HPCS HPCchallenge Benchmark Suite

David Koester, Ph.D. (MITRE)
Jack Dongarra (UTK)
Piotr Luszczek (ICL/UTK)

28 September 2004

Outline

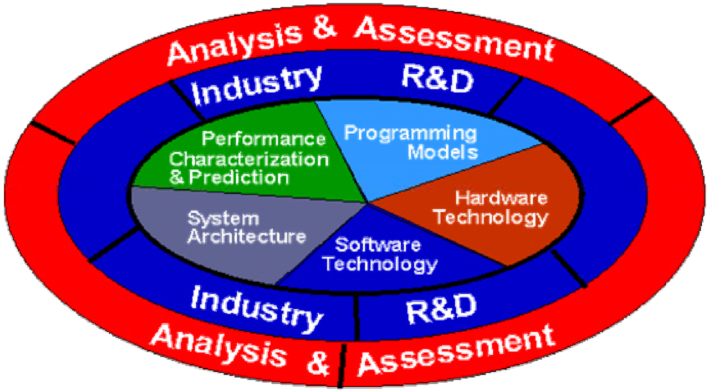
- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

High Productivity Computing Systems

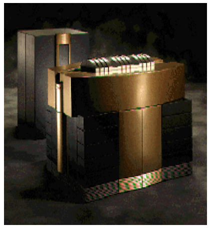
- **Create a new generation of economically viable computing systems and a procurement methodology for the security/industrial community (2007 – 2010)**

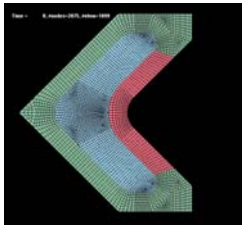
Impact:

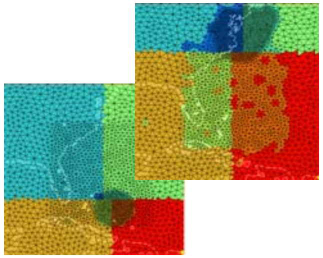
- **Performance** (time-to-solution): speedup critical national security applications by a factor of 10X to 40X
- **Programmability** (idea-to-first-solution): reduce cost and time of developing application solutions
- **Portability** (transparency): insulate research and operational application software from system
- **Robustness** (reliability): apply all known techniques to **protect against outside attacks**, hardware faults, & programming errors

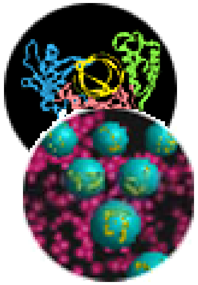


HPCS Program Focus Areas









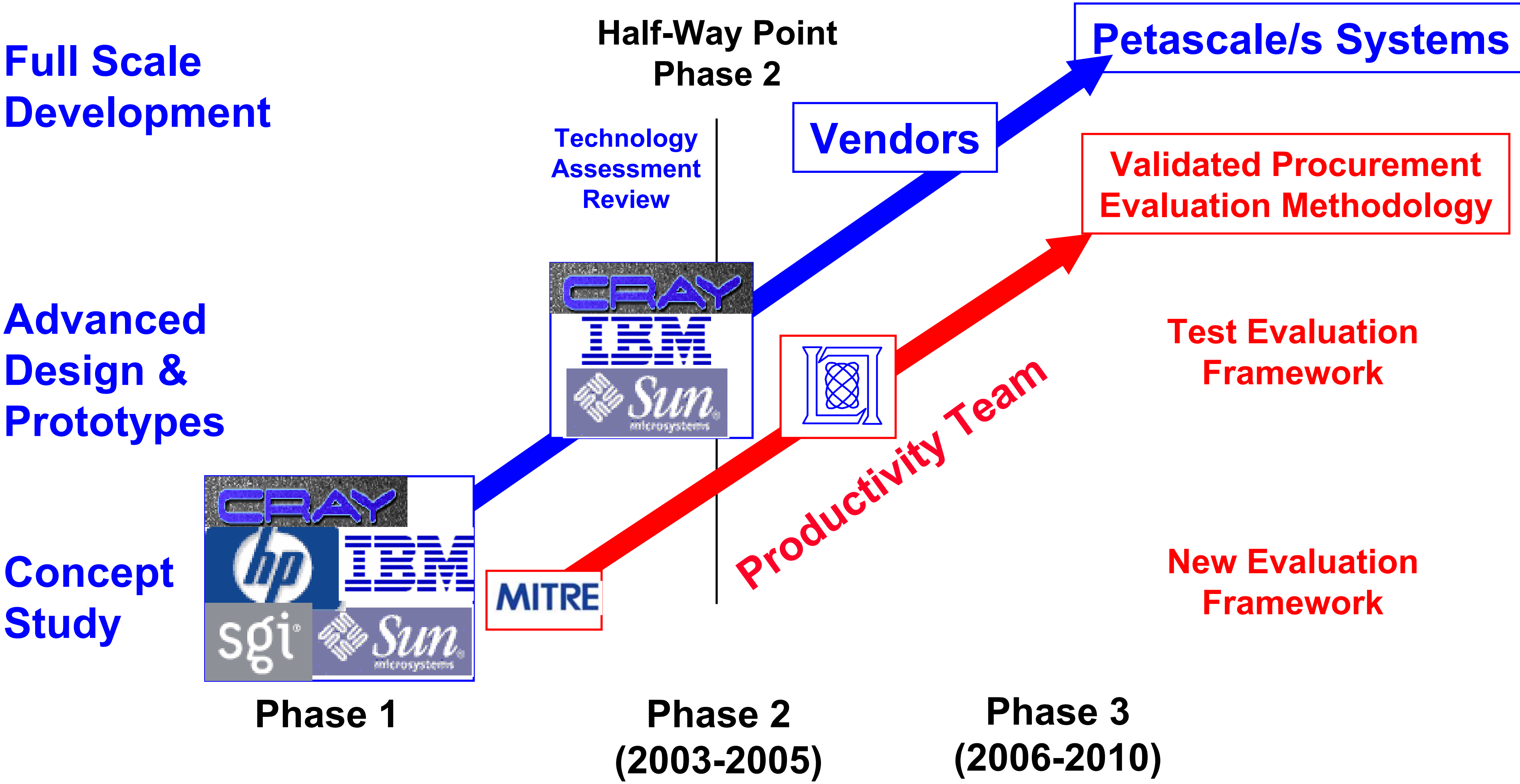
Applications:

