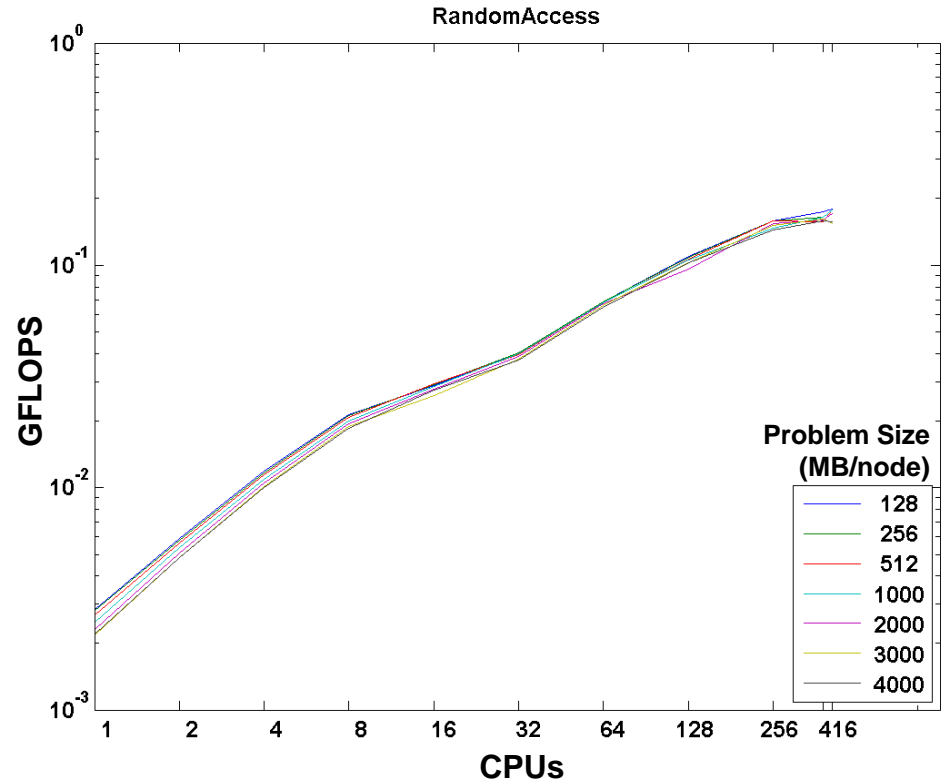


Network Latency: RandomAccess

Ethernet vs. InfiniBand



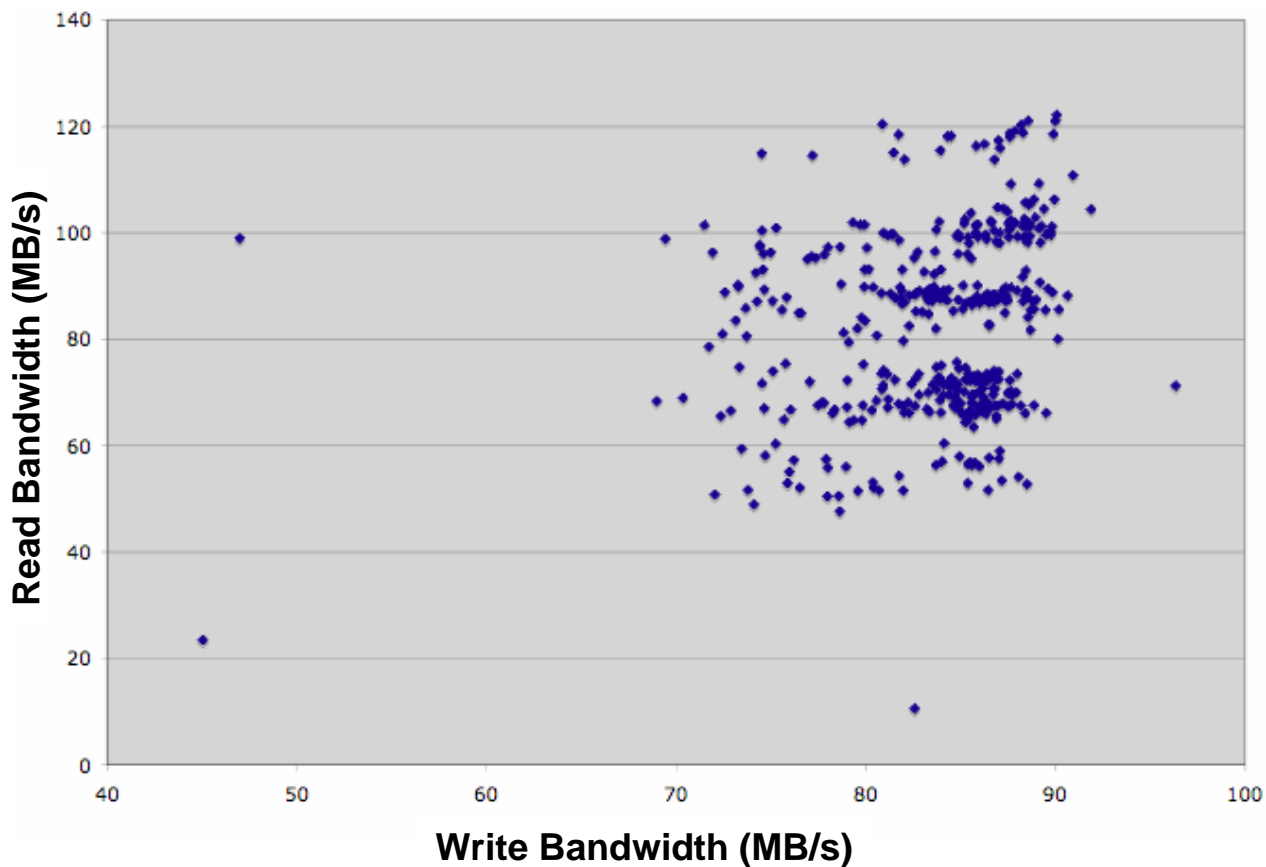
Problem Size Comparison - InfiniBand



- Low latency of InfiniBand drives GUPS performance
- Strong mapping to embedded interprocessor comms



