X-factors: The X-caliber Approach to Codesign, Data Movement, and Pico-Joules

Richard C. Murphy

X-caliber Project PI

Scalable Computer Architectures Department

Sandia National Laboratories

Affiliated Faculty, New Mexico State University

October 27, 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.

Who Are We?

Richard Murphy, Jim Ang, Ron Brightwell, Peter Kogge, Scott Hemmert, Bob Lucas, Kunle Olukotun, Marc Snir, Thomas Sterling, David Bayliss, Jeff Draper, Gary Bernstein, Jacque Chame, Pedro Diniz, Bill Gropp, Vince Freeh, Bruce Jacob, Tony Lentine, Supratik Mukhopadhyay, Ram Ramnujam, Dave Resnick, Arun Rodrigues, Subhash Shinde, Sudha Yalamanchili

Team Formed 2 Years Ago

Sandia National Laboratories

Micron

LexisNexis

Leland Stanford Junior University 1891

GT

LSU

NC State

Notre Dame

ISI Information Sciences Institute USC Viterbi School of Engineering

University of Maryland

NNSA

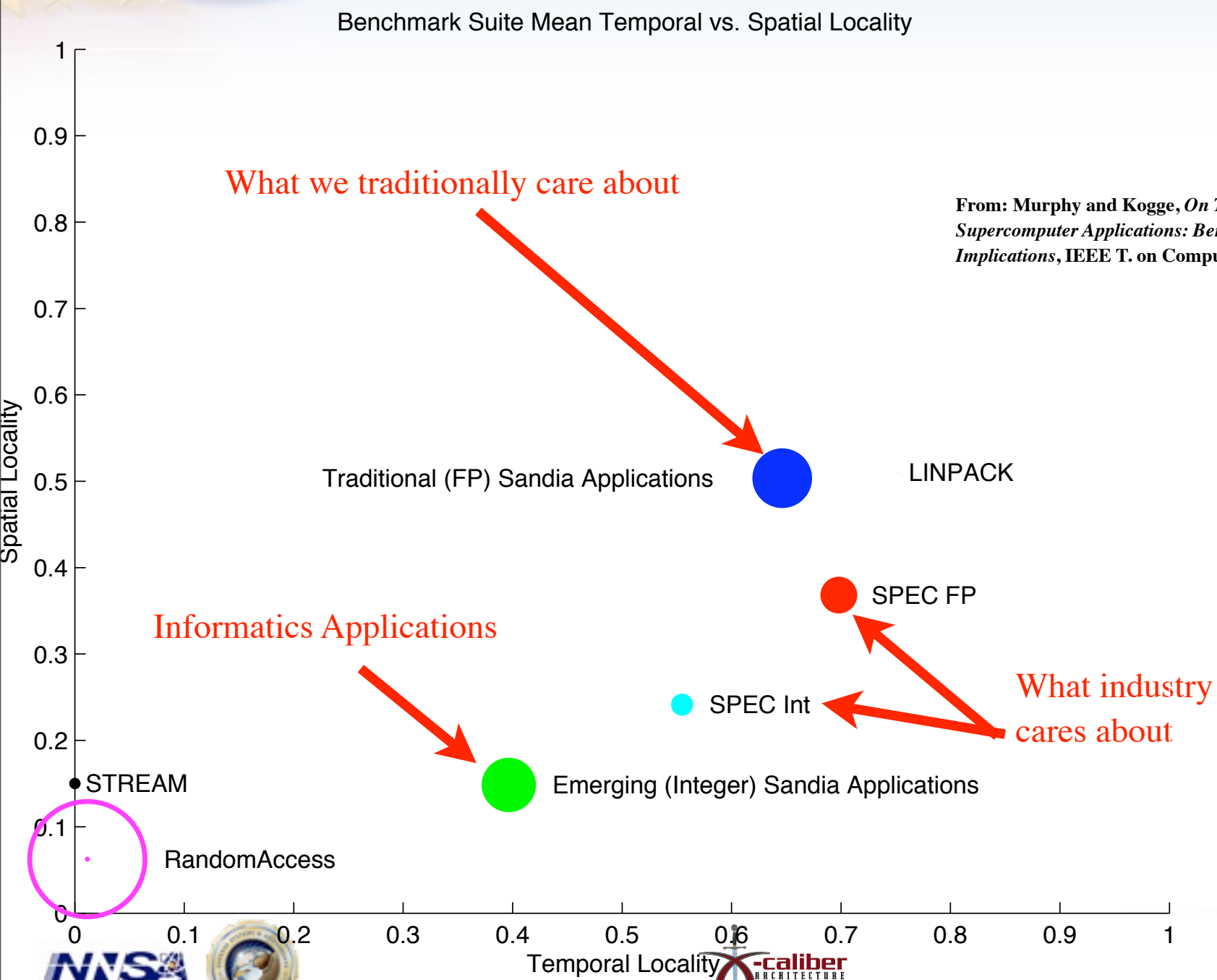
X-caliber Architecture

Sandia National Laboratories

What does the Exascale Era Look Like?