- Intelligence/surveillance, reconnaissance, cryptanalysis, weapons analysis, airborne contaminant modeling and biotechnology

Fill the Critical Technology and Capability Gap
Today (late 80's HPC technology).....*to*.....Future (Quantum/Bio Computing)

High Productivity Computing Systems
-Program Overview-

- Create a new generation of economically viable computing systems and a procurement methodology for the security/industrial community (2007 – 2010)



Full Scale Development

Advanced Design & Prototypes

Concept Study

Half-Way Point Phase 2

Technology Assessment Review

Petascale/s Systems

Vendors

Validated Procurement Evaluation Methodology

Test Evaluation Framework

New Evaluation Framework

Productivity Team

Phase 1

Phase 2 (2003-2005)

Phase 3 (2006-2010)

HPCS Program Goals‡

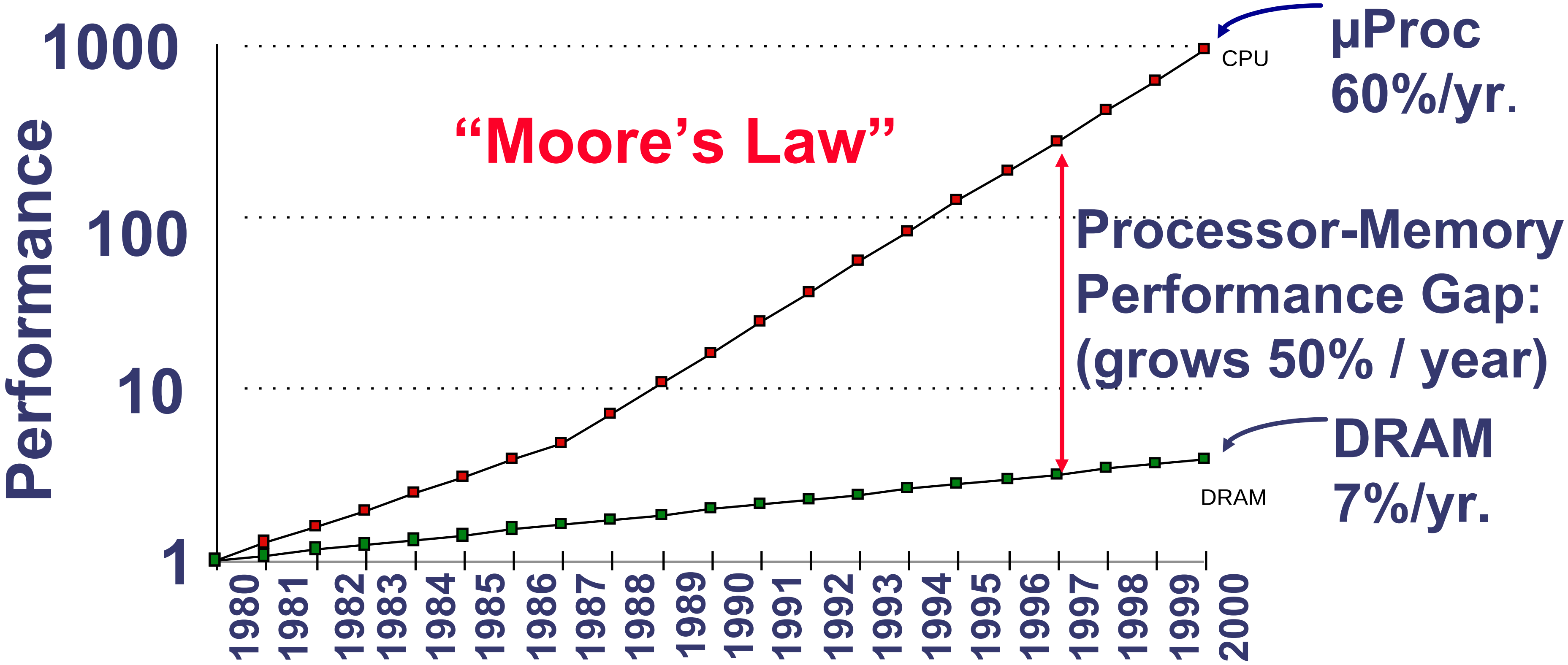
- HPCS overall productivity goals:
  - Execution (sustained performance)
    - 1 Petaflop/sec (scalable to greater than 4 Petaflop/sec)
    - Reference: Production workflow
  - Development
    - 10X over today's systems
    - Reference: Lone researcher and Enterprise workflows
- Productivity Framework
  - Base lined for today's systems
  - Successfully used to evaluate the vendors emerging productivity techniques
  - Provide a solid reference for evaluation of vendor's proposed Phase III designs.
- Subsystem Performance Indicators
  1) 2+ PF/s LINPACK
  2) 6.5 PB/sec data STREAM bandwidth
  3) 3.2 PB/sec bisection bandwidth
  4) 64,000 GUPS

‡Bob Graybill (DARPA/IPTO)
(Emphasis added)

Outline

- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

Processor-Memory Performance Gap



"Moore's Law"

µProc 60%/yr.

Processor-Memory Performance Gap: (grows 50% / year)

DRAM 7%/yr.

- Alpha 21264 full cache miss / instructions executed: 180 ns/1.7 ns =108 clks x 4 or 432 instructions
- Caches in Pentium Pro: 64% area, 88% transistors

*Taken from Patterson-Keeton Talk to SigMod

Processing vs. Memory Access

- Doesn't cache solve this problem?
  - It depends. With small amounts of contiguous data, usually. With large amounts of non-contiguous data, usually not
  - In most computers the programmer has no control over cache
  - Often "a few" Bytes/FLOP is considered OK

- However, consider operations on the transpose of a matrix (e.g., for adjunct problems)
  - Xa= b $X^Ta = b$
  - If X is big enough, 100% cache misses are guaranteed, and we need at least 8 Bytes/FLOP (assuming a and b can be held in cache)

- Latency and limited bandwidth of processor-memory and node-node communications are major limiters of performance for scientific computation

Processing vs. Memory Access High Performance LINPACK

Consider another benchmark: Linpack

$$A\,x = b$$

Solve this linear equation for the vector x, where A is a known matrix, and b is a known vector. Linpack uses the BLAS routines, which divide A into blocks.

On the average Linpack requires 1 memory reference for every 2 FLOPs, or 4Bytes/Flop.