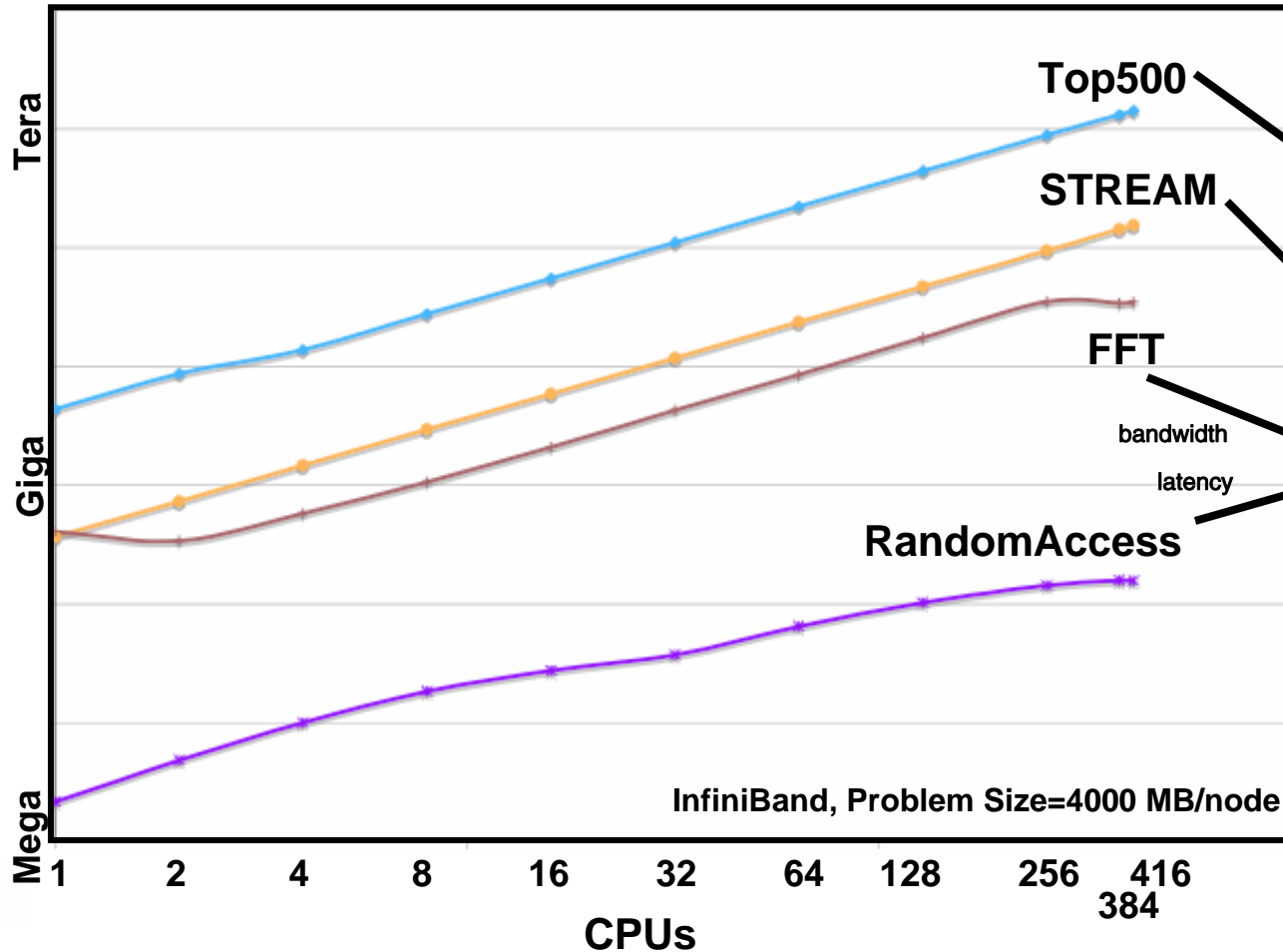
iozone Results



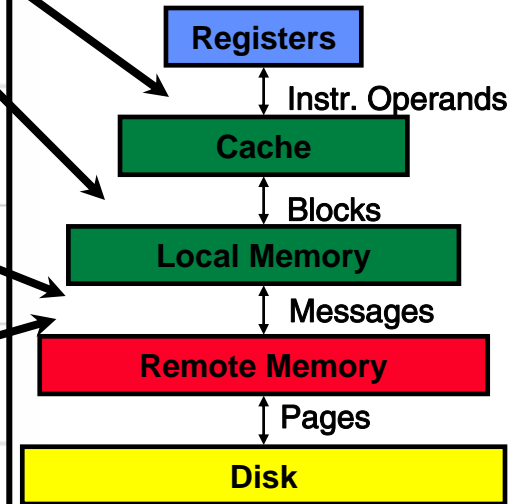
- **98% of the RAID arrays exceeded 50 MB/s throughput**
- **Comparable performance to embedded storage**

HPC Challenge Comparison

Effective Bandwidth (words/second)



