Opportunities Beyond Single-Core Microprocessors

HPCA/PPOPP 2009 Panel

Mark D. Hill	U. of Wisconsin
Sarita V. Adve	U. of Illinois
David A. Bader	Georgia Tech
William Dally	Nvidia & Stanford
Vivek Sarkar	Rice

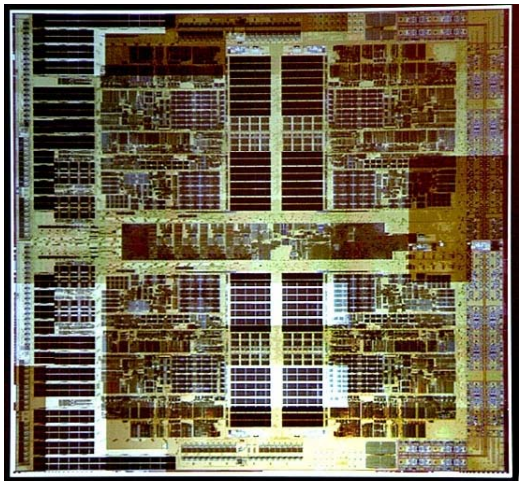
20th Century Computer Architecture Research Achievements (per David Patterson)

- ✖ Invent Stored Program Computer
- ✖ Built 1st Digital Computers & Computer Companies
  - ENIAC, EDSAC, UNIVAC, …
  - Eckert-Mauchy, …
- ✖ Invent Memory Hierarchies
  - Virtual Memory, Cache
- ✖ Computer Architecture vs. Implementation
  - Instruction Set Architecture
  - Vacuum Tubes, Magnetic Core, Magnetic Disks, VLSI
  - Cost performance improved 10^{11} in 55 years
- ✖ Instruction Level Parallelism
  - Pipelining, Vector, Branch Prediction, Superscalar, Out-Of-Order Execution, Speculation, …
- ✖ Could Do Similar List for Software (but Hill didn't)

Q: Why This HPCA/PPOPP Panel?

A: Multicore chips here & cores multiplying fast!

4 cores now

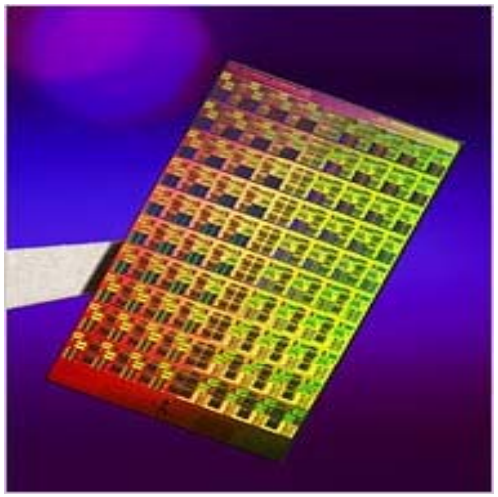


AMD Quad Core

16 cores 2009

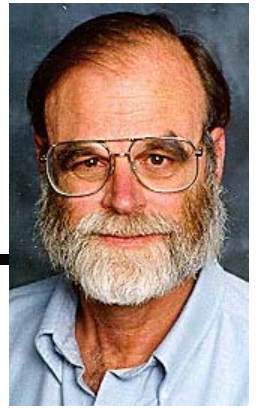
Sun Rock

80 cores in 20??



Intel TeraFLOP

Properties of a Research Goal



- Simple to state
- Not obvious how to do it
- Clear benefit
- Progress and solution are testable
- Can be broken into smaller steps
  - So that you can see intermediate progress

http://research.Microsoft.com/~Gray/talks/Turing2.ppt

Opportunities Beyond Single-Core Microprocessors

HPCA/PPOPP 2009 Panel

Mark D. Hill	U. of Wisconsin
Sarita V. Adve	U. of Illinois
David A. Bader	Georgia Tech
William Dally	Nvidia & Stanford
Vivek Sarkar	Rice

Opportunities Beyond Single-Core Microprocessors:

Towards a Computing Renaissance

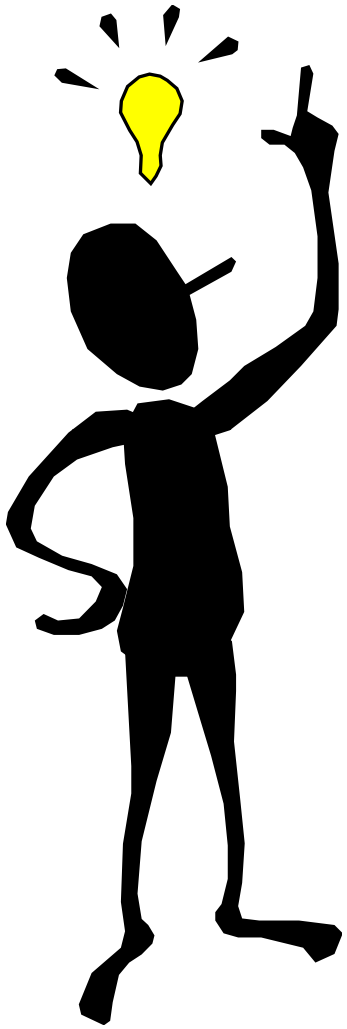
HPCA/PPoPP'09 Panel

Sarita Adve
University of Illinois
sadve@illinois.edu

UPCRC Illinois
Universal Parallel Computing Research Center

CS 101

Java



UPCRC Illinois
Universal Parallel Computing Research Center

CS 101

Threads

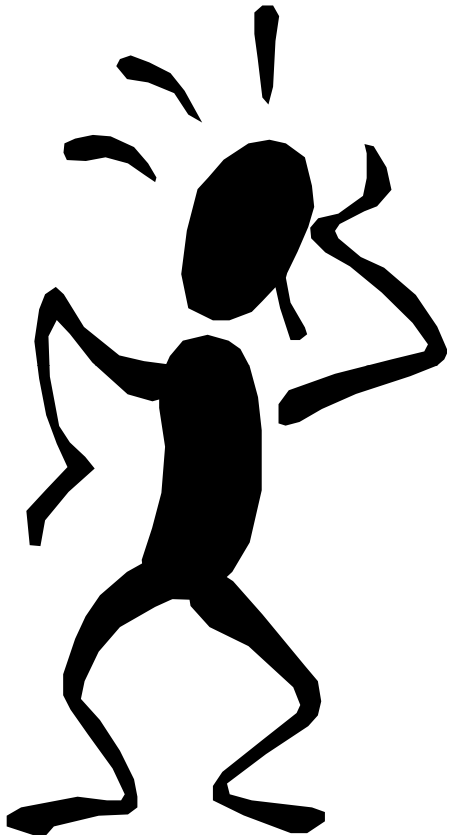
Java

CS 101

Threads

Java

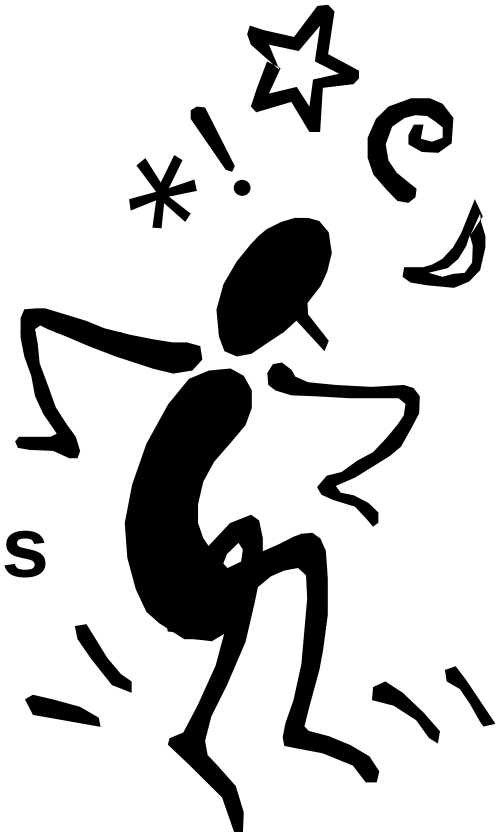
Data races



CS 101

Threads

Java



Data races

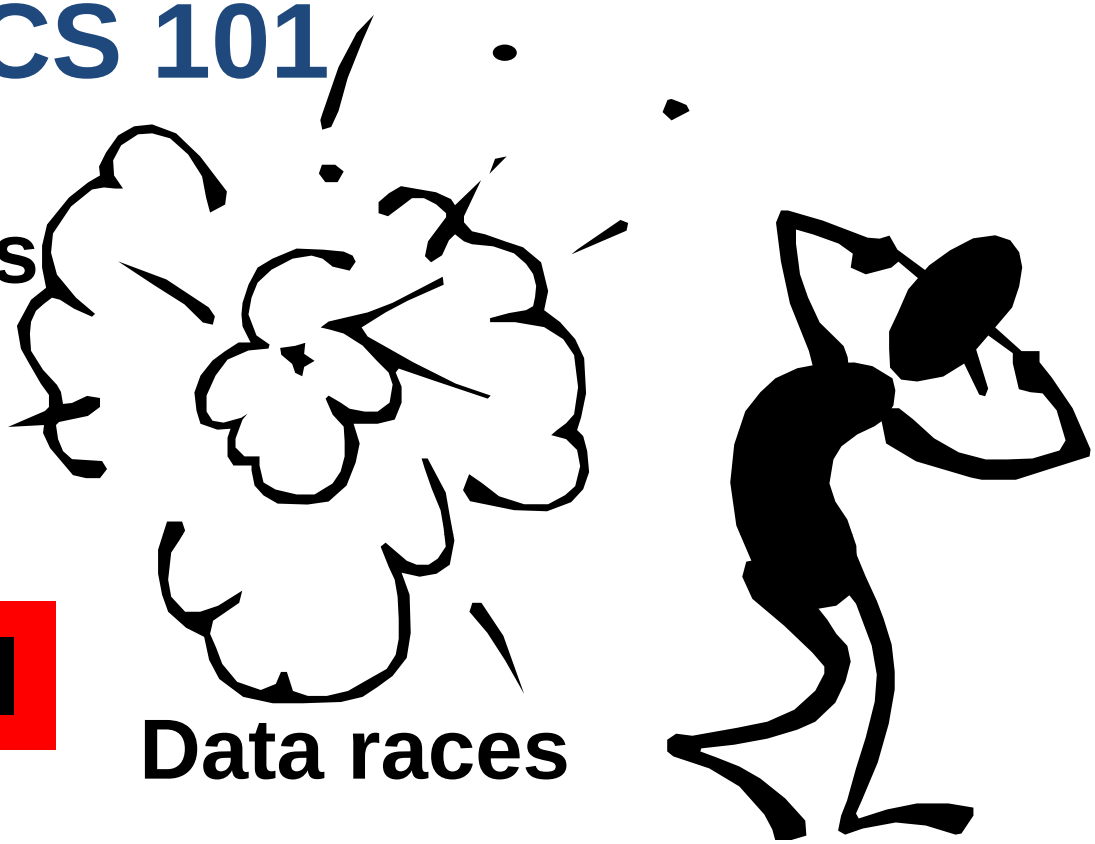
Non-determinism

(just because you got correct output for your code, I don't have to)

CS 101

Threads

Java



Memory Model

Data races

Non-determinism

(just because you got correct output for your code, I don't have to)

Software Research

Current parallel languages are fundamentally broken

Software Research

Theorem: Current parallel languages are fundamentally broken

Proof: See the Java Memory Model

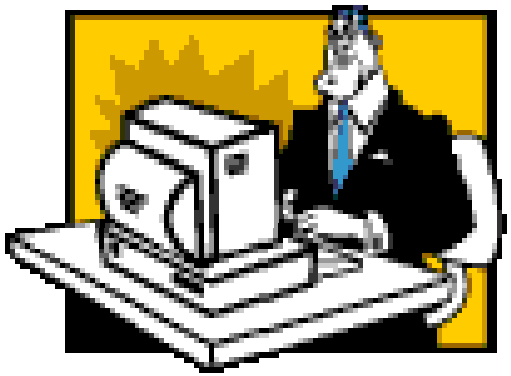
Need safe parallel languages that

- Banish data races
- Provide determinism by default
- Support only explicit and controlled non-determinism

[Full-disclosure: Illinois working on such a language]

What About Architecture?

- How to design 1000 core system for
  - High performance, scalability
  - Low power, energy, temperature
  - High reliability



- For what software?

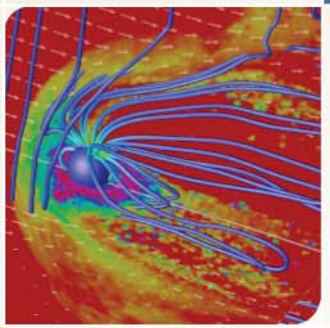
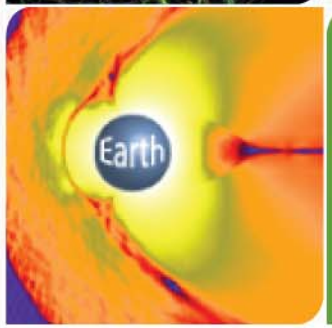
Towards a Computing Renaissance

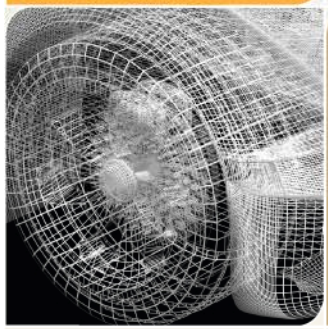
For architects:
Dream opportunity to influence software stack

For software:
Time to ask architects for help

Once-in-a-lifetime chance to rethink the system, together

UPCRC Illinois
Universal Parallel Computing Research Center

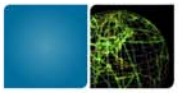





Exascale Analytics: Real-World Challenges

David A. Bader

Georgia Tech College of Computing
Computational Science and Engineering



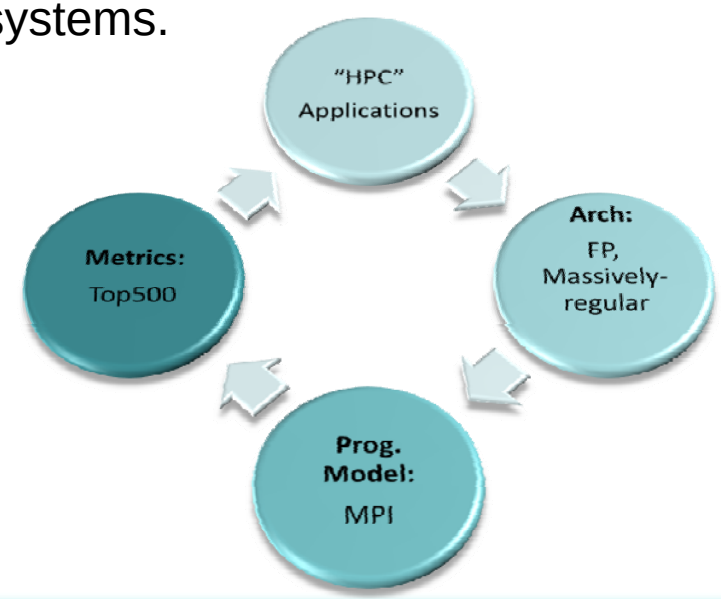
Exascale Analytics: Real-world challenges

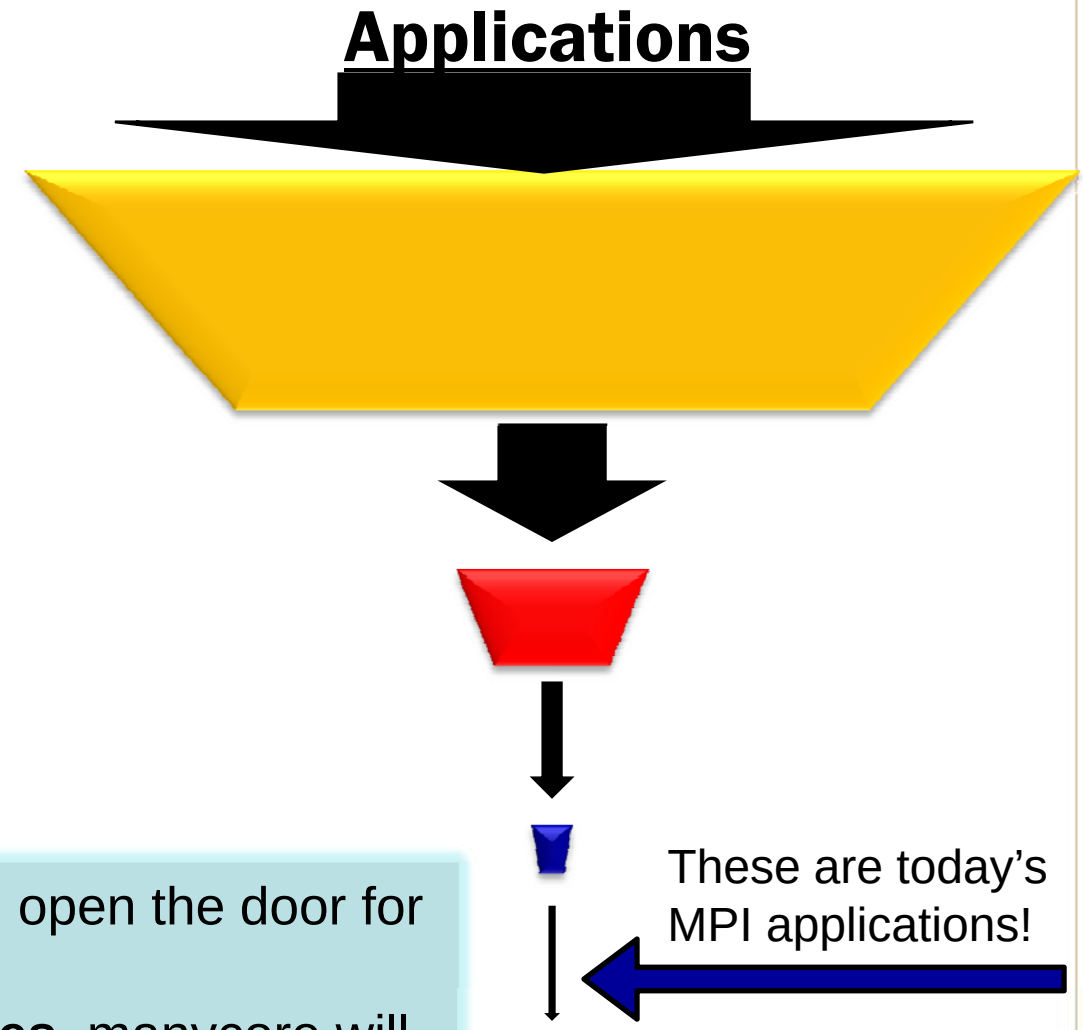
- Health care → disease spread, detection and prevention of epidemics/pandemics (e.g. Avian flu)
- Massive social networks → energy conservation requires social change, modeling pandemic spread, transportation and evacuation
- Intelligence → business analytics, anomaly detection, security, knowledge discovery from massive data sets
- Systems Biology → understanding complex life systems, drug design, microbial research, unravel the mysteries of the HIV virus; understand life, disease, and evolution
- Electric Power Grid → communication, transportation, energy, water, food supply
- Modeling and Simulation → Perform full-scale economio-socio-politico simulations

All require dynamic spatio-temporal interaction networks and graphs

HPC's success has been its failure!

For decades, HPC has been on a vicious cycle of enabling applications that run well on HPC systems.

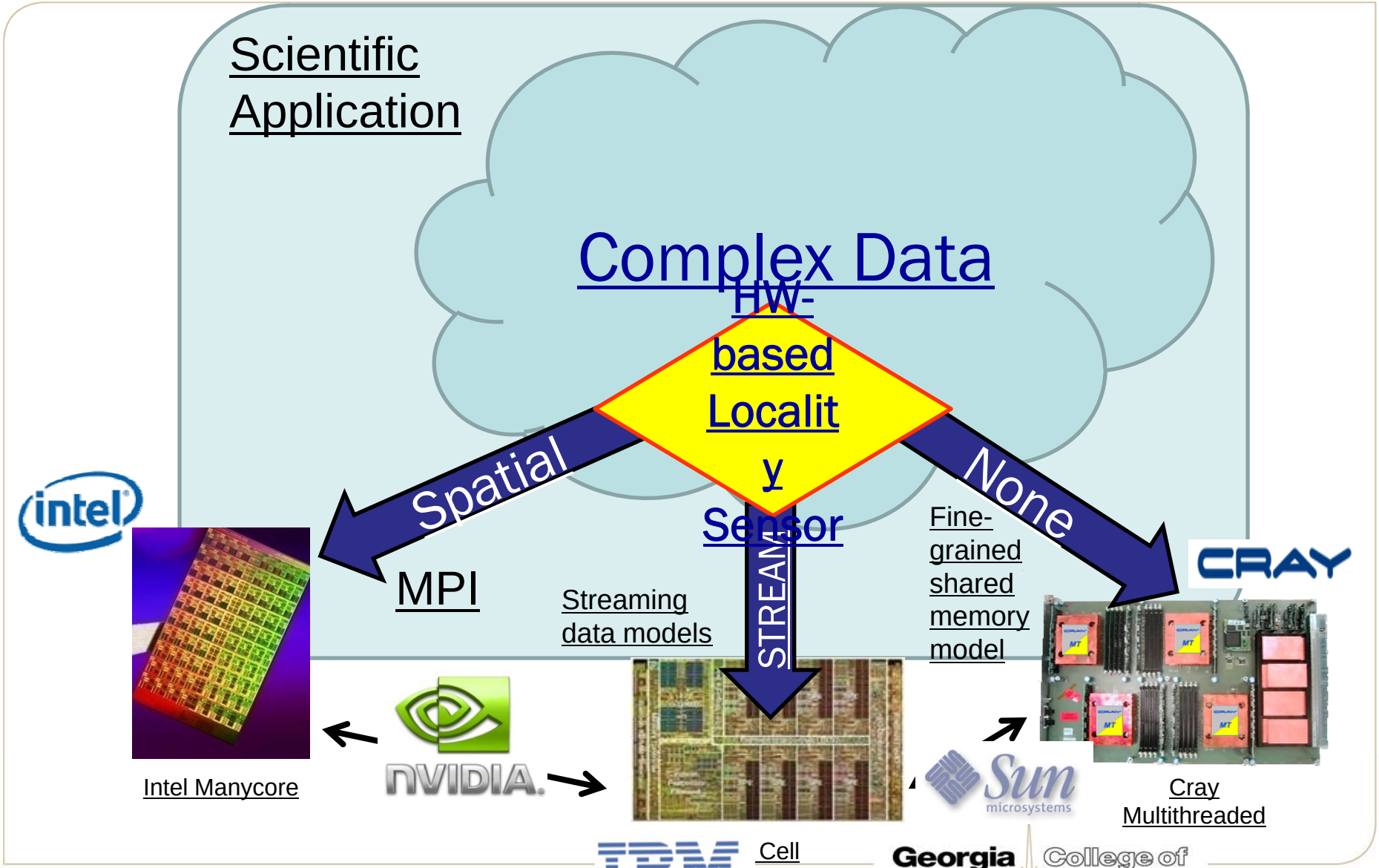




➔The manycore disruption can open the door for a revolutionary transformation!

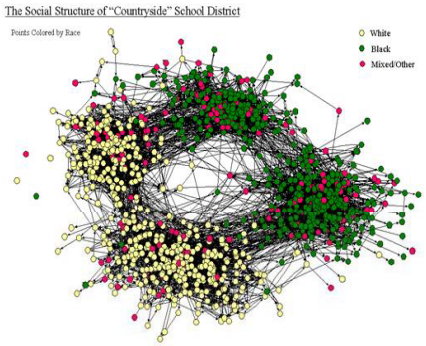
- For the first time in **decades**, manycore will allow innovation in real-world algorithms

Designing A Data-Centric Exascale System



Center for Adaptive Supercomputing Software (CASS-MT)

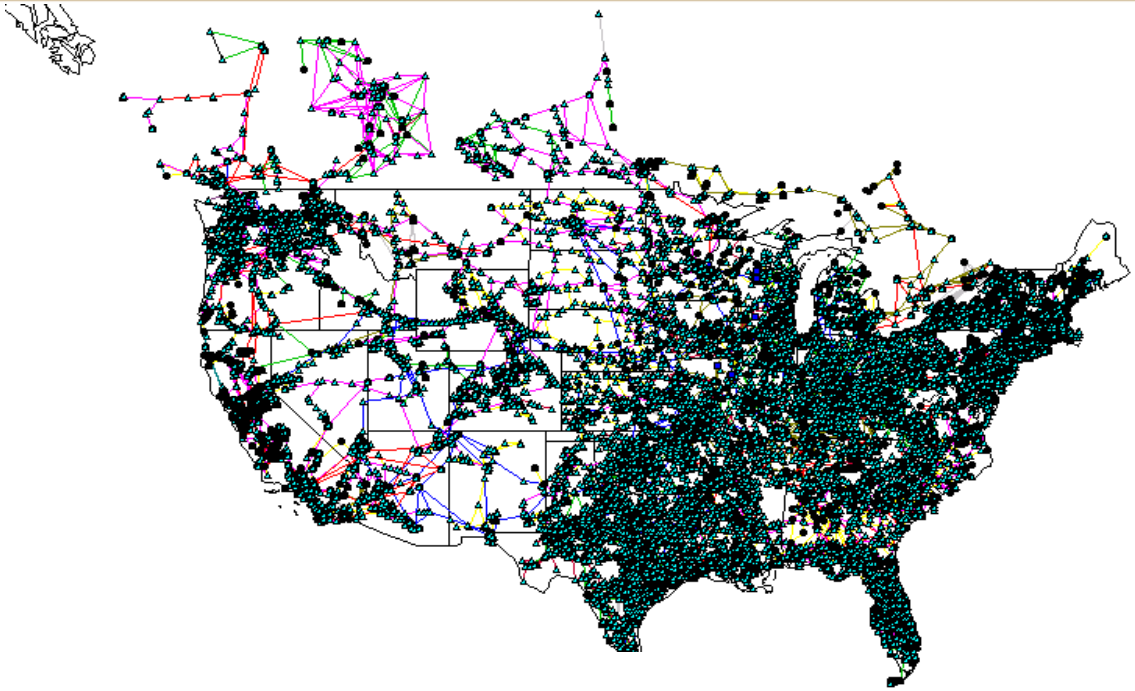
- CASS-MT, launched July 2008
- Pacific-Northwest Lab
  - Georgia Tech, Sandia, WA State, Delaware
- The newest breed of supercomputers have hardware set up not just for speed, but also to better tackle large networks of seemingly random data. And now, a multi-institutional group of researchers has been awarded $4.0 million to develop software for these supercomputers. Applications include anywhere complex webs of information can be found: from internet security and power grid stability to complex biological networks.



CRAY

Georgia Tech College of Computing

US High Voltage Transmission Grid (>150,000 miles of line)



The New York Times
Thursday, September 4, 2008

Report on Blackout Is Said To Describe Failure to React

By MATTHEW L. WALD
Published: November 12, 2003

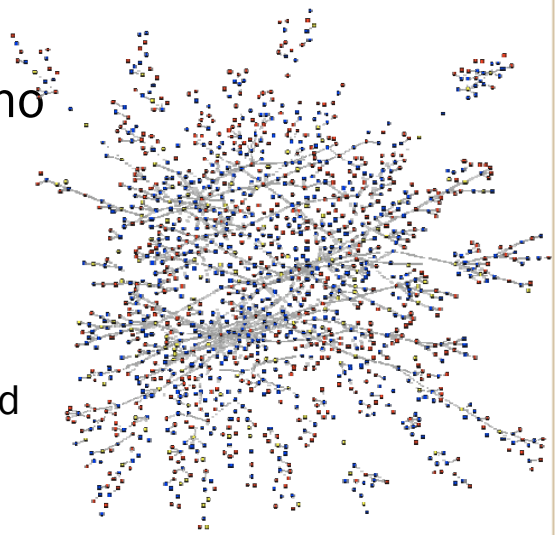
A report on the Aug. 14 blackout identifies specific lapses by various parties, including FirstEnergy's failure to react properly to the loss of a transmission line, people who have seen drafts of it say.

A working group of experts from eight states and Canada will meet in private on Wednesday to evaluate the report, people involved in the

E-MAIL
PRINT
SINGLE-PAGE
REPRINTS
SAVE
SHARE

Social Networks

- Facebook has more than 150 million active users who have returned to the site in the last 30 days (as of January 2009)
- Example application: Malcolm Gladwell, in *The Tipping Point*, identifies three personality types that play central roles in epidemic/viral spread: Connectors, Mavens, and Salespeople. We can identify, for example, Connectors who are people who bridge between social communities.

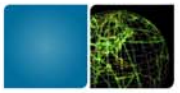


- Traditional graph partitioning often fails:
  - **Topology**: Interaction graph is low-diameter, and has no good separators
  - **Irregularity**: Communities are not uniform in size
  - **Overlap**: individuals are members of one or more communities

Sample Queries in a massive, dynamic graph

- Allegiance switching: identify entities that switch communities.
- Community structure: identify the genesis and dissipation of communities
- Phase change: identify significant change in the network structure

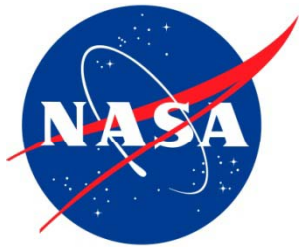
Suddenly, the flock became suspicious:
How come the newcomer wasn't shorn?



Additional Open Questions

- Are there new analytics for massive spatio-temporal interaction networks and graphs (STING)?
- Do current methods scale up from thousands to millions and billions?
- How do I visualize a STING with O(1M) entities? O(1B)? O(100B)? with scale-free power law distribution of vertex degrees and diameter =6 …
- Which algorithms are efficient using map/reduce on a cloud rather than special-purpose parallel architectures (e.g. Cray XMT, Sun Rock, etc.)
- Can accelerators (e.g. Cell and GPUs) aid in processing streaming graph data?
- Can we influence commodity architectures (e.g. x86) to improve capabilities for STING?
- Will hardware transactional memory be useful?

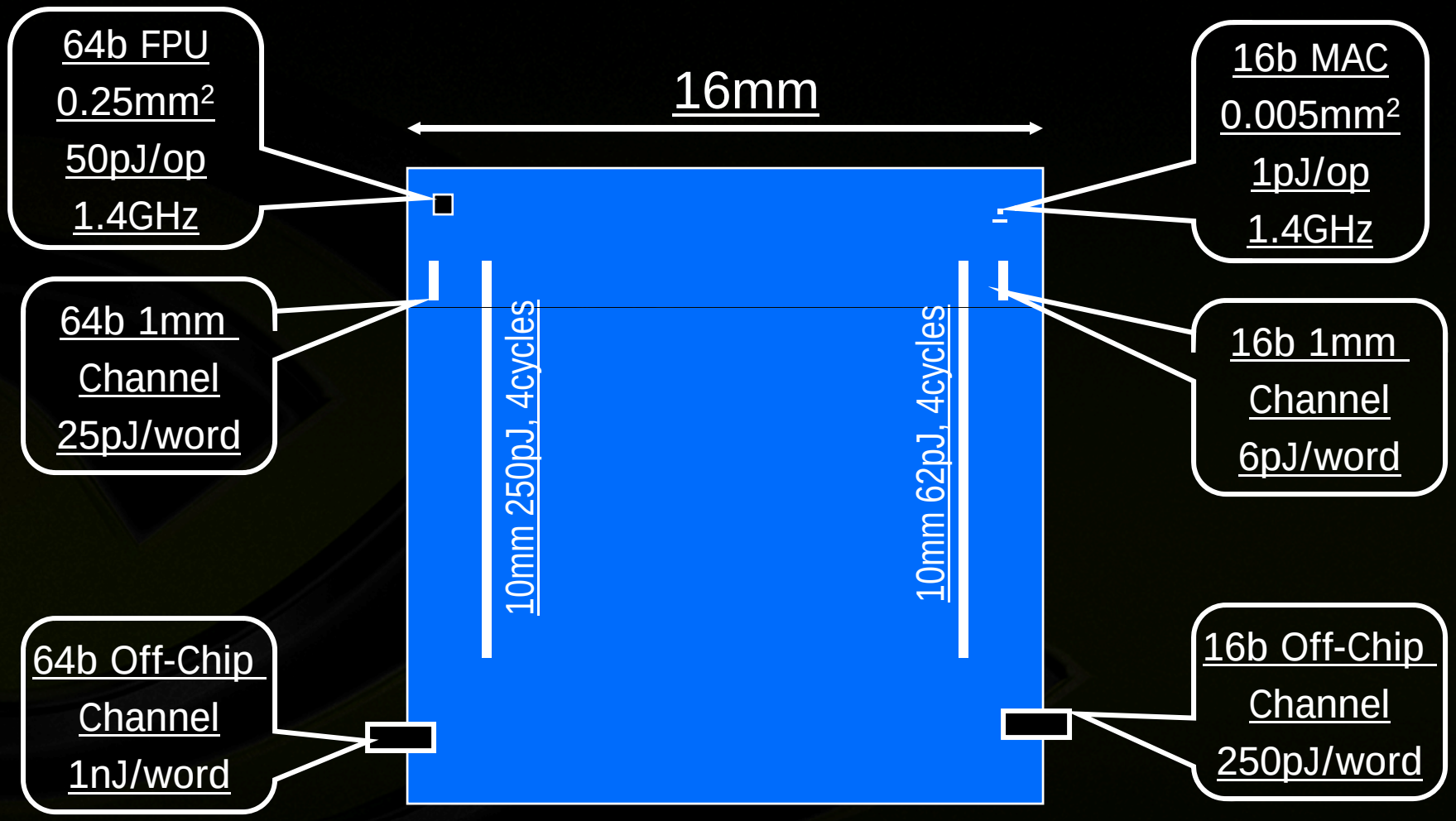
Acknowledgment of Support



Computer Architecture Research in the 2010s

Bill Dally
Chief Scientist & VP Research
NVIDIA Corporation
Bell Professor of Engineering, Stanford

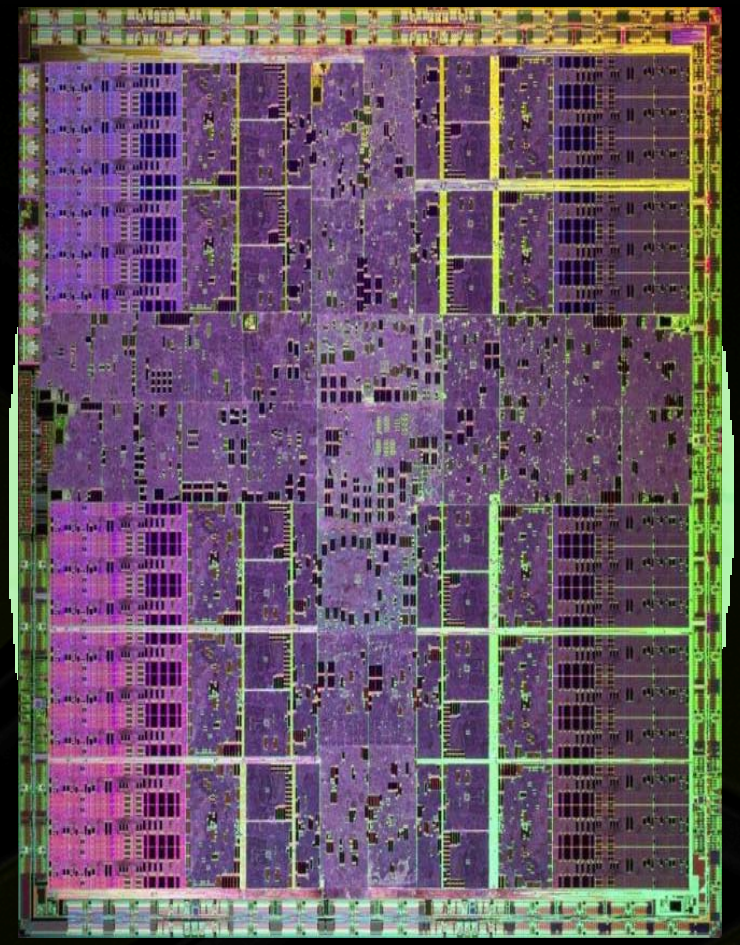
The “canvas” for today’s Architects



In Today's Many-Core World

- Performance = Parallelism
- Efficiency = Locality

240 Cores Today, 1000s Soon

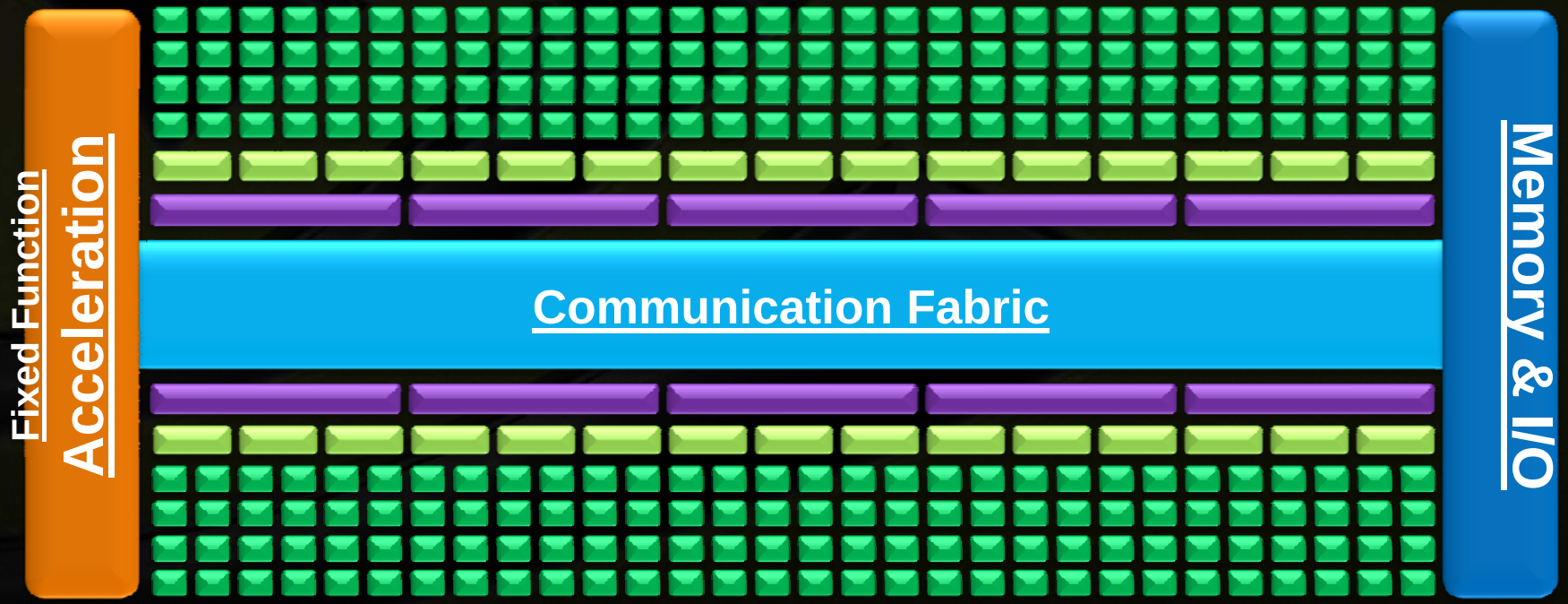


NVIDIA GT200 GPU

240 32b/64b FP Cores

Research Goals

- Make this efficient
  - High fraction of area & energy into useful work
- Make this programmable
  - Easy to get parallel programs working
  - Easy to get high fraction of peak performance



Some Specific Topics

- Efficient communication and synchronization mechanisms
  - In support of productive parallel programming systems
- On-chip interconnection networks
- Memory/Communication Hierarchy
  - Organization and programming

Desiderata

- High Impact
  - If you are successful it should make a BIG difference (>2x)
  - 20% improvements are not interesting
- Solve a real problem
  - With a clear transition path to impact a product
- Don’t be constrained
  - By programming system, benchmarks, etc…
  - Real progress usually involves changing ALL the variables
- Long term
  - Academic research should be looking 5+ years out
  - Industry is far better at tuning the next few generations of products
  - New ideas are far more important than polished results

Research Opportunities for Many-Core Microprocessors

Vivek Sarkar, Rice University

Three areas:

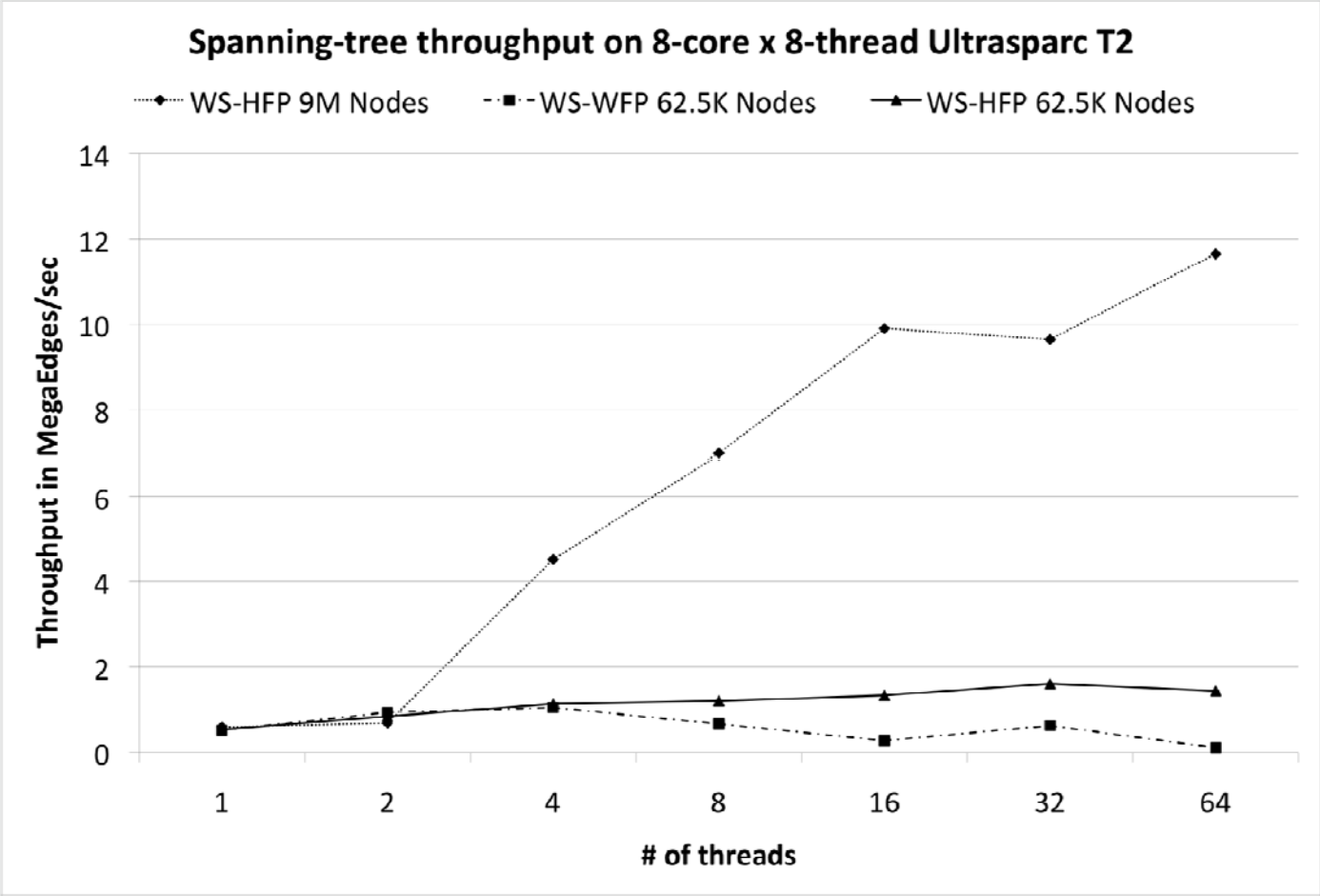
1. Scalable Concurrency
2. Data Affinity and Ownership
3. Software-Hardware Interfaces

Focus of this presentation is on software-hardware co-design challenges at the microprocessor level

1. The Scalable Concurrency Opportunity

- **Challenge:** Support parallel programs with $O(10^5)$ pending computational tasks on a single microprocessor, with 100x increase in ops/byte compared to current sequential programs
- **Benefit:** Forward portability of software by leaping directly into manycore era instead of a following a long incremental path of 2x increases in degree of parallelism

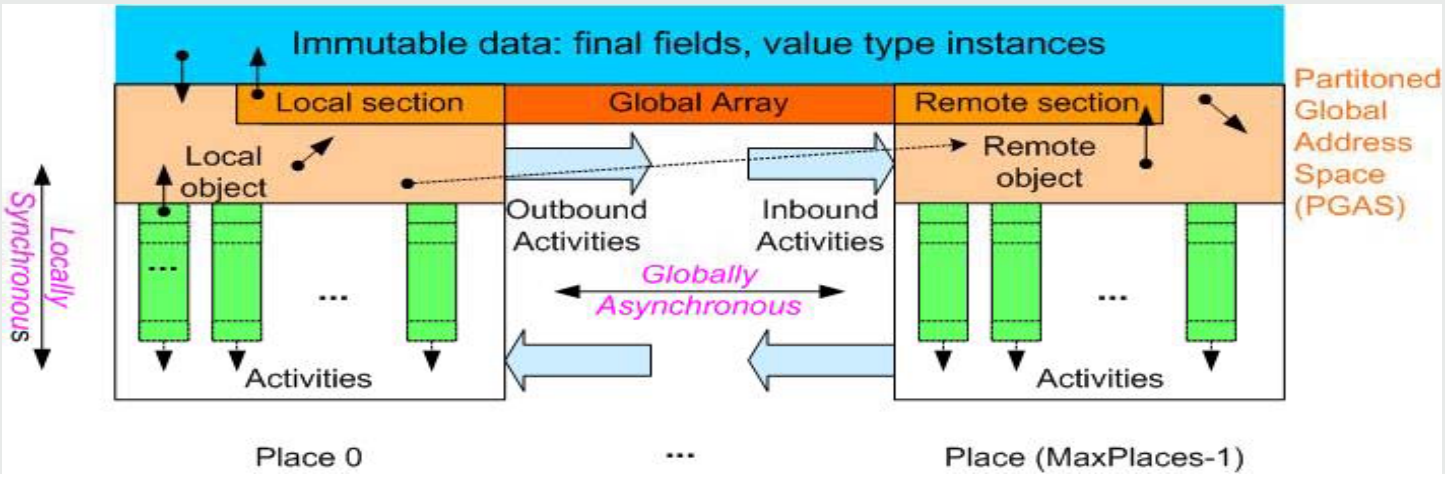
Example: Stack Size Limitation in Spanning Tree Example using Help-First and Work-First Scheduling Policies



"Work-First and Help-First Scheduling Policies for Async-Finish Task Parallelism" Y.Guo, R.Barik, R.Raman, V.Sarkar. To appear in IPDPS 2009.

2. The Data Affinity and Ownership Opportunity

- **Challenge:** Co-align task and data mappings so as to minimize the number of "remote" data accesses; also support checking of ownership violations with $O(10^5)$ tasks
- **Benefit:** orders-of-magnitude improvements in data access latencies and energy-intensive data movement operations, with software-friendly system support



Storage classes:
Activity-local
Place-local
Partitioned global
Immutable

3. The Software-Hardware Interface Opportunity

- **Challenge:** enrich software-hardware interface to include
  - Memory access patterns for different sets of locations (read-only, sw/hw coherence, cache bypass, …)
  - Interconnect bandwidth partitioning for different data movement operations (shared memory, DMA, messaging, streaming, …)
  - Lightweight profiling, Power management, …
- **Benefit:** orders-of-magnitude performance and efficiency improvements

Opportunities Beyond Single-Core Microprocessors

HPCA/PPOPP 2009 Panel

Mark D. Hill	U. of Wisconsin
Sarita V. Adve	U. of Illinois
David A. Bader	Georgia Tech
William Dally	Nvidia & Stanford
Vivek Sarkar	Rice