Many of these can be cache references

Processing vs. Memory Access STREAM TRIAD

Consider the simple benchmark: STREAM TRIAD

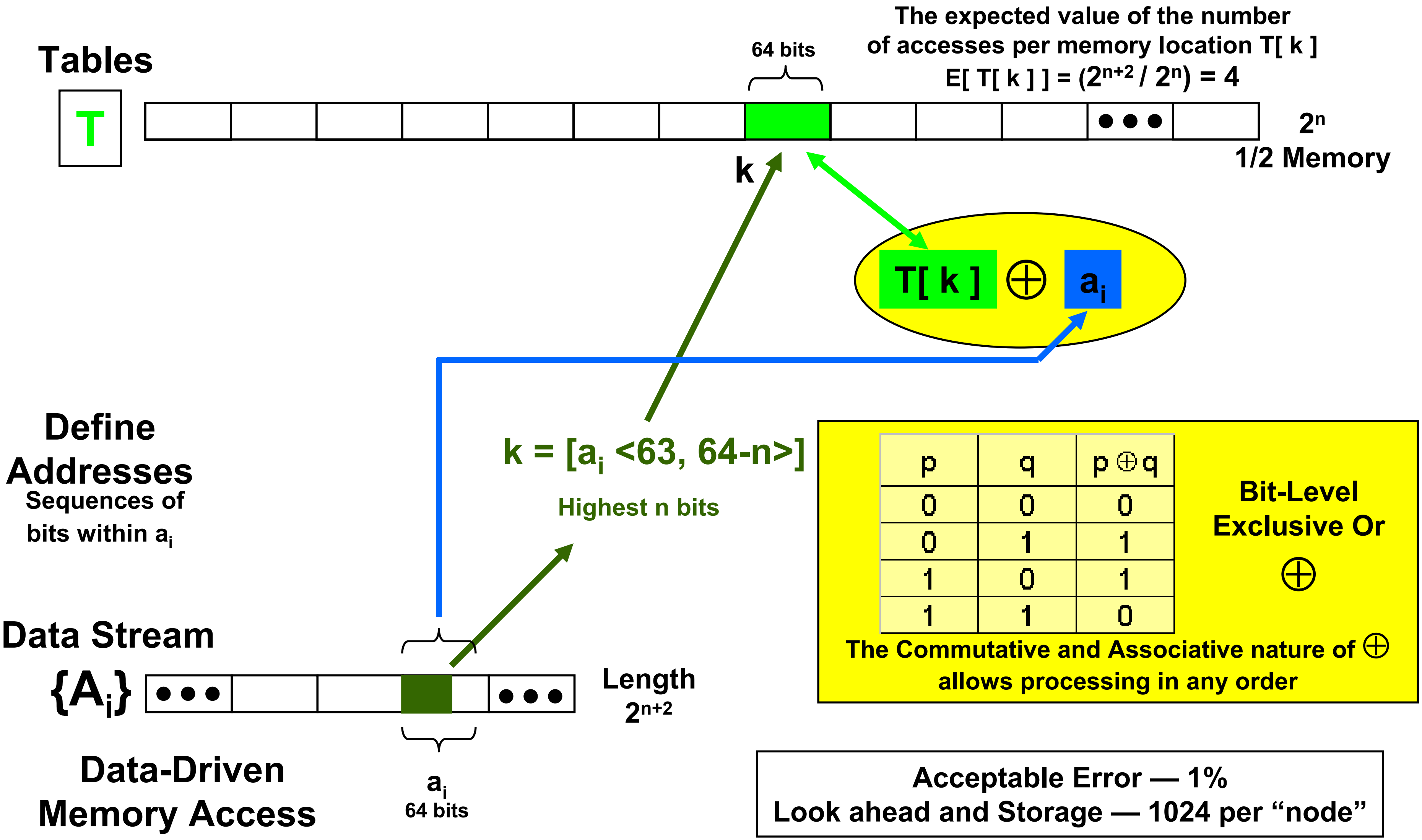
```
a(i) = b(i) + q * c(i)
```

`a(i)`, `b(i)`, and `c(i)` are vectors; q is a scalar
Vector length is chosen to be much longer than cache size

Each execution includes
2 memory loads + 1 memory store
2 FLOPs
12 Bytes/FLOP (assuming 32 bit precision)

No computer has enough memory bandwidth to reference
12 Bytes for each FLOP!

Processing vs. Memory Access RandomAccess



The expected value of the number of accesses per memory location T[k]