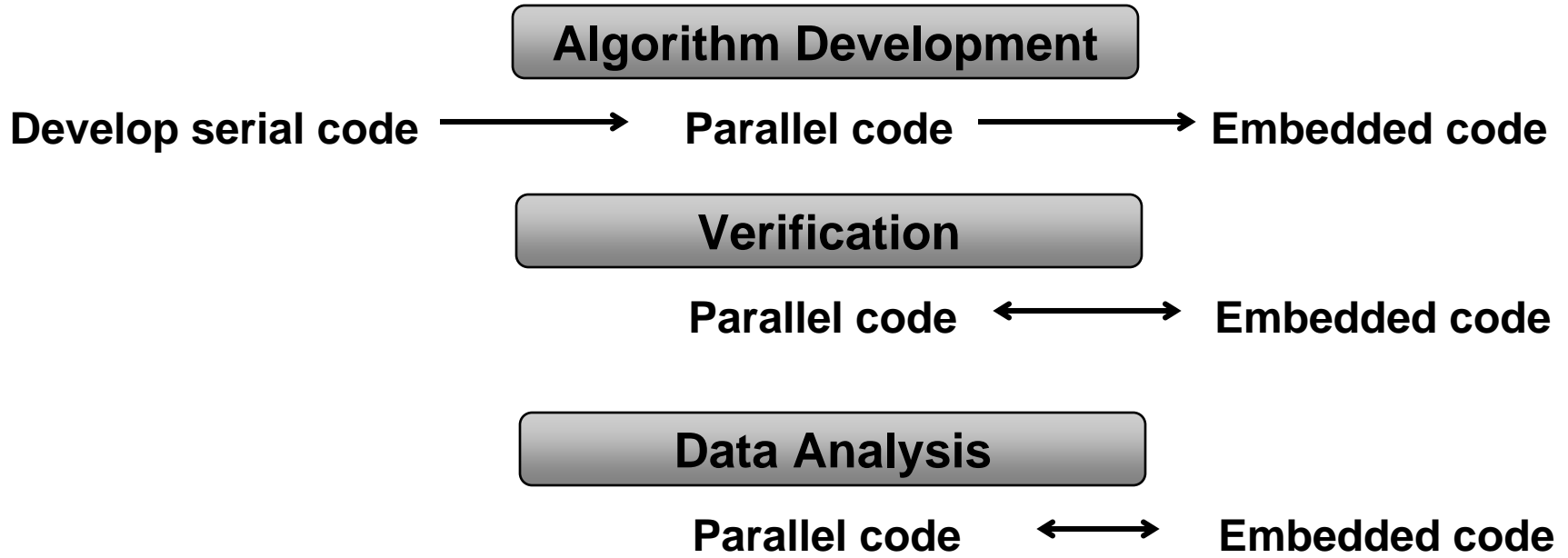
Memory Hierarchy



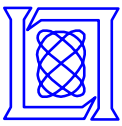
- All results in words/second
- Highlights memory hierarchy



Summary



