Data Intensive Computing and the Graph 500

Richard C. Murphy

Scalable Computer Architectures Department

Sandia National Laboratories

Affiliated Faculty, New Mexico State University

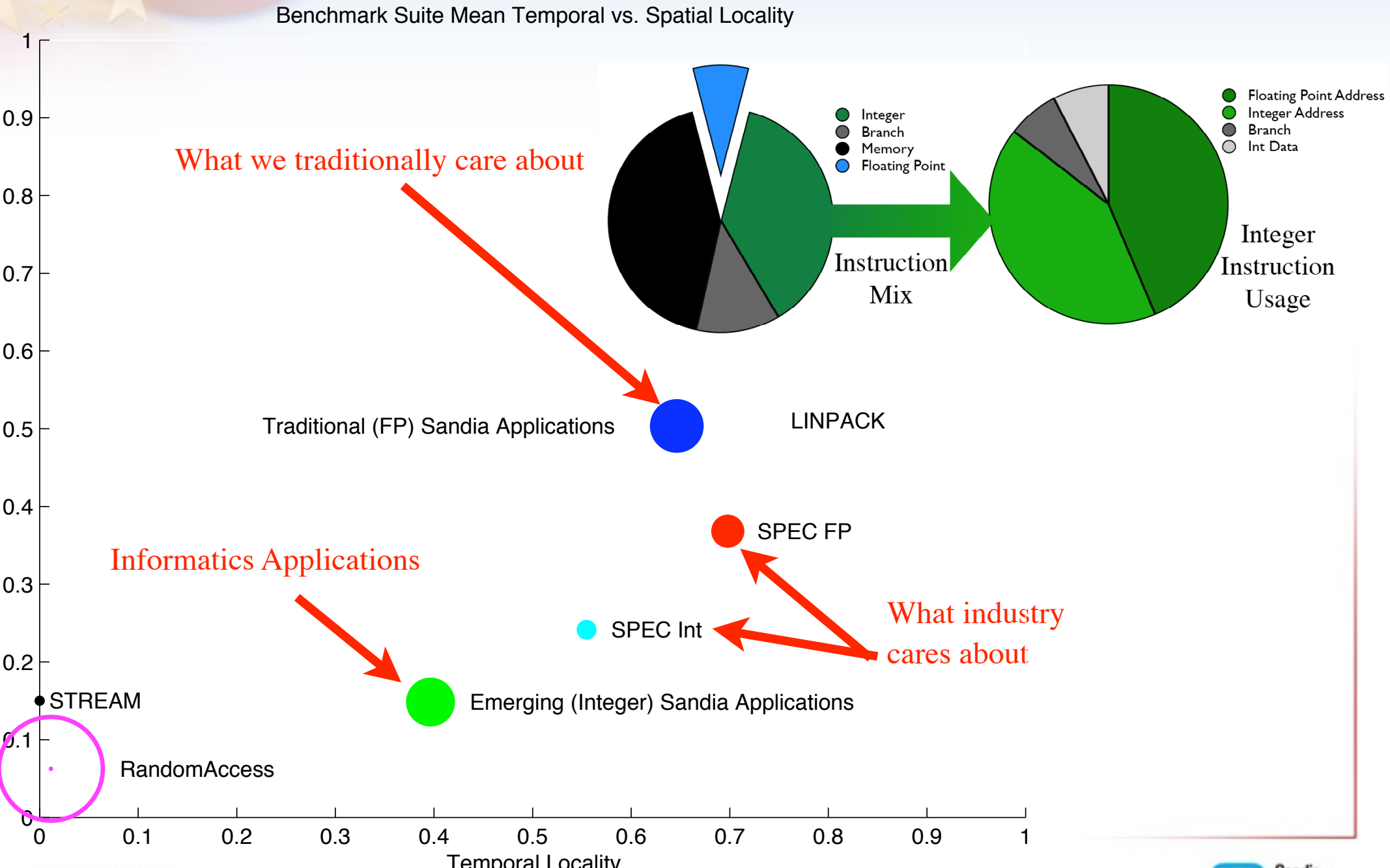
September 15, 2010

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.

What is My Definition of Data Intensive Computing?