$E[\, T[\, k\,]\,] = (2^{n+2} / 2^n) = 4$

Define Addresses

Sequences of bits within a_i

$k = [a_i <63, 64\text{-}n>]$

Highest n bits

Data-Driven Memory Access

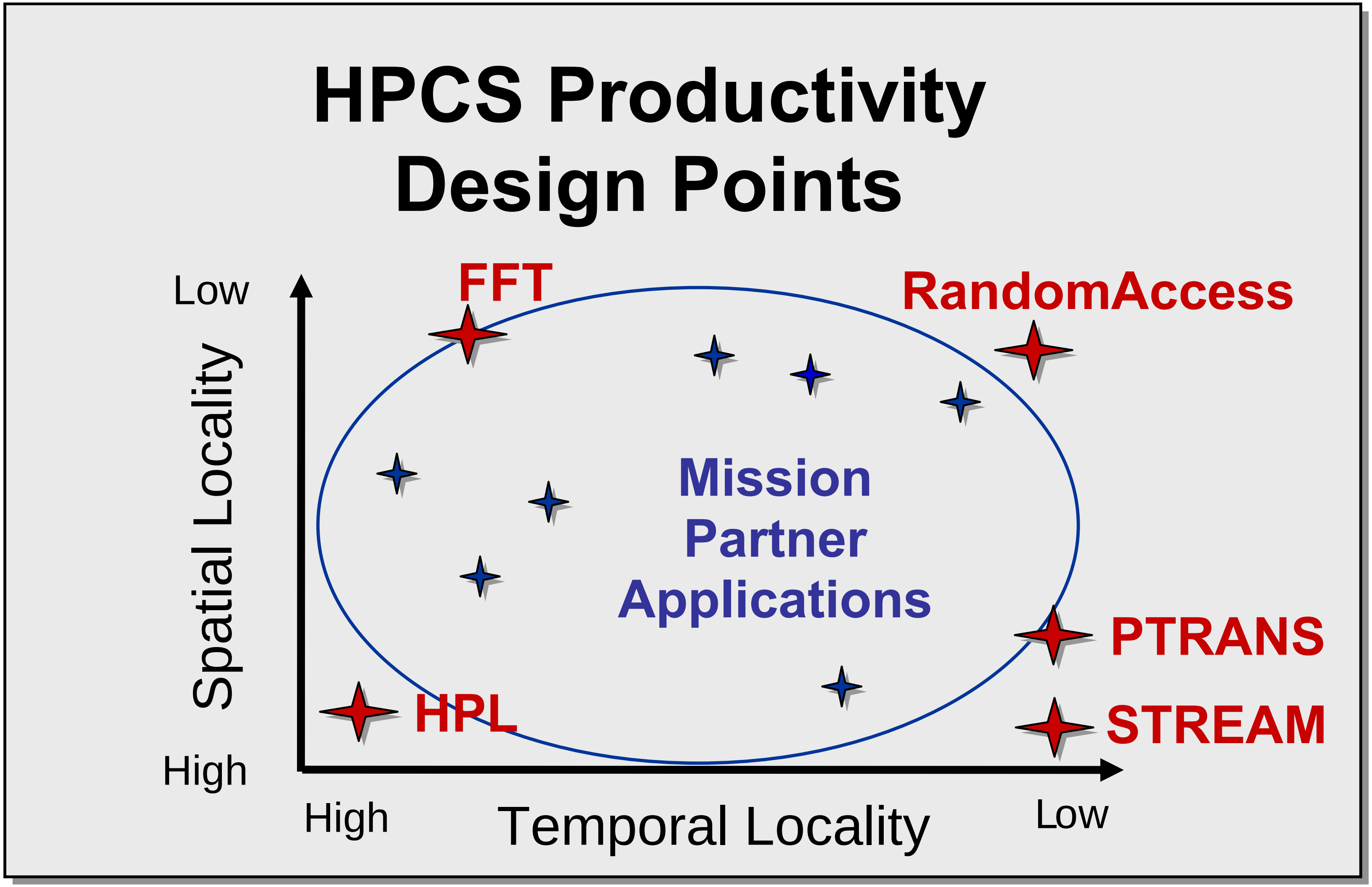
p	q	p $\oplus$ q
0	0	0
0	1	1
1	0	1
1	1	0

Bit-Level Exclusive Or $\oplus$

The Commutative and Associative nature of $\oplus$ allows processing in any order

Acceptable Error — 1%
Look ahead and Storage — 1024 per “node”

Bounding Mission Partner Applications



Outline

- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

HPCS HPCchallenge Benchmarks

- HPCSchallenge Benchmarks
  - Being developed by Jack Dongarra (ICL/UT)
  - Funded by the DARPA High Productivity Computing Systems (HPCS) program (Bob Graybill (DARPA/IPTO))

To examine the performance of High Performance Computer (HPC) architectures using kernels with more *challenging* memory access patterns than High Performance Linpack (HPL)

HPCchallenge Goals

- To examine the performance of HPC architectures using kernels with more *challenging* memory access patterns than HPL
  - HPL works well on all architectures — even cache-based, distributed memory multiprocessors due to
    1. Extensive memory reuse
    2. Scalable with respect to the amount of computation
    3. Scalable with respect to the communication volume
    4. Extensive optimization of the software
- To *complement* the Top500 list
- To provide benchmarks that *bound* the performance of many real applications as a function of memory access characteristics — e.g., spatial and temporal locality

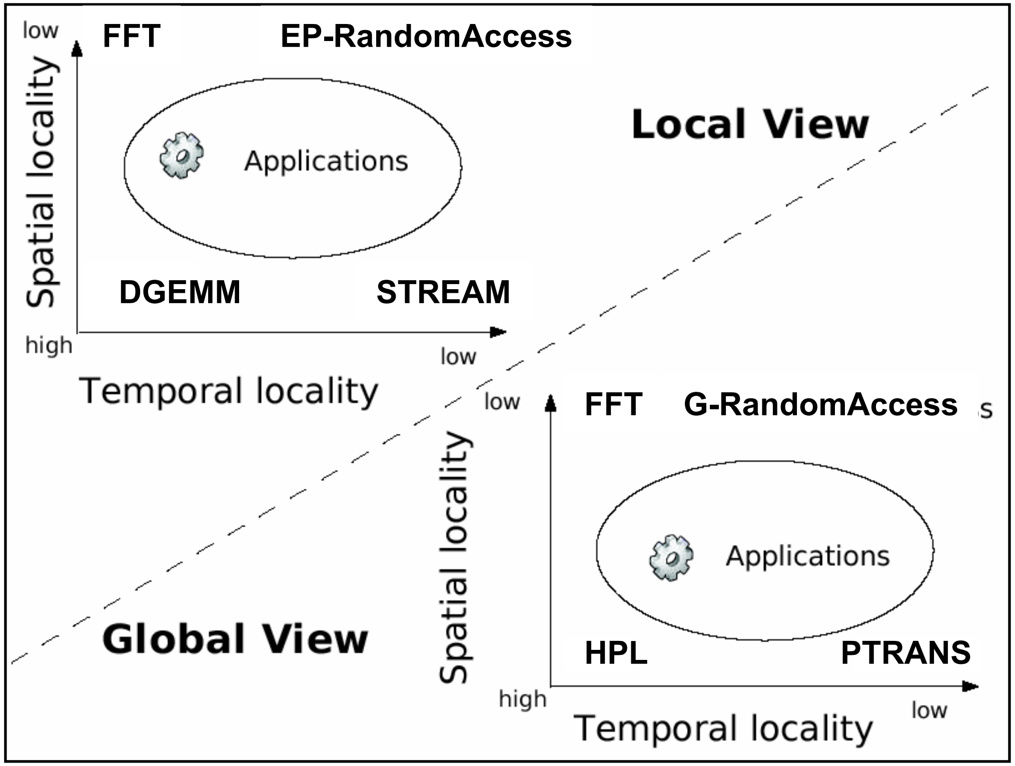
HPCchallenge Benchmarks

Local

- DGEMM (matrix x matrix multiply)
- STREAM
  - COPY
  - SCALE
  - ADD
  - TRIADD
- EP-RandomAccess
- 1D FFT