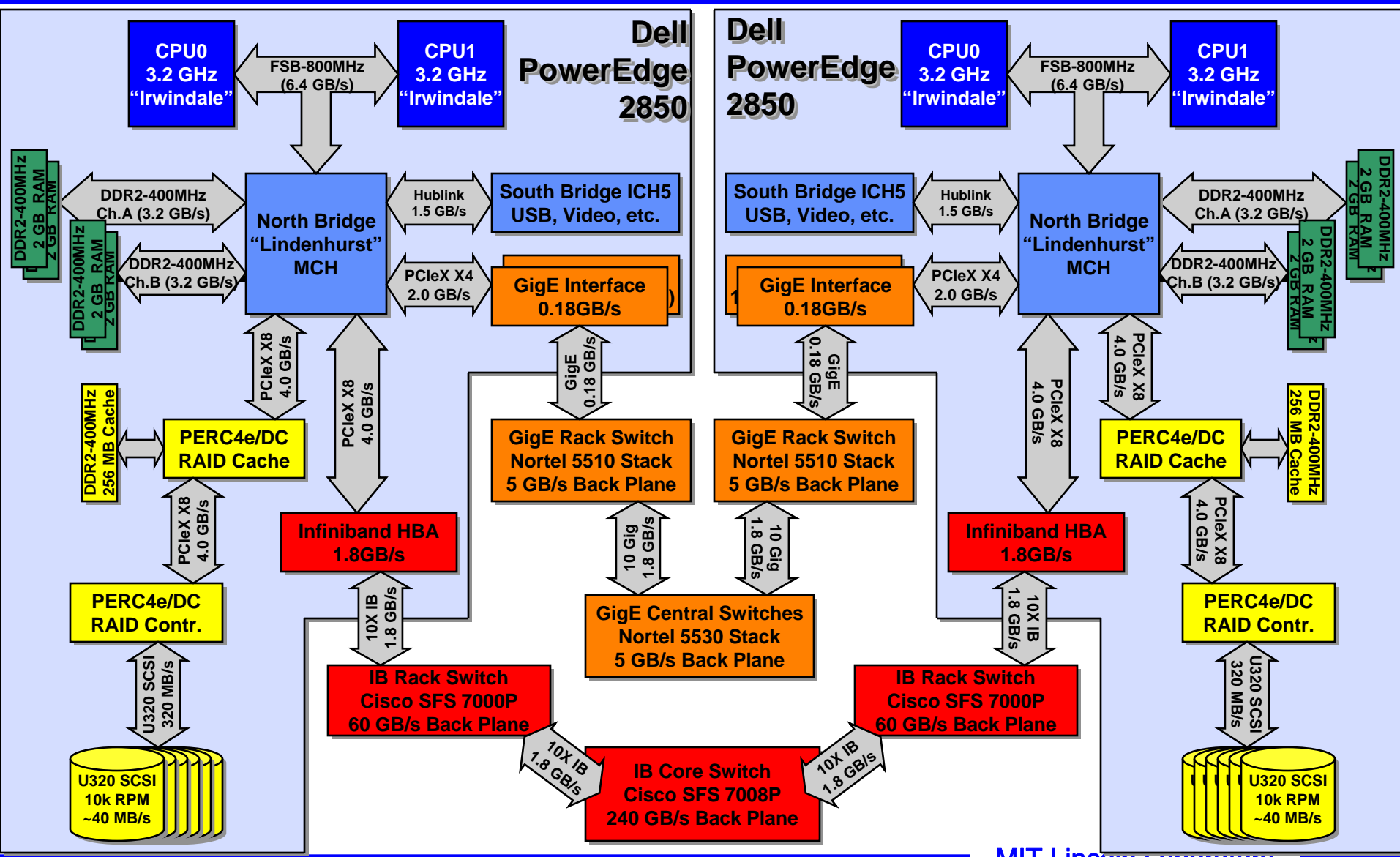
- Low barrier-to-entry
 - Interactive
 - Immediate execution
 - High Performance
- } Parallel Matlab and User Statistics
- } HPC Challenge and Results