- The problem is in N-space, not "the real world" (3D)
  - Requires Global Machine Reach (in communication and naming)
  - Generally extremely sparse problems with very little local work
- Application parallelism is abundant and asynchronous
  - e.g., concurrent graph search
- Dense techniques will not work
  - There is no hope for accelerators, GPGPUs, etc.
  - Worry about how many GUPS your machine can sustain
- Even a PGAS Load/Store architecture is probably insufficient
  - Need the ability to move work to the data

We have an emerging application/architecture mismatch...



From: Murphy and Kogge, *On The Memory Access Patterns of Supercomputer Applications: Benchmark Selection and Its Implications*, IEEE T. on Computers, July 2007

NNSA

Sandia National Laboratories

How do we know we got the right answer?

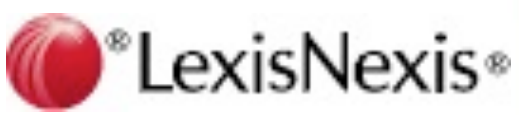
- "Data Intensive Computing" generally involves analyses of non-numeric data and the number of combinatorial possibilities grows rapidly. The objective of the analysis is to find in the data, meaningful relationships. How do we (a) test for convergence when not evaluating all possible combinations and (b) test for statistical significance--when the data is non-numeric?
  - (a) Build a Bigger Computer with Better Algorithms
    - If possible, equally effective faster methods (ensembles, neural networks, etc.)
  - (b) The data-intensive world is generally more about reasonable/ appropriate answers than statistical provability
    - If you want a deep answer, go read about Robert Hecht-Nielsen's confabulation theory...
      - Find the most likely not-know-to-be-wrong answer, not the Bayesian answer

Incomplete Data

- "Data Intensive Computing" often involves the use of incomplete data. How does this affect the analysis process?
- This is a survival of the fittest question... the person with more complete data or better algorithms will win by:
  - Being more profitable
  - Beating you to the nobel prize
  - Etc.
- Just like people, the "smarter" computer will be able to cope with incomplete data better
  - Or a lucky one..
- Again, try to find the most likely not-know-to-be-wrong answer, not the Bayesian answer

Now to the question that interests me most...

- If you could design an ideal computing architecture for Data Intensive Computing, what would it look like?
  - In fact we are doing this
  - We begin with data movement
    - It is the hardest problem
    - It consumes the most power
    - It defines success, despite people typically being more fixated on unsustainable peak FLOPS



Information Sciences Institute
USC Viterbi School of Engineering

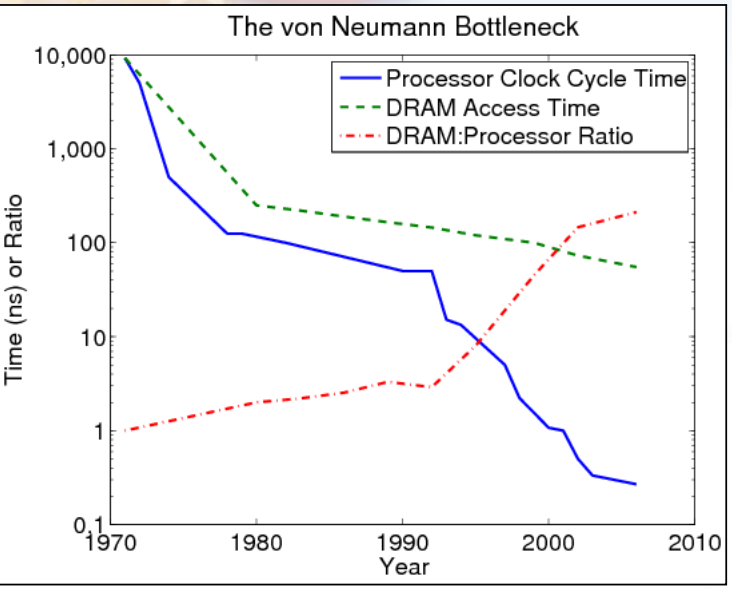
Goals of X-caliber: Reinvent Computing

- We (particularly the DOE "physics" community) are stuck in highly optimized model of computation
  - The execution model (MPI, mostly BSP) matches the architecture (MPP) which matches the applications (3D Physics)
  - but is it a local minima?
- However
  - Technology trends demand new architectures and threaten traditional "machine balance"
  - Applications are increasingly unstructured
    - Informatics informatics apps and even traditional physics apps
  - The result stresses the execution model
- Consequently
  - We have an opportunity to rethink the computer driven by application requirements
  - Will match technology, architecture, and execution model
  - Codesign is the process