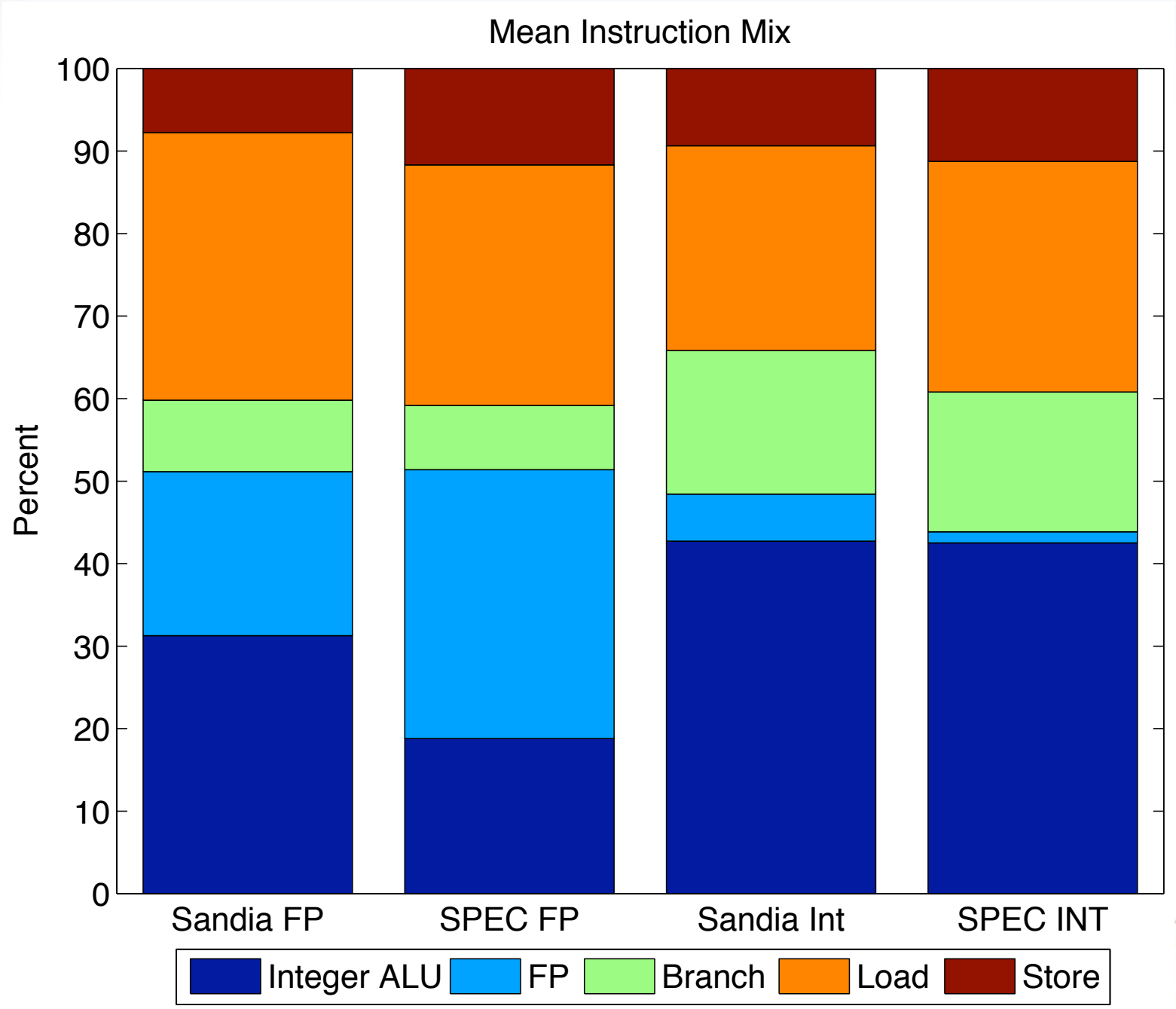
- The Technology Facts
  - Without intervention conventional architectures and technology will produce 125 MW (+) exascale machines
  - We can't cool aggressively clocked processors, so increased parallelism is required to achieve performance
  - Data movement, not computation dominates both energy and performance costs
- Application Trends
  - Science codes will be increasingly unstructured
  - (My guess - and hope - is) new informatics and knowledge applications will be a huge fraction of supercomputer workloads in both national security and open science
- Results for the architect
  - Today's petascale design targets represent the past, not the future
  - We have a limited opportunity to affect programming and execution models

How are applications changing?

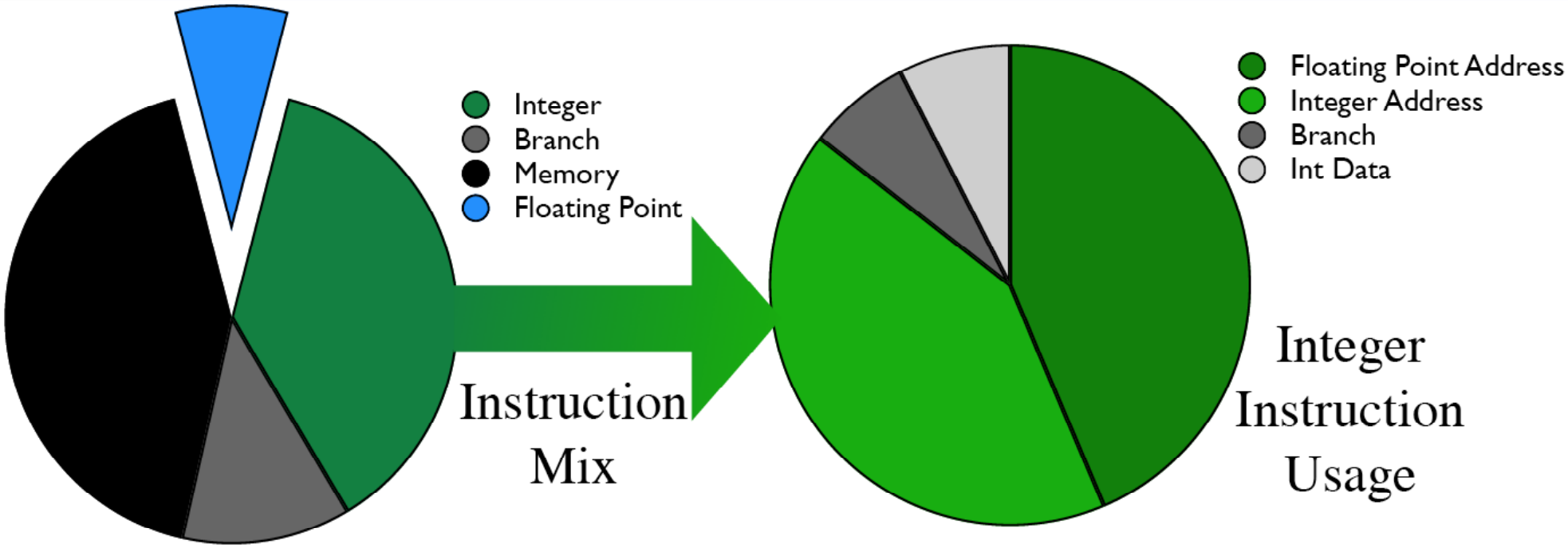


From: Murphy and Kogge, *On The Memory Access Patterns of Supercomputer Applications: Benchmark Selection and Its Implications*, IEEE T. on Computers, July 2007