Back Up Slides

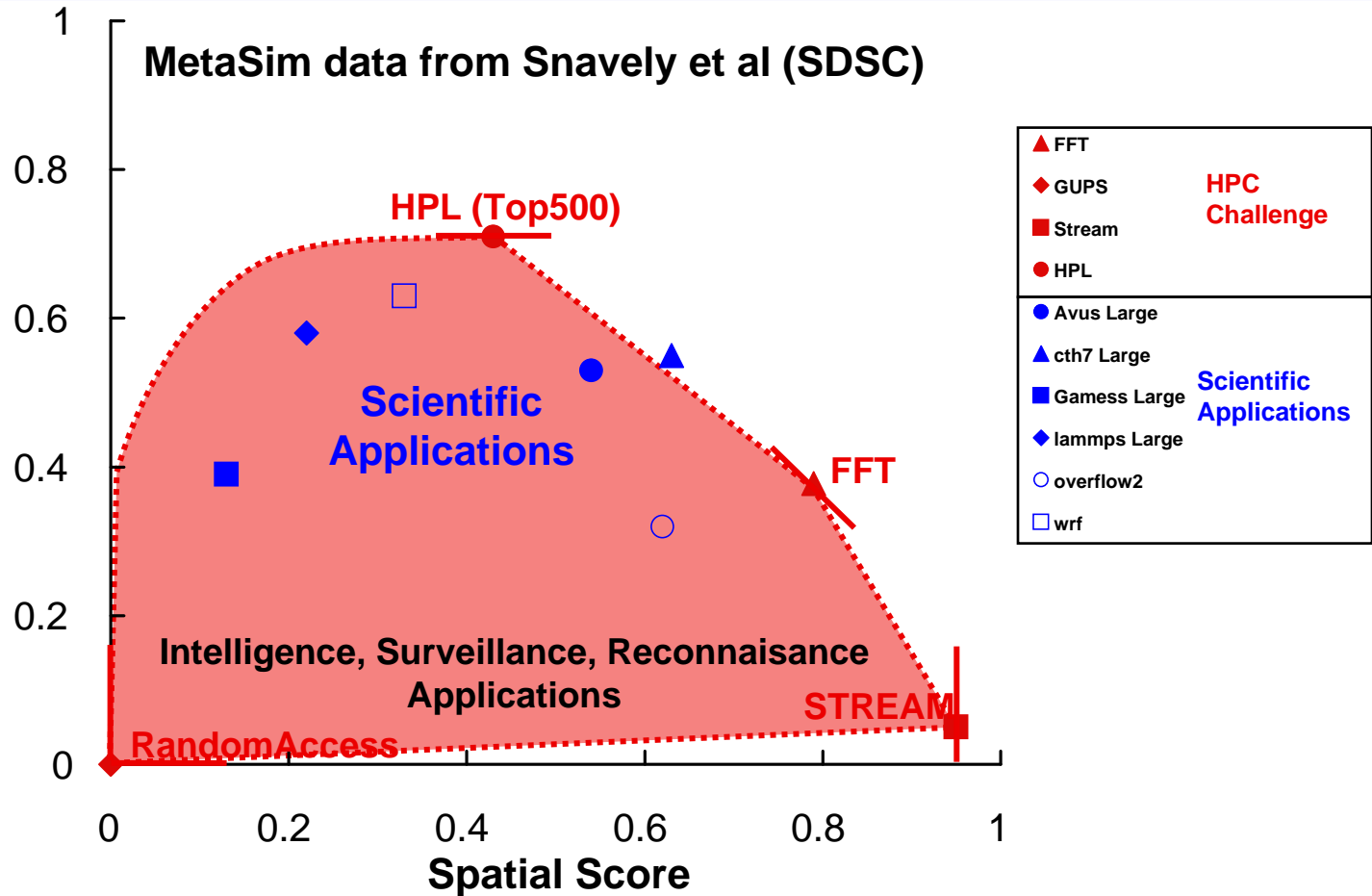
TX-2500: Analysis of Potential Bottlenecks