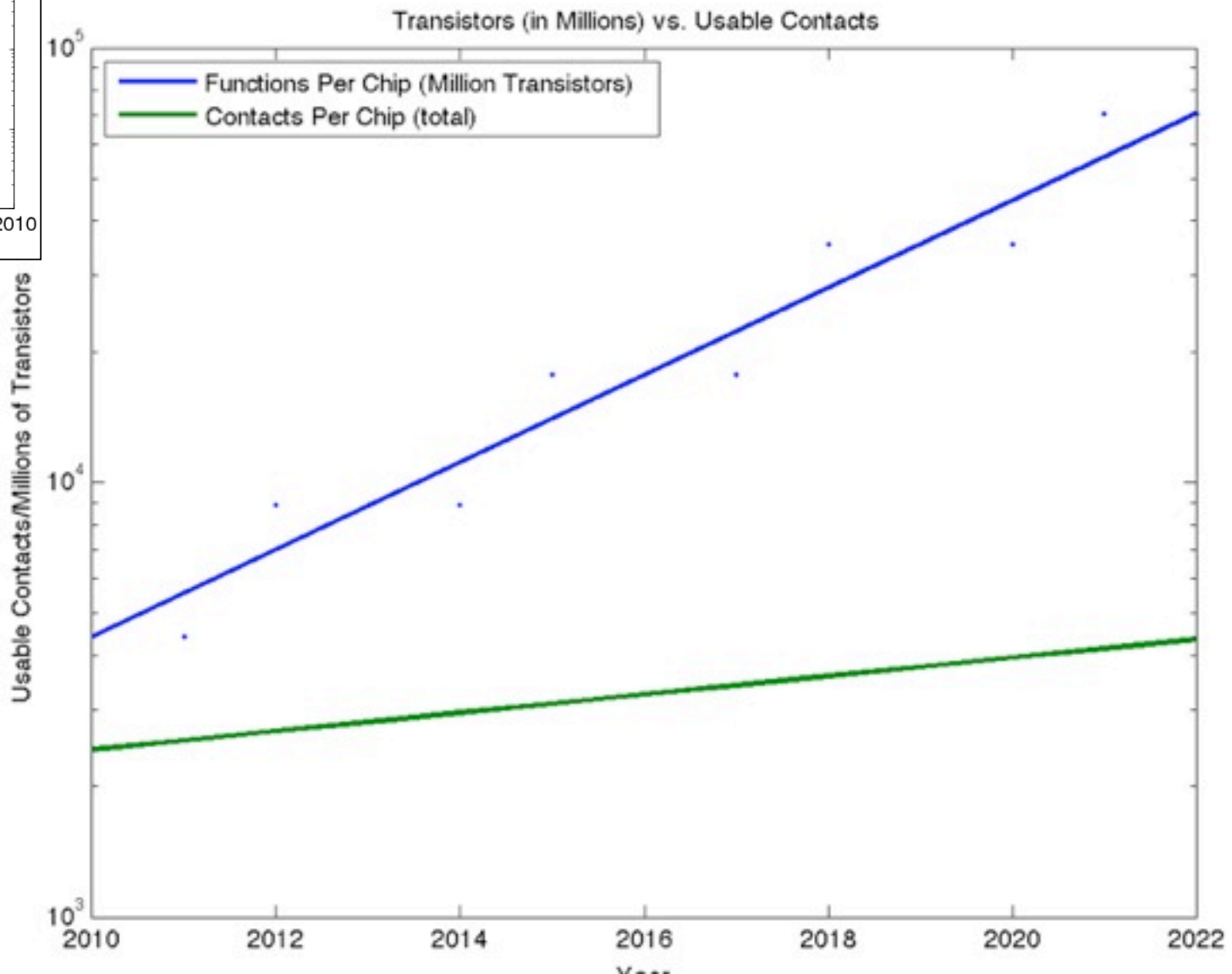
In What Application Context?