Most Real Applications Do Memory Accesses, Not Floating Point

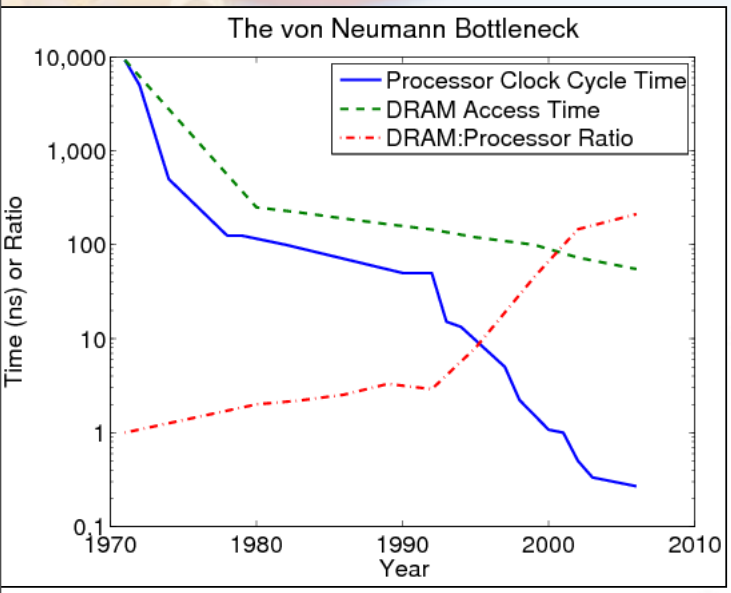


What about DOE "Physics" Applications?