Peak Bandwidths





Spatial/Temporal Locality Results

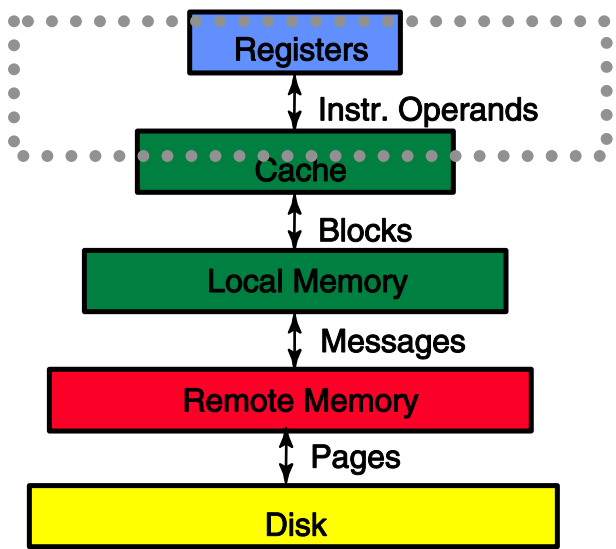


• **HPC Challenge bounds real applications**
 – Allows us to map between applications and benchmarks

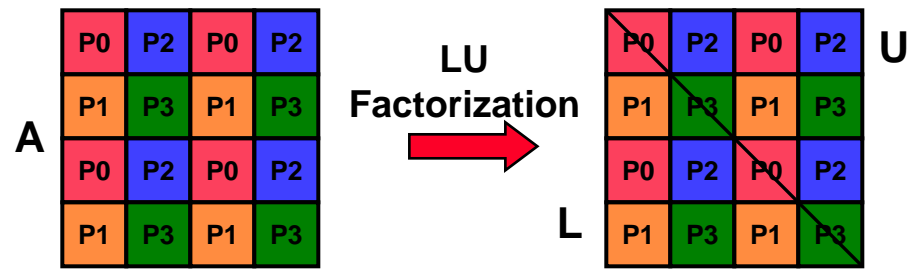


HPL “Top500” Benchmark

- High Performance Linpack (HPL) solves a system $Ax = b$
- Core operation is a LU factorization of a large $M \times M$ matrix
- Results are reported in floating point operations per second (flops)