- Graph
  - Understand the state of the world
- Stream
  - Pull in and perform preliminary analysis of new sensor data
- Decision Support (Chess)
  - Classic AI community pruning search trees
- Shock Physics (CTH)
  - Understand kinetic impact
- Materials (LAMMPS)
  - Understand materials and interaction

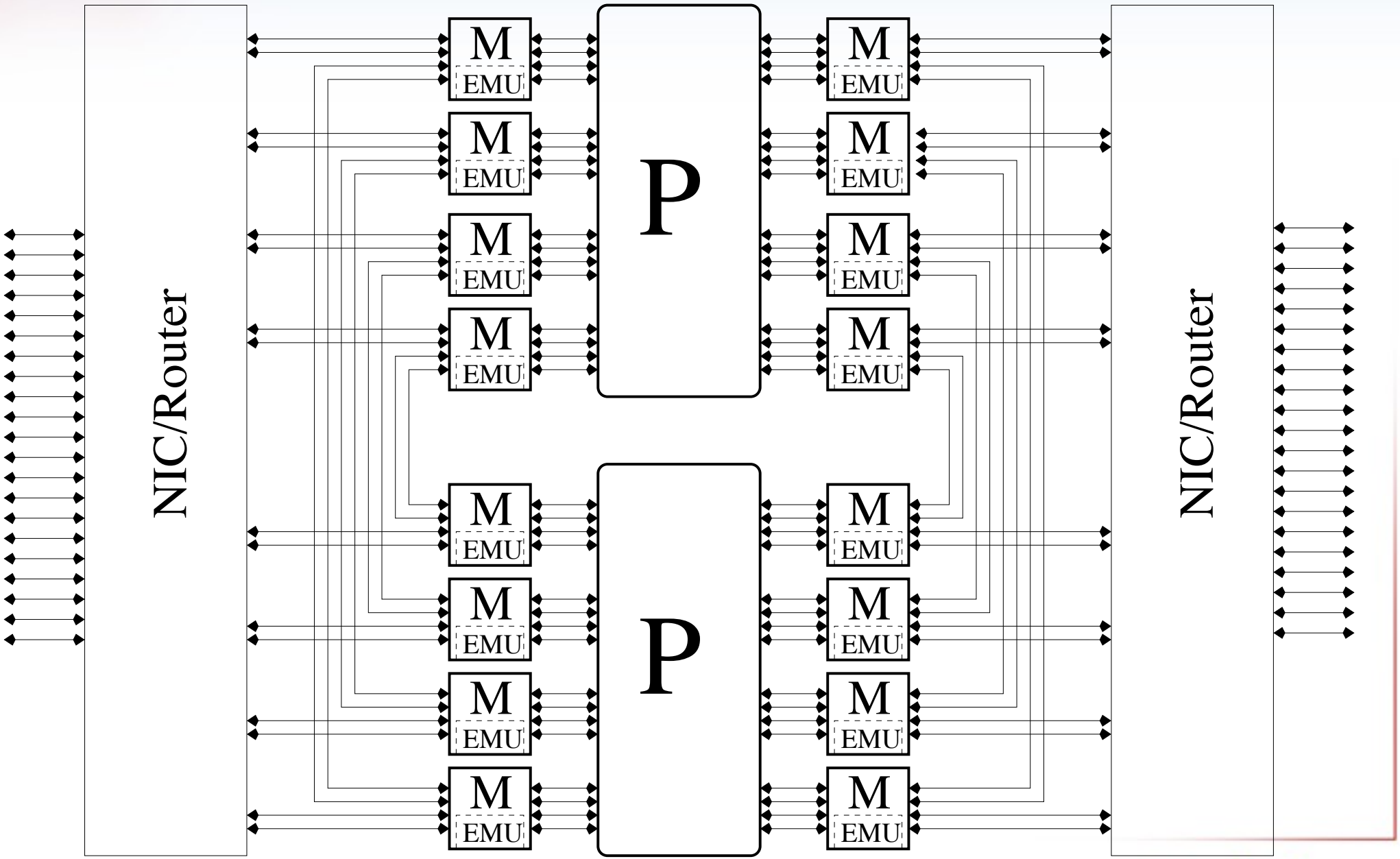
Our Focus is Memory...