Global

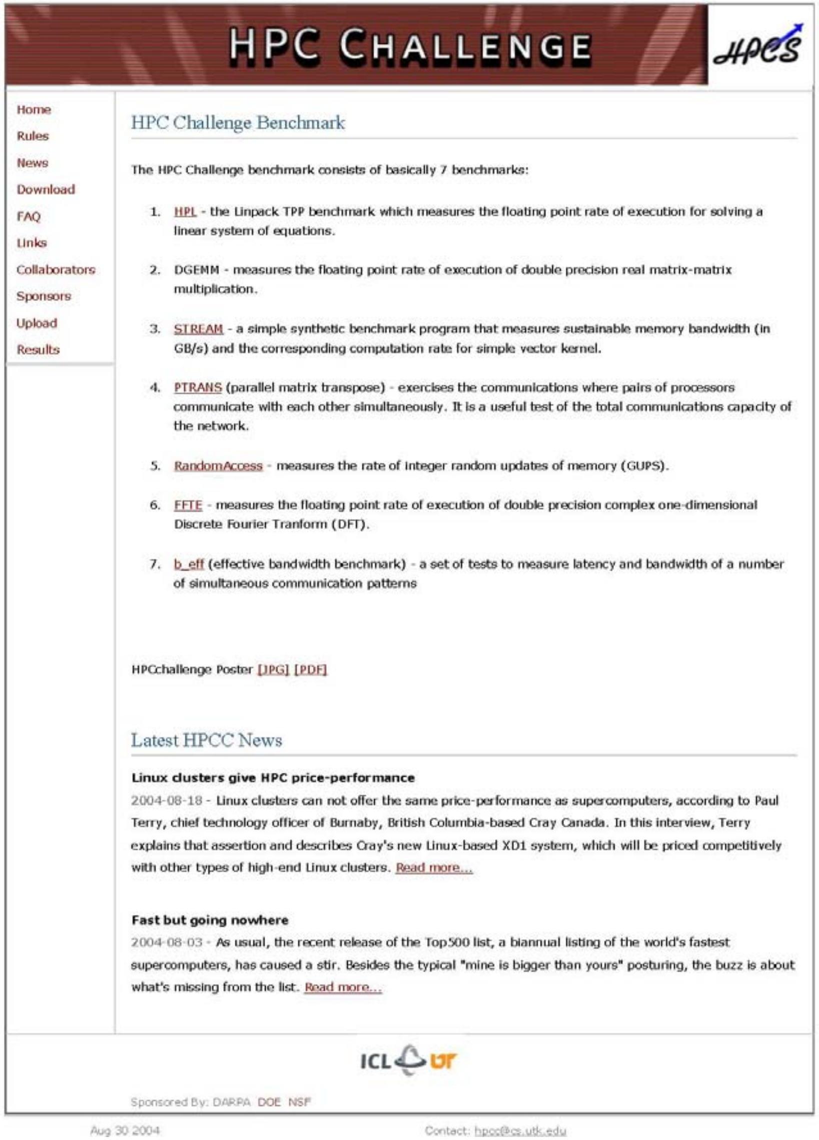
- High Performance LINPACK (HPL)
- PTRANS — parallel matrix transpose
- G-RandomAccess
- 1D FFT
- b_eff — interprocessor bandwidth and latency



- HPCchallenge pushes spatial and temporal boundaries; sets performance bounds
- Available for download http://icl.cs.utk.edu/hpcc/

Web Site
http://icl.cs.utk.edu/hpcc/

- Home
- Rules
- News
- Download
- FAQ
- Links
- Collaborators
- Sponsors
- Upload
- Results



HPC Challenge Benchmark

The HPC Challenge benchmark consists of basically 7 benchmarks:

1. HPL - the Linpack TPP benchmark which measures the floating point rate of execution for solving a linear system of equations.
2. DGEMM - measures the floating point rate of execution of double precision real matrix-matrix multiplication.
3. STREAM - a simple synthetic benchmark program that measures sustainable memory bandwidth (in GB/s) and the corresponding computation rate for simple vector kernel.
4. PTRANS (parallel matrix transpose) - exercises the communications where pairs of processors communicate with each other simultaneously. It is a useful test of the total communications capacity of the network.
5. RandomAccess - measures the rate of integer random updates of memory (GUPS).
6. FFTE - measures the floating point rate of execution of double precision complex one-dimensional Discrete Fourier Tranform (DFT).
7. b_eff (effective bandwidth benchmark) - a set of tests to measure latency and bandwidth of a number of simultaneous communication patterns

HPCchallenge Poster [JPG] [PDF]

Latest HPCC News

Linux clusters give HPC price-performance

2004-08-18 - Linux clusters can not offer the same price-performance as supercomputers, according to Paul Terry, chief technology officer of Burnaby, British Columbia-based Cray Canada. In this interview, Terry explains that assertion and describes Cray's new Linux-based XD1 system, which will be priced competitively with other types of high-end Linux clusters. Read more...

Fast but going nowhere

2004-08-03 - As usual, the recent release of the Top500 list, a biannual listing of the world's fastest supercomputers, has caused a stir. Besides the typical "mine is bigger than yours" posturing, the buzz is about what's missing from the list. Read more...