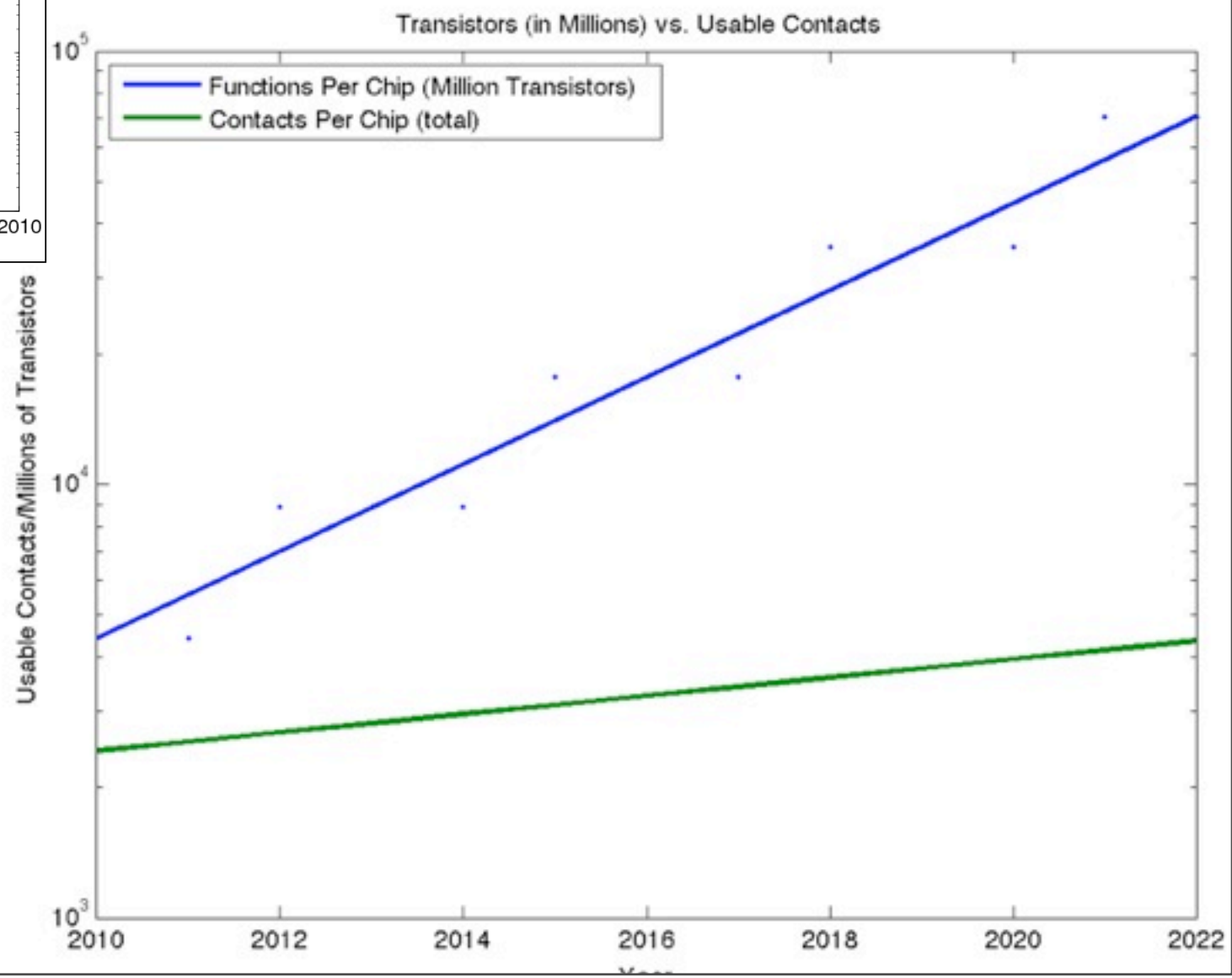


Real Physics Applications Primarily Do SLOW Memory References

The New Memory Wall



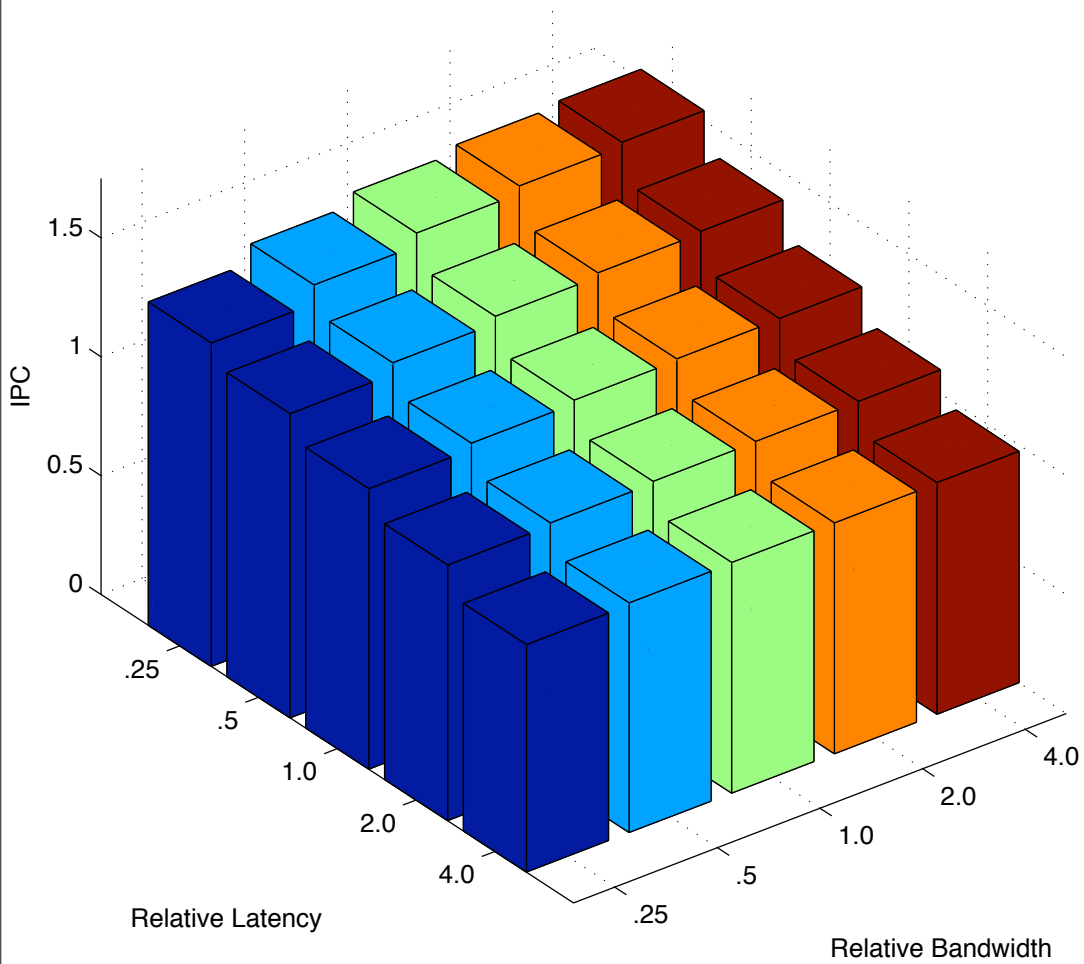
Throughput = $\frac{\text{Concurrency}}{\text{Latency}}$



Latency Dominates Bandwidth (Concurrency Decreases Effective Latency)