Parallel Algorithm



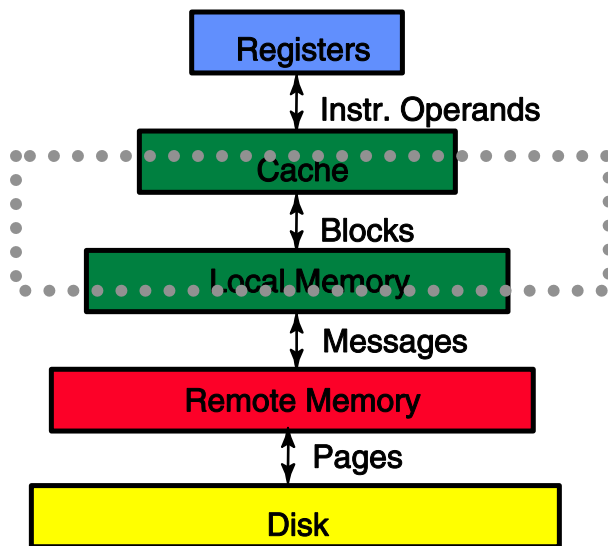
2D block cyclic distribution is used for load balancing

- Linear system solver (requires all-to-all communication)
- Stresses local matrix multiply performance
- DARPA HPCS goal: 2 Petaflops (8x over current best)



STREAM Benchmark

- Performs scalar multiply and add
- Results are reported in bytes/second



Parallel Algorithm

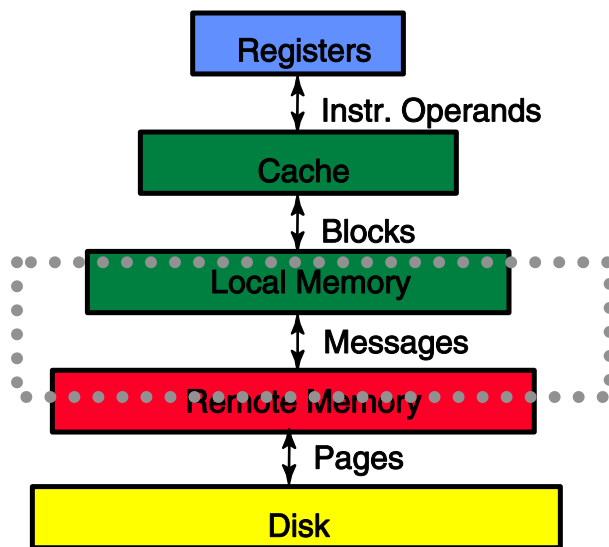
$$\begin{array}{r}
 \mathbf{A} \\
 = \\
 \mathbf{B} \\
 + \\
 \mathbf{s} \times \mathbf{C}
 \end{array}
 \begin{array}{c}
 \begin{array}{|c|c|} \hline 0 & 1 \\ \hline \end{array} \quad \cdots \quad \begin{array}{|c|} \hline \mathbf{Np-1} \\ \hline \end{array} \\
 \\
 \begin{array}{|c|c|} \hline 0 & 1 \\ \hline \end{array} \quad \cdots \quad \begin{array}{|c|} \hline \mathbf{Np-1} \\ \hline \end{array} \\
 \\
 \begin{array}{|c|c|} \hline 0 & 1 \\ \hline \end{array} \quad \cdots \quad \begin{array}{|c|} \hline \mathbf{Np-1} \\ \hline \end{array}
 \end{array}$$

- Basic operations on large vectors (requires no communication)
- Stresses local processor to memory bandwidth
- DARPA HPCS goal: 6.5 Petabytes/second (40x over current best)



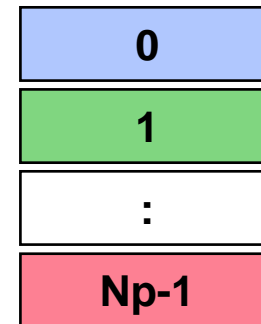
FFT Benchmark

- 1D Fast Fourier Transforms an N element complex vector
- Typically done as a parallel 2D FFT
- Results are reported in floating point operations per second (flops)

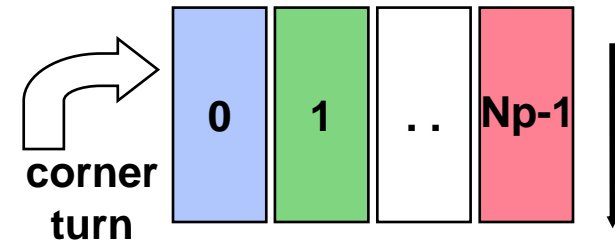


Parallel Algorithm

FFT rows →



FFT columns

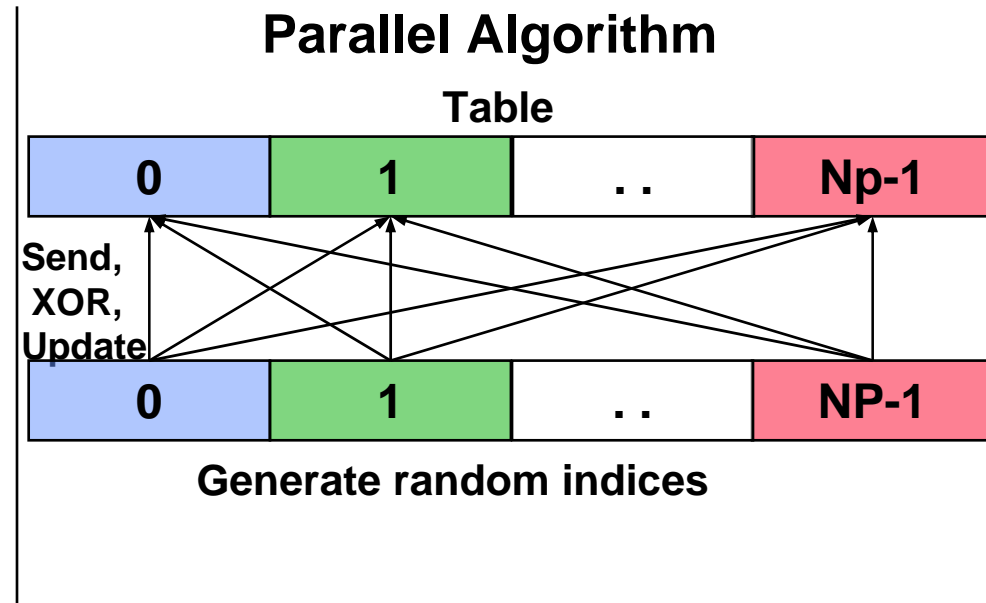
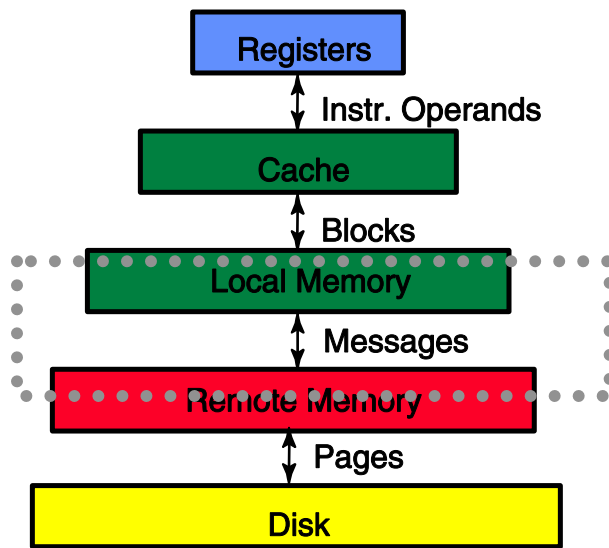


- FFT a large complex vector (requires all-to-all communication)
- Stresses interprocessor communication of *large* messages
- DARPA HPCS goal: 0.5 Petaflops (200x over current best)



RandomAccess Benchmark

- Randomly updates N element table of unsigned integers
- Each processor generates indices, sends to all other processors, performs XOR
- Results are reported in Giga Updates Per Second (GUPS)



- Randomly updates memory (requires all-to-all communication)
- Stresses interprocessor communication of *small* messages
- DARPA HPCS goal: 64,000 GUPS (2000x over current best)



IO Zone File System Benchmark

- File system benchmark tool
- Can measure large variety of file system characteristics
- We benchmarked:
 - Read and write throughput performance (MB/s)
 - 16 GB files (to test RAID, not caches)
 - 64 kB blocks (best performance on hardware)
 - On six-disk hardware RAID set
 - On all 432 compute nodes

Our iozone tests



Local Disk Memory Hierarchy