Sponsored By: DARPA DOE NSF

Aug 30 2004 Contact: hpcc@cs.utk.edu

Outline

- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

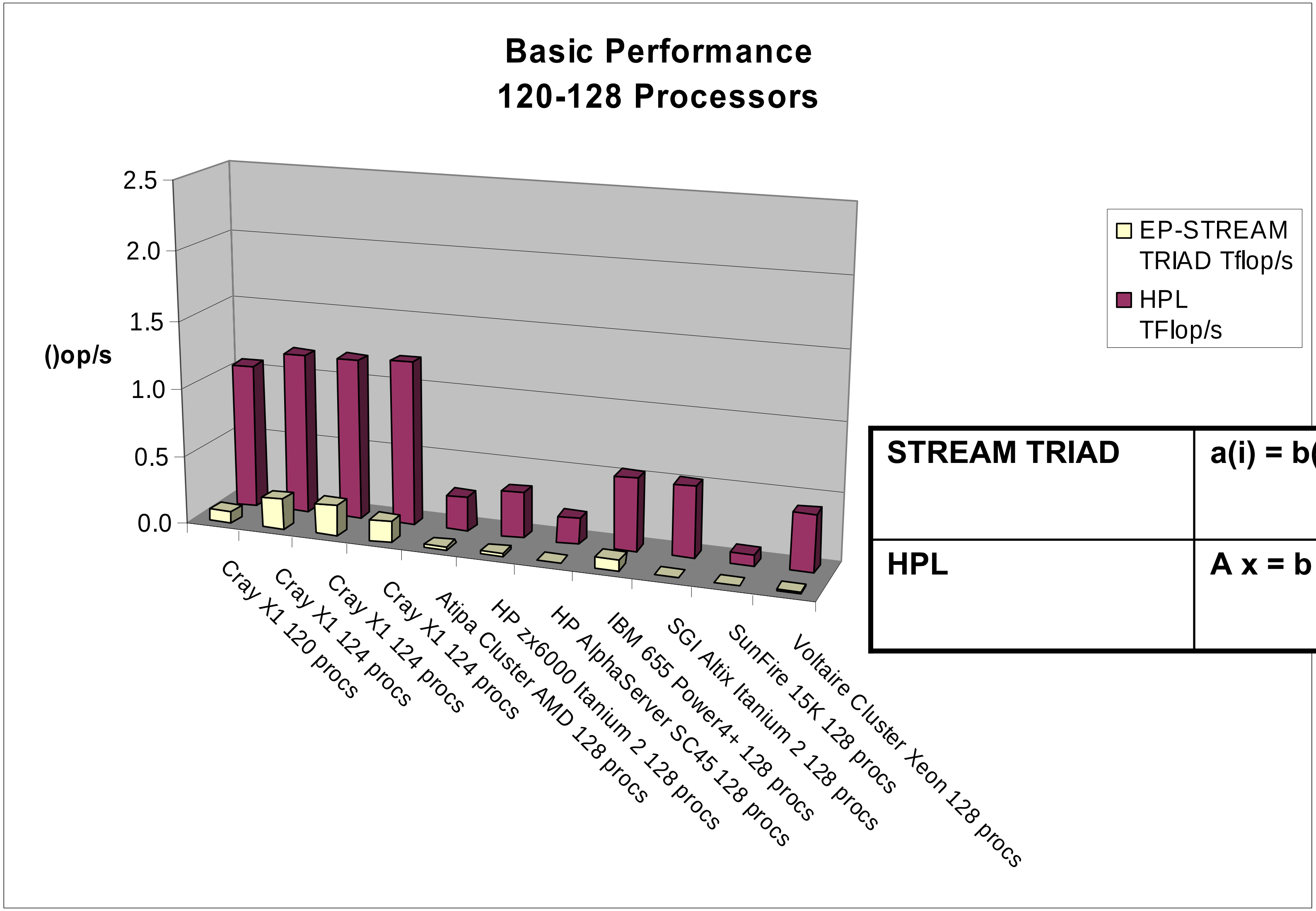
Preliminary Results Machine List (1 of 2)

Affiliatio n	Manufacturer	System	ProcessorType	Procs
U Tenn	Atipa Cluster AMD 128 procs	Conquest cluster	AMD Opteron	128
AHPCRC	Cray X1 124 procs	X1	Cray X1 MSP	124
AHPCRC	Cray X1 124 procs	X1	Cray X1 MSP	124
AHPCRC	Cray X1 124 procs	X1	Cray X1 MSP	124
ERDC	Cray X1 60 procs	X1	Cray X1 MSP	60
ERDC	Cray X1 60 procs	X1	Cray X1 MSP	60
ORNL	Cray X1 252 procs	X1	Cray X1 MSP	252
ORNL	Cray X1 252 procs	X1	Cray X1 MSP	252
AHPCRC	Cray X1 120 procs	X1	Cray X1 MSP	120
ORNL	Cray X1 64 procs	X1	Cray X1 MSP	64
AHPCRC	Cray T3E 1024 procs	T3E	Alpha 21164	1024
ORNL	HP zx6000 Itanium 2 128 procs	Integrity zx6000	Intel Itanium 2	128
PSC	HP AlphaServer SC45 128 procs	AlphaServer SC45	Alpha 21264B	128
ERDC	HP AlphaServer SC45 484 procs	AlphaServer SC45	Alpha 21264B	484

Preliminary Results
Machine List (2 of 2)

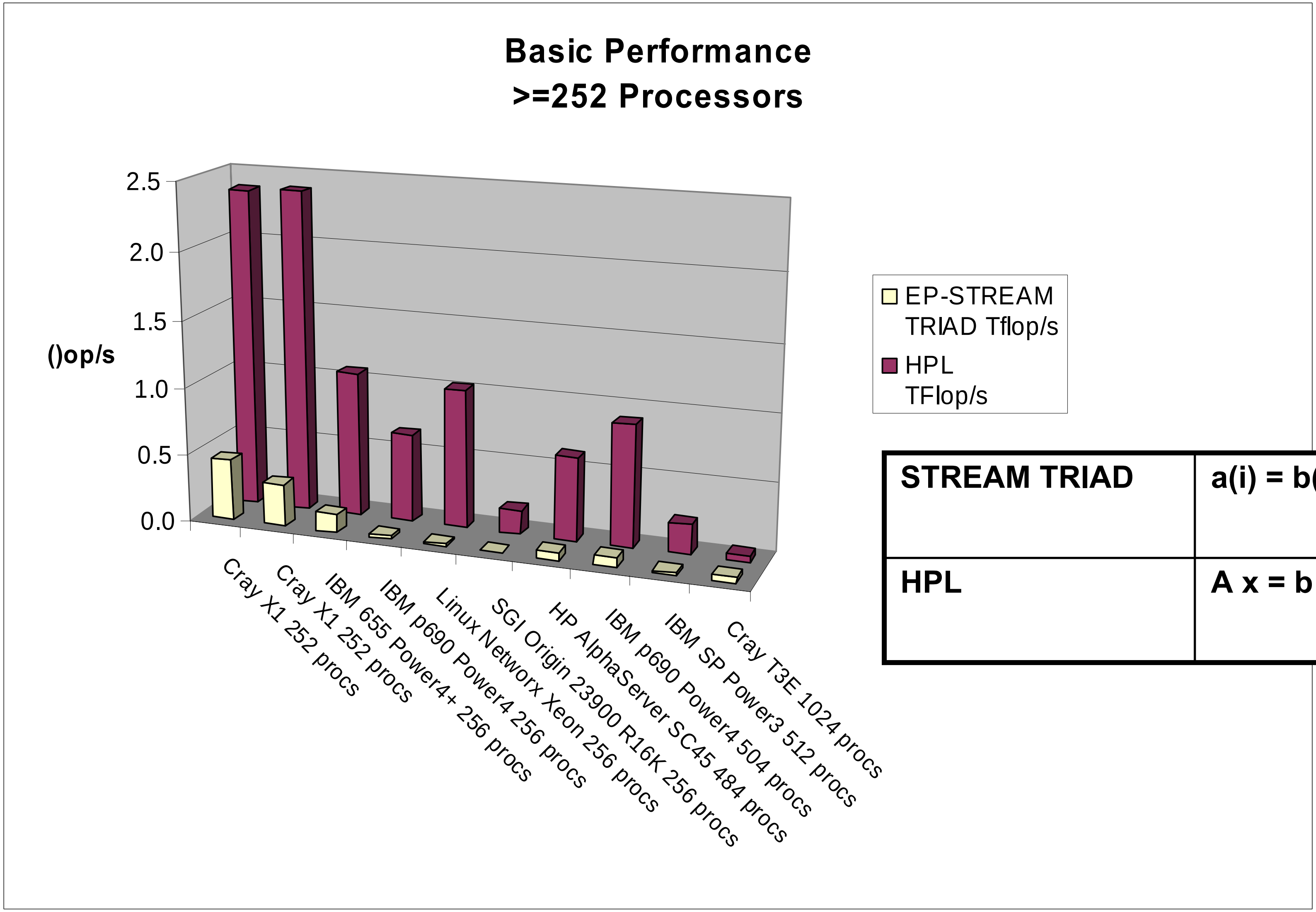
Affiliation	Manufacturer	System	ProcessorType	Procs
IBM	IBM 655 Power4+ 64 procs	eServer pSeries 655	IBM Power 4+	64
IBM	IBM 655 Power4+ 128 procs	eServer pSeries 655	IBM Power 4+	128
IBM	IBM 655 Power4+ 256 procs	eServer pSeries 655	IBM Power 4+	256
NAVO	IBM p690 Power4 504 procs	p690	IBM Power 4	504
ARL	IBM SP Power3 512 procs	RS/6000 SP	IBM Power 3	512
ORNL	IBM p690 Power4 256 procs	p690	IBM Power 4	256
ORNL	IBM p690 Power4 64 procs	p690	IBM Power 4	64
ARL	Linux Networx Xeon 256 procs	Powell	Intel Xeon	256
U Manchester	SGI Altix Itanium 2 32 procs	Altix 3700	Intel Itanium 2	32
ORNL	SGI Altix Itanium 2 128 procs	Altix	Intel Itanium 2	128
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
NASA ASC	SGI Origin 23900 R16K 256 procs	Origin 3900	SGI MIPS R16000	256
U Aachen/RWTH	SunFire 15K 128 procs	Sun Fire 15k/6800 SMP-Cluster	Sun UltraSparc III	128
OSC	Voltaire Cluster Xeon 128 procs	Pinnacle 2X200 Cluster	Intel Xeon	128