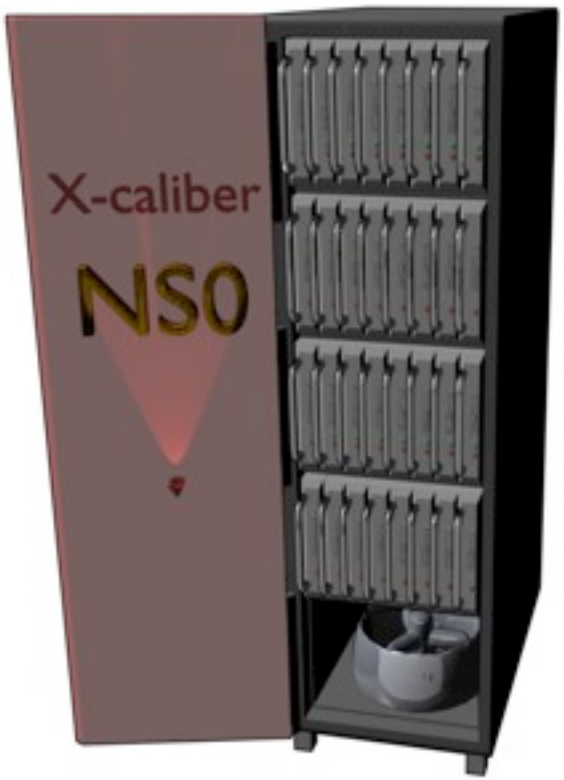
Throughput = Concurrency / Latency



Node Architecture (Continued)



Target Scales



- **Rack Scale**
  - **Processing:128 Nodes, 1 (+) PF/s**
  - **Memory:**
    - **128 TB DRAM**
    - **0.4 PB/s Aggregate Bandwidth**

  - **NV Memory**
    - **1 PB Phase Change Memory (addressable)**
    - **Additional 128 for Redundancy/RAID**
  - **Network**
    - **0.13 PB/sec Injection, 0.06 PB/s Bisection**

Deployment	Nodes	Topology	Compute	Mem BW	Injection BW	Bisection BW
Module	1	N/A	8 TF/s	3 TB/s	1 TB/s	N/A
Deployable Cage	22	All-to-All	176 TF/s	67.5 TB/s	22.5 TB/s	31 TB/s
Rack	128	Flat. Butterfly	1 PF/s	.4 PB/s	0.13 PB/s	0.066 PB/s
Group Cluster	512	Flat. Butterfly	4.1 PF/s	1.6 PB/s	0.52 PB/s	0.26 PB/s
National Resource	128k	Hier. All-to-All	1 EF/s	0.4 EB/s	0.13 EB/s	16.8 PB/s
Max Configuration	2048k	Hier. All-to-All	16 EF/s	6.4 EB/s	2.1 EB/s	0.26 EB/s

Graph500

- Announced at ISC'10, first list at SC10
- Key Kernel: Concurrent Search, growing to three
  - Search: Concurrent Graph Traversal
  - Optimization: Single Source Shortest Path
  - Edge-Oriented: Maximal Independent Set
- Five "business area" data sets
  - Cybersecurity
  - Medical Informatics
  - Data Enrichment
  - Social Networks
  - Symbolic Applications
- See: www.graph500.org with a 10/1 benchmark release
- International, Multidisciplinary Steering Committee
  - Jim Ang, David Bader, Brian Barrett, Jon Berry, Bill Brantley, Almadena Chtchelkanova, John Daly, John Feo, Michael Garland, John Gilbert,Bill Gropp, Bill Harrod, Bruce Hendrickson, Jure Leskovec, Bob Lucas, Andrew Lumsdaine, Mike Merrill, Hans Meuer, David Mizell, Shoaib Mufti, Richard Murphy, Nick Nystrom, Fabrizio Petrini, Wilf Pinfold, Steve Poole, Arun Rodrigues, Rob Schreiber, John Simmons, Marc Snir, Thomas Sterling, Blair Sullivan, T.C. Tuan, Jeff Vetter, Mike Vildibill