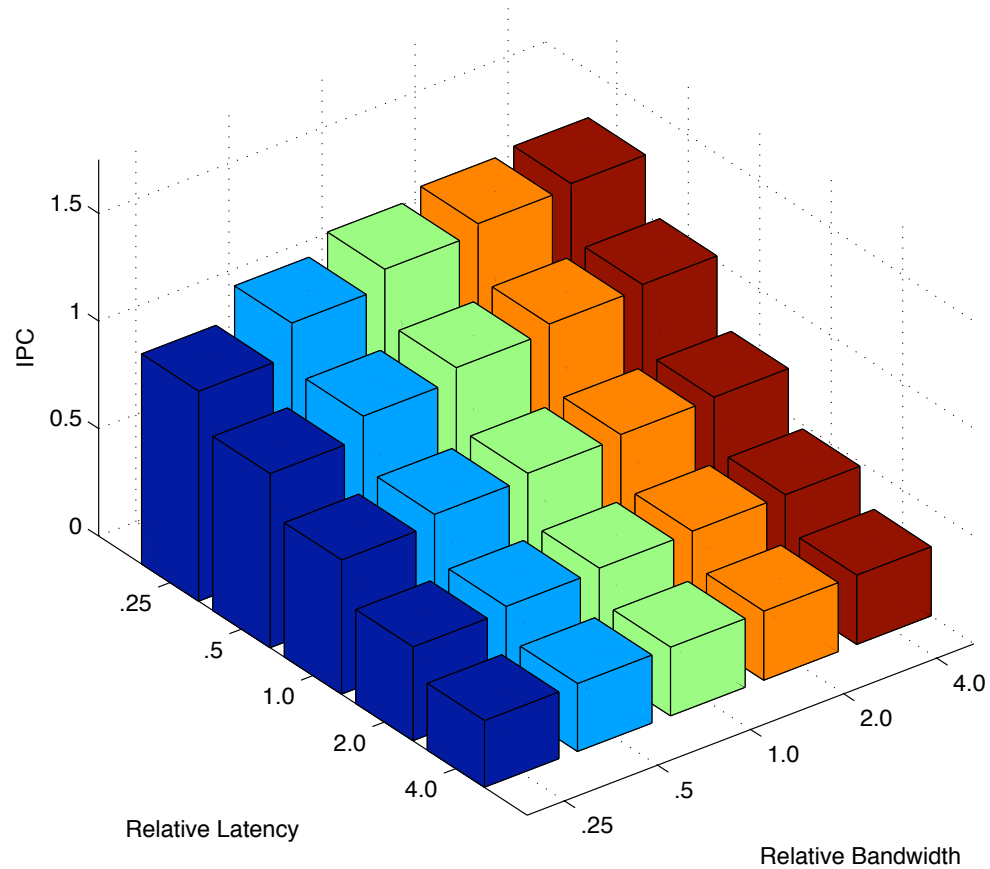
Physics

Average Sandia FP Latency and Bandwidth vs. Performance



Informatics

Average Sandia Int Latency and Bandwidth vs. Performance

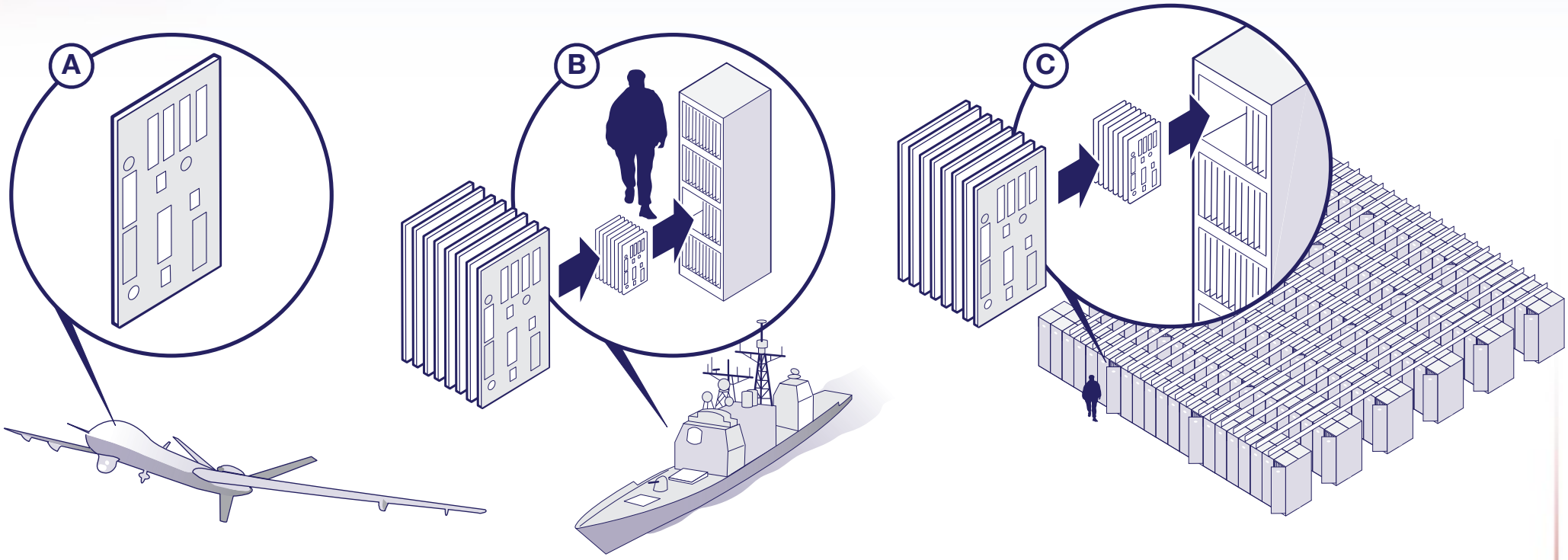


From: Murphy, *On the Effects of Latency and Bandwidth on Supercomputer Application Performance*, in the Proceedings of the IEEE International Symposium on Workload Characterizattion 2007 (IISWC07), Boston, MA September 27-29, 2007.

Goal: Reinvent Computing

- We (particularly DOE) are stuck in highly optimized model of computation -- but is it a local minima?
  - The execution model (MPI, mostly BSP) matches the architecture (MPP) which matches the applications (3D Physics)
- 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

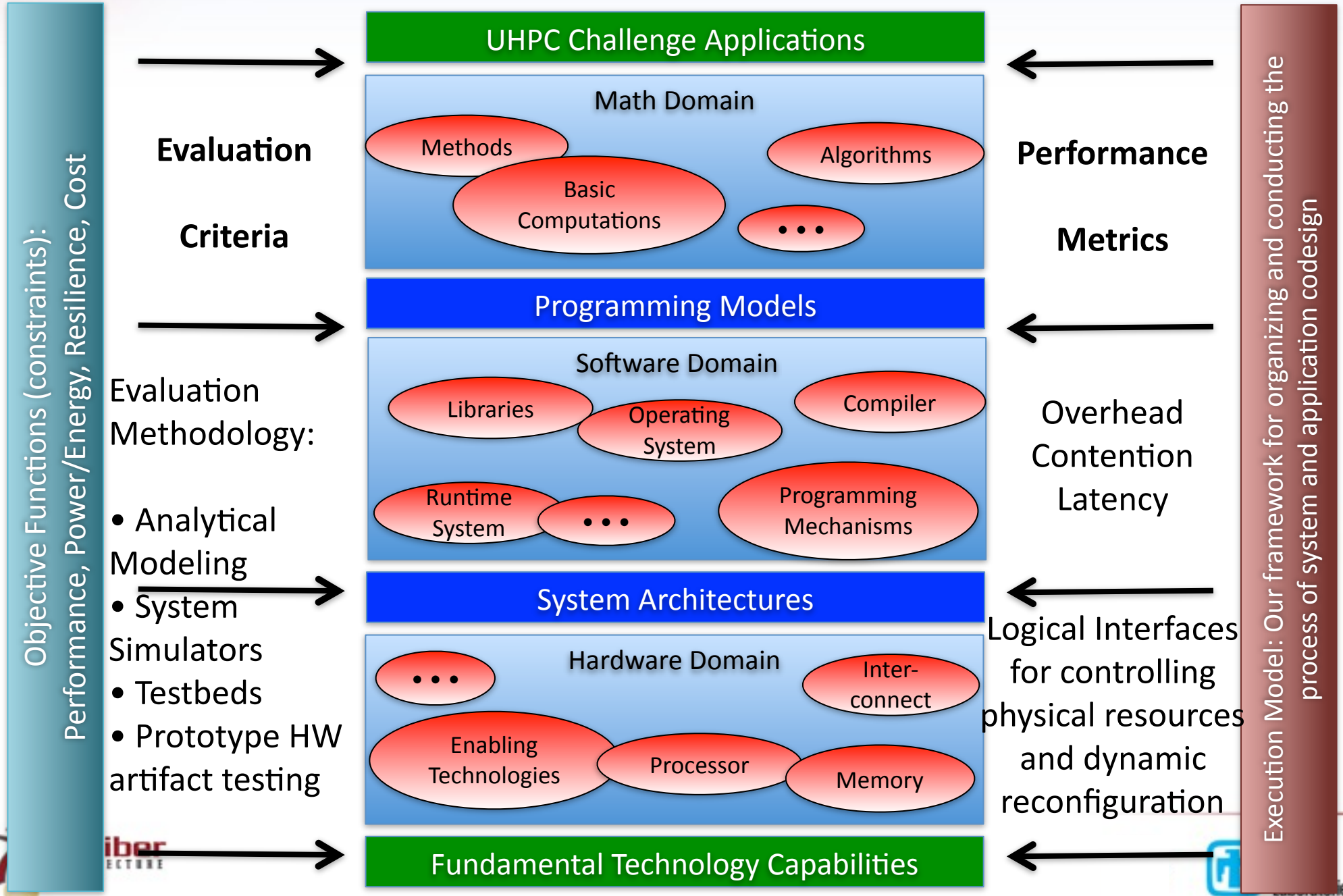
Scale and Scalability



We want to deploy the **same** hardware, execution model, system software, and ultimately applications across all of these scales!

Each exists in totally different operational environments today.

What is codesign?



How does the X-caliber team think about codesign?

- **Model of Computation (AKA: Execution Model)**
  - Enables discussion of the <u>semantics</u> of a machine separate from the <u>implementation... why?</u>
    - How else do people at different layers communicate new ideas?
    - How else do you optimize between layers?
  - Not the traditional approach of a hardware implementation being thrown (at application developers) over the fence
- **My five elements of an execution model...**
  - Concurrency
  - Coordination
  - Movement
    - of work
    - of data
  - Naming
  - Introspection

ParalleX

Element	Parallex Mechanism
Concurrency	Lightweight Threads/Codelets (lightweight, h/w scheduled, for latency tolerance not throughput!)
Coordination	Lightweight Control Objects (LCOs) for construction of mutexes, futures, producer/consumer interactions, etc.
Movement	Of Work: Parcels (lightweight active messages) Of Data: PGAS and Bulk Transfer
Naming	Global Name Space and Global Address Space
Introspection	Unified publication at all levels via System Knowledge Graph (SKG)

ParalleX vs. Today's Dominant Model

Element	Parallex Mechanism	Stylized Communicating Sequential Processes
Concurrency	Lightweight Threads/ Codelets	MPI Ranks/Processes
Coordination	Lightweight Control Objects (LCOs) (fine-grained)	Bulk Synchronous (or maybe by teams and messages) (coarse-grained)
Movement	of Work: Parcels of Data: PGAS and Bulk	of Work: None of Data: Bulk
Naming	Global Name Space Global Address Space	Coarse, rank/node names
Introspection	System Knowledge Graph (enables dynamic/ adaptive)	Not specified by the model, in practice out-of-bands RAS network

ParalleX vs. PGAS

Element	Parallex Mechanism	Stylized Communicating Sequential Processes
Concurrency	Lightweight Threads/ Codelets	Processes
Coordination	Lightweight Control Objects (LCOs) (fine-grained)	Bulk Synchronous/teams Barriers (coarse-grained)
Movement	of Work: Parcels of Data: PGAS and Bulk	of Work: None of Data: Load/Store
Naming	Global Name Space Global Address Space	Global Address Space
Introspection	System Knowledge Graph (enables dynamic/ adaptive)	Not specified by the model, in practice out-of-bands RAS network

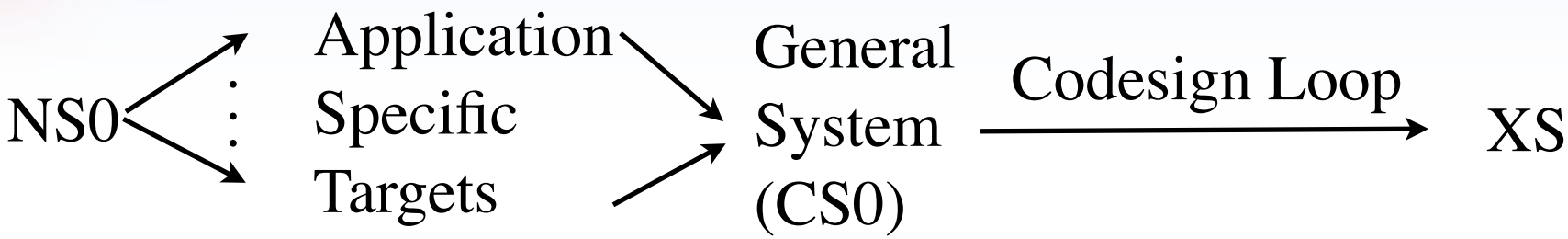
What about GPUs?

Element	Parallex Mechanism	GPUs
Concurrency	Threads/Codelets	SIMD/lock-step threads
Coordination	Lightweight Control Objects (fine-grained)	Local Scratchpad Explicit Coordination
Movement	of Work: Parcels of Data: PGAS and Bulk	of Work: None of Data: Bulk transfer from memory
Naming	Global Name Space Global Address Space	Global Address Space
Introspection	System Knowledge Graph (enables dynamic/ adaptive)	None

The X-caliber Approach

- Two approaches to addressing the "data movement" problem discussed in the UHPC meetings so far:
  - Dominant Approach (all other teams): manage the locality of deep memory hierarchies
  - Our approach: perform the computation as close as possible to the data
    - Relatively shallow (5-layer) memory hierarchy (registers, scratchpad, DRAM, other cubes on a node, other cubes elsewhere)
    - Relaxed consistency model (move the computation, explicitly control data word state)
    - Memory-centric, not processor centric (cubes are homenodes, work initiators, control flow managers, etc. *not processors*)
    - Compute at the closest possible location: bottom of the via stack
- Observation about the dominant approach
  - The best data locality manager today is MPI, and it likely can't solve two of the challenge problems well
  - We know we can't manage the locality of a meaningful graph problem. Period.

Codesign Process



- **Why do it?**
  - **Like embedded systems in the early 1990s we've hit hard constraints (power, energy)... it's why systems are multicore**
- **What is it?**
  - **Directed partitioning of functionality between system layers and choice of implementation under constraints and objective functions**
- **How does it relate to the ParalleX Execution Model?**
  - **Decouples "semantics" and "implementation"**
  - **Serves as a useful early Application Target**
  - **Leave "function" implementation (mechanism) and partitioning (hardware, software, OS, runtime, compiler, etc.) free to be redefined**

System Balance

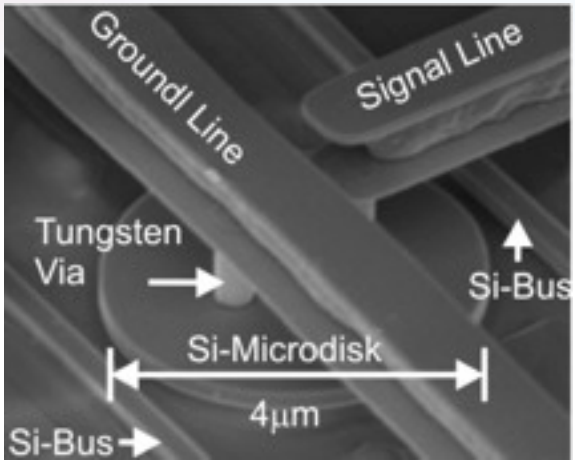
- System Balance
  - Because we're memory centric, we're focused on bandwidth, capacity, and scalability of the memory system (near and far)
  - X-caliber compared to the state of the art (scaled to 2018):
    - 5X the FLOPs of Red Storm
    - 2X the memory capacity
    - Similar network bandwidth ratio
  - Other approaches (aggregate from what I've seen):
    - 10X the FLOPs of Red Storm, Half or less the memory capacity

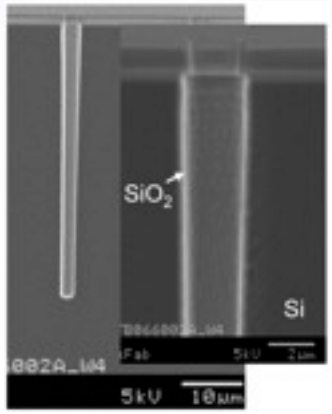
System	Injection BW	FLOPS	B/F	Ratio	Comment
X-caliber	133 TB/s - 266 TB/s	1.0 - 1.4 PF/s	0.095 - 0.266	1.21 - 3.38	Adaptive
Other Proposals	205 TB/s	2.6 PF/s	0.0788	0.82 - 0.30	Static

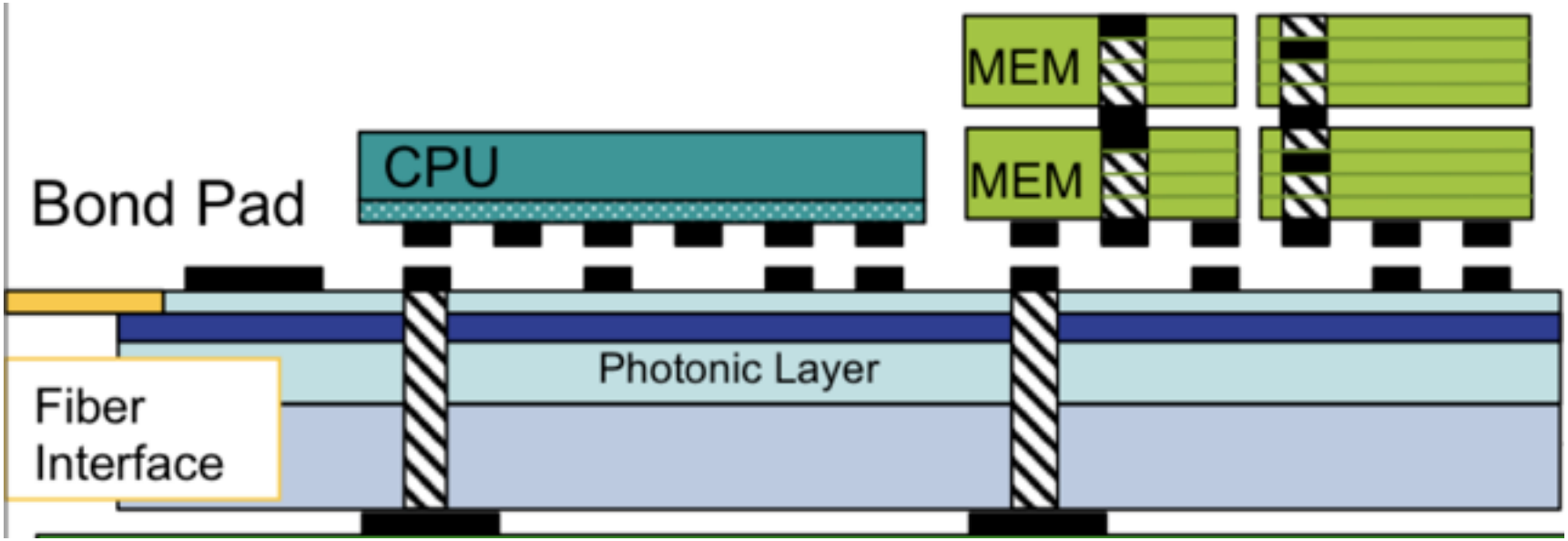
DARPA Challenge Problems

Problem Area	Standin Problem	Executes	Researcher Responsible	Quality
Graph	Graph500 Concurrent Search	EMP	Brian Barrett and Bruce Hendrickson	Integer Pointer Dereference
Stream	GUPS	EMP	Steve Plimpton	Input + Integer Pointer Dereference (latter harder)
Decision Support	Chess	EMP	Thomas Sterling	Integer Pointer Dereference
Shock Physics	MiniFE	EMP + P	Mike Heroux	Integer Pointer Dereference + 12% FP
Molecular Dynamics	MiniMD	P	Marc Snir and Steve Plimpton	15% FP with lots of local references

Our Enabling Technologies: Advanced Packaging, 3D Integration, Optics







We are not investing in baseline transistor technology since fab companies have to make that work for industry.

Why Invest in Optics (when the Kogge-reports says they're not viable)?

$$\frac{8MW}{8*10^{18}b/s} \approx 1pJ/bit$$

Modulation (E to O):

- **3 fJ/bit** modulation has been demonstrated[3]

Modulator thermal control and trimming:

- 106 GHz within die frequency variation has been measured and a thermal resonance shift of 4.4 μW/GHz has been demonstrated[1] → **<23 fJ/bit>** thermal trimming power
- 4.4 fJ/bit-°C thermal control has been demonstrated[2], so for +/- 10 °C swing in operating temperature → **<44 fJ/bit>** thermal control power

Optical Demux:

- 2 ring-filter with thermal control and trimming → **<134 fJ/bit>**

Receiver (O to E):

- A -18.9 dBm sensitivity integrated Ge receiver has been demonstrated[4] at 5 Gbps, with a total power consumption of **690 fJ/bit**

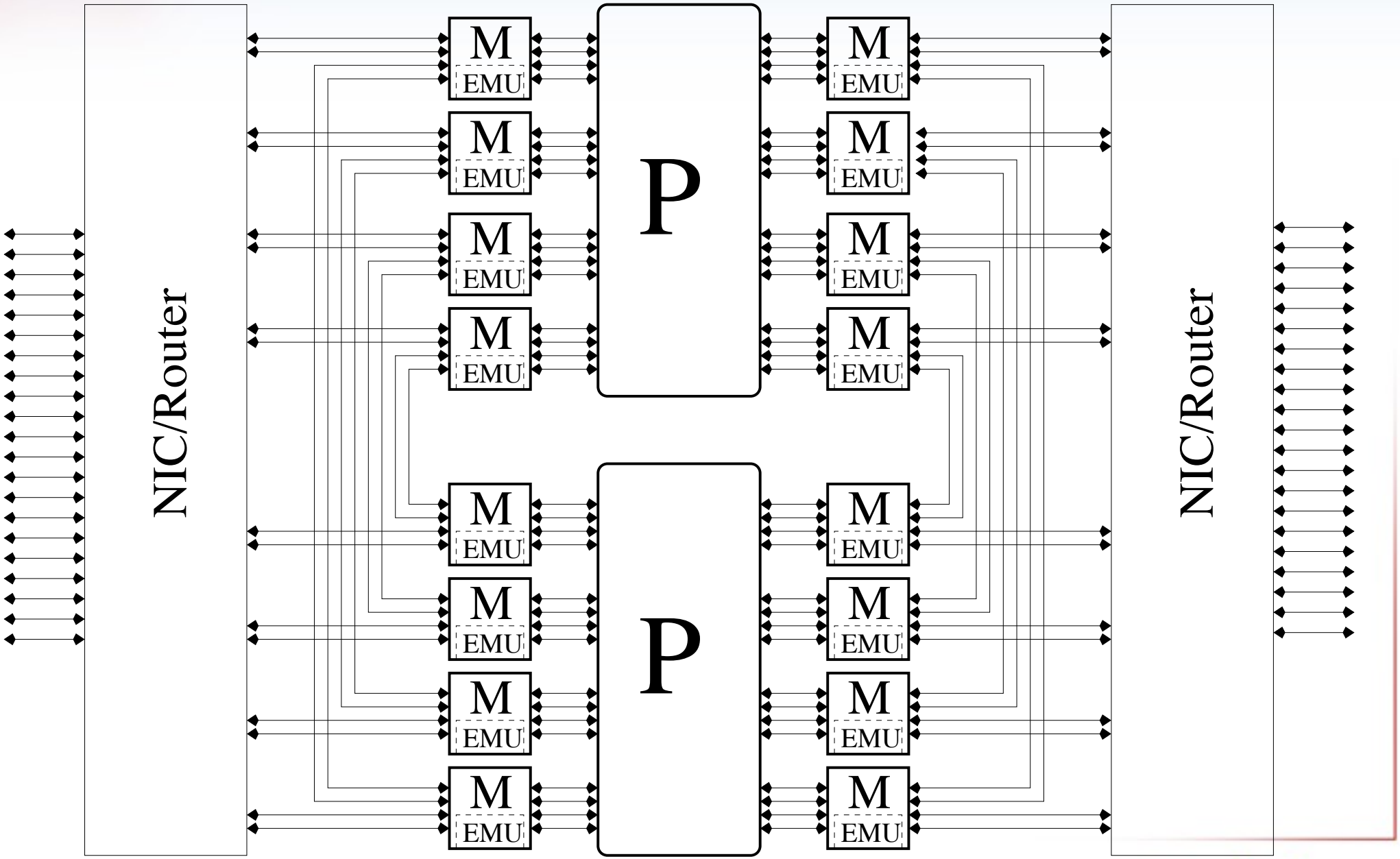
Optical Source:

- -18 dBm power required at receiver. Assuming 2 dB/facet coupling loss (demonstrated) and another 2 dB of on-chip loss then for a wall plug efficiency of 10 % (includes TEC power consumption) then 1 mW is required for the optical source → **100 fJ/bit**