STREAM TRIAD vs HPL 120-128 Processors



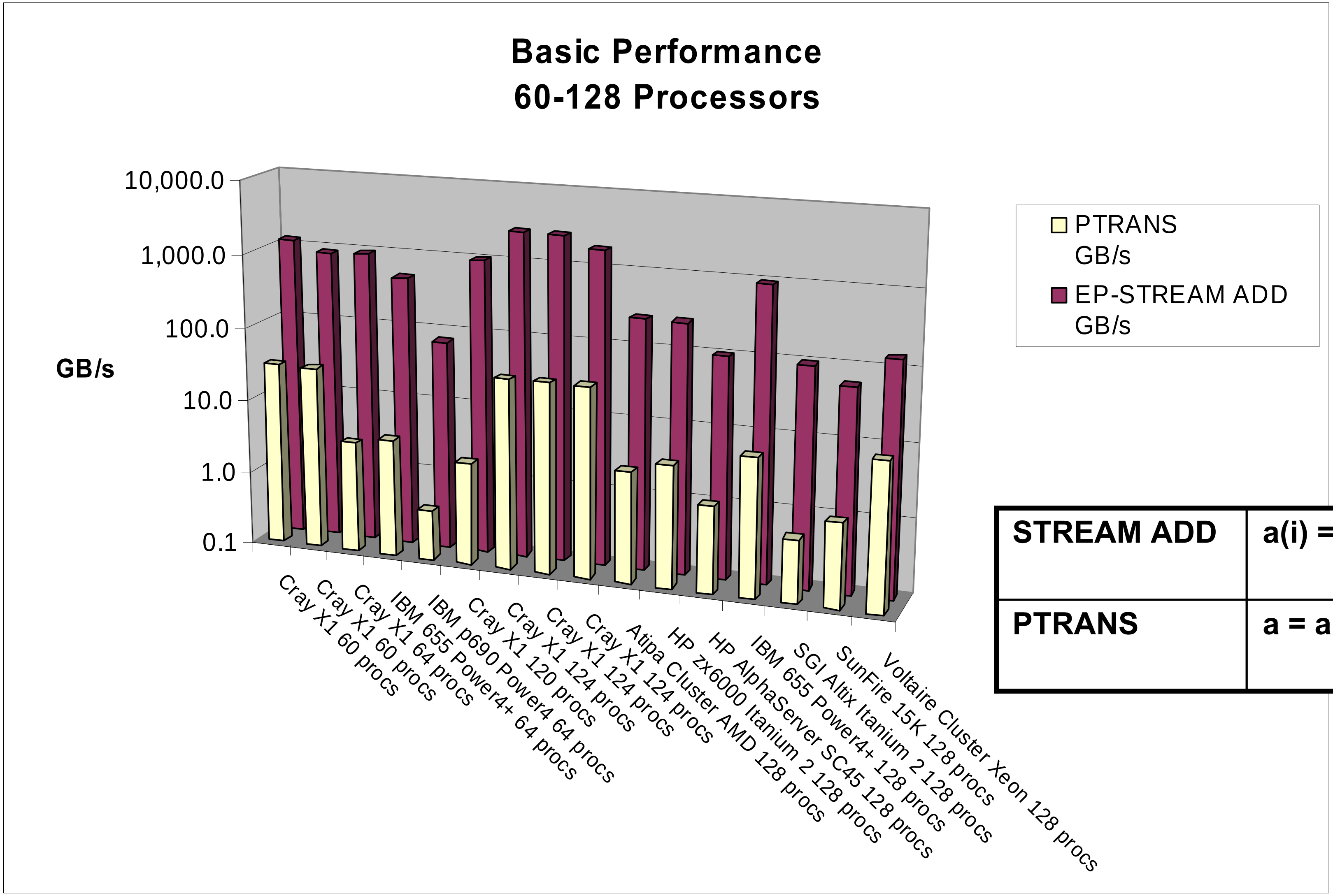
STREAM TRIAD	**a(i) = b(i) + q *c(i)**
HPL	**A x = b**

STREAM TRIAD vs HPL >252 Processors



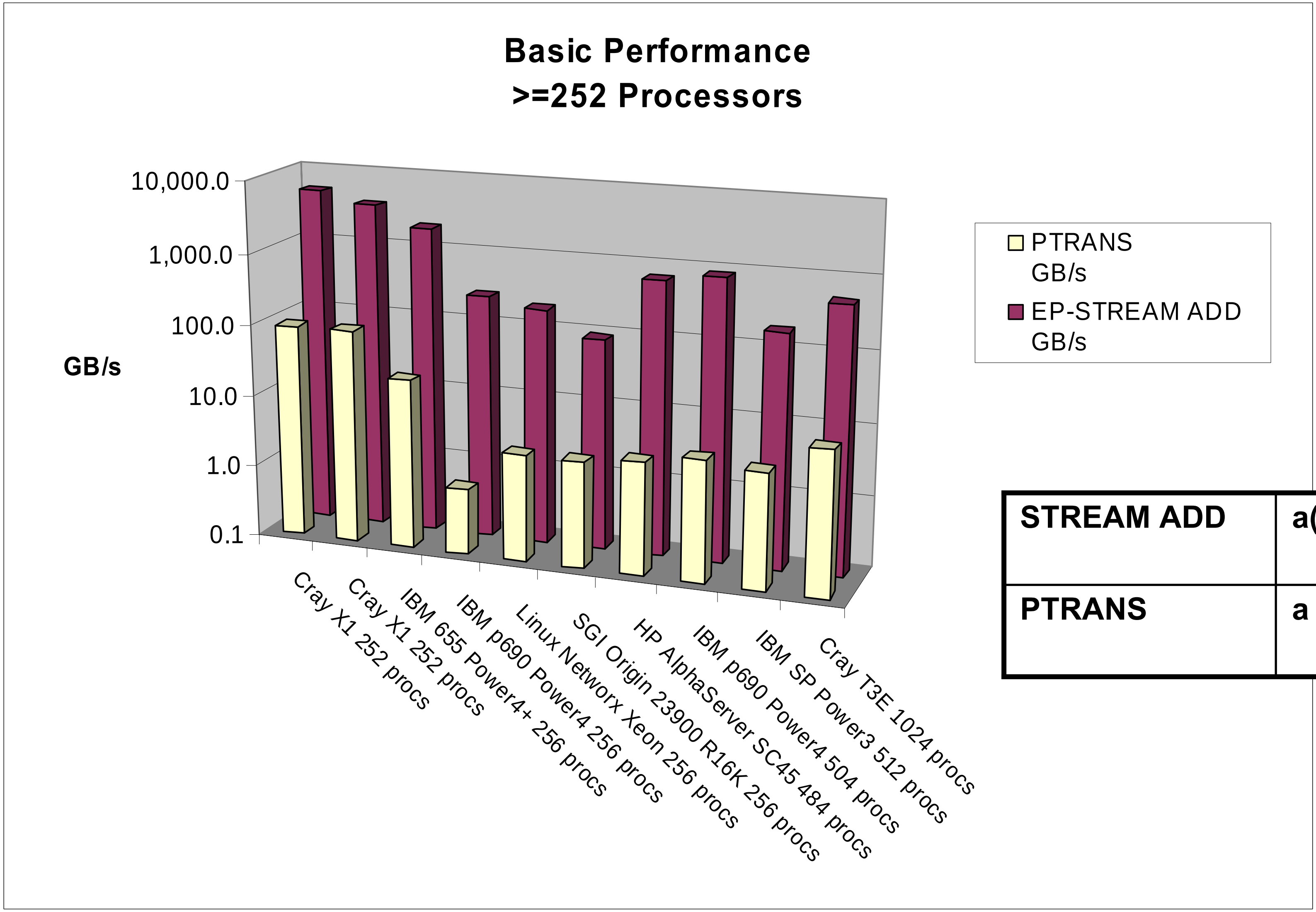
STREAM TRIAD	**a(i) = b(i) + q *c(i)**
HPL	**A x = b**

STREAM ADD vs PTRANS 60-128 Processors



STREAM ADD	**$a(i) = b(i) + c(i)$**
PTRANS	**$a = a + b^T$**

STREAM ADD vs PTRANS >252 Processors



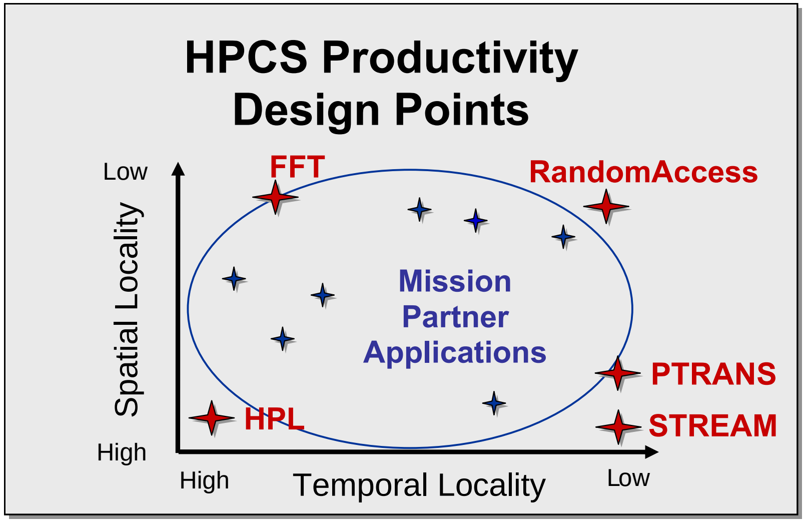
STREAM ADD	**$a(i) = b(i) + c(i)$**
PTRANS	**$a = a + b^T$**

Outline

- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

Summary

- DARPA HPCS Subsystem Performance Indicators
  - 2+ PF/s LINPACK
  - 6.5 PB/sec data STREAM bandwidth
  - 3.2 PB/sec bisection bandwidth
  - 64,000 GUPS
- Important to understand architecture/application characterization
  - Where did all the lost "Moore's Law performance go?"
- HPCchallenge Benchmarks — http://icl.cs.utk.edu/hpcc/
  - Peruse the results!
  - Contribute!


HPCS Productivity
Design Points
Low
Spatial Locality
High
FFT
RandomAccess
Mission
Partner
Applications
PTRANS
STREAM
HPL
High
Temporal Locality
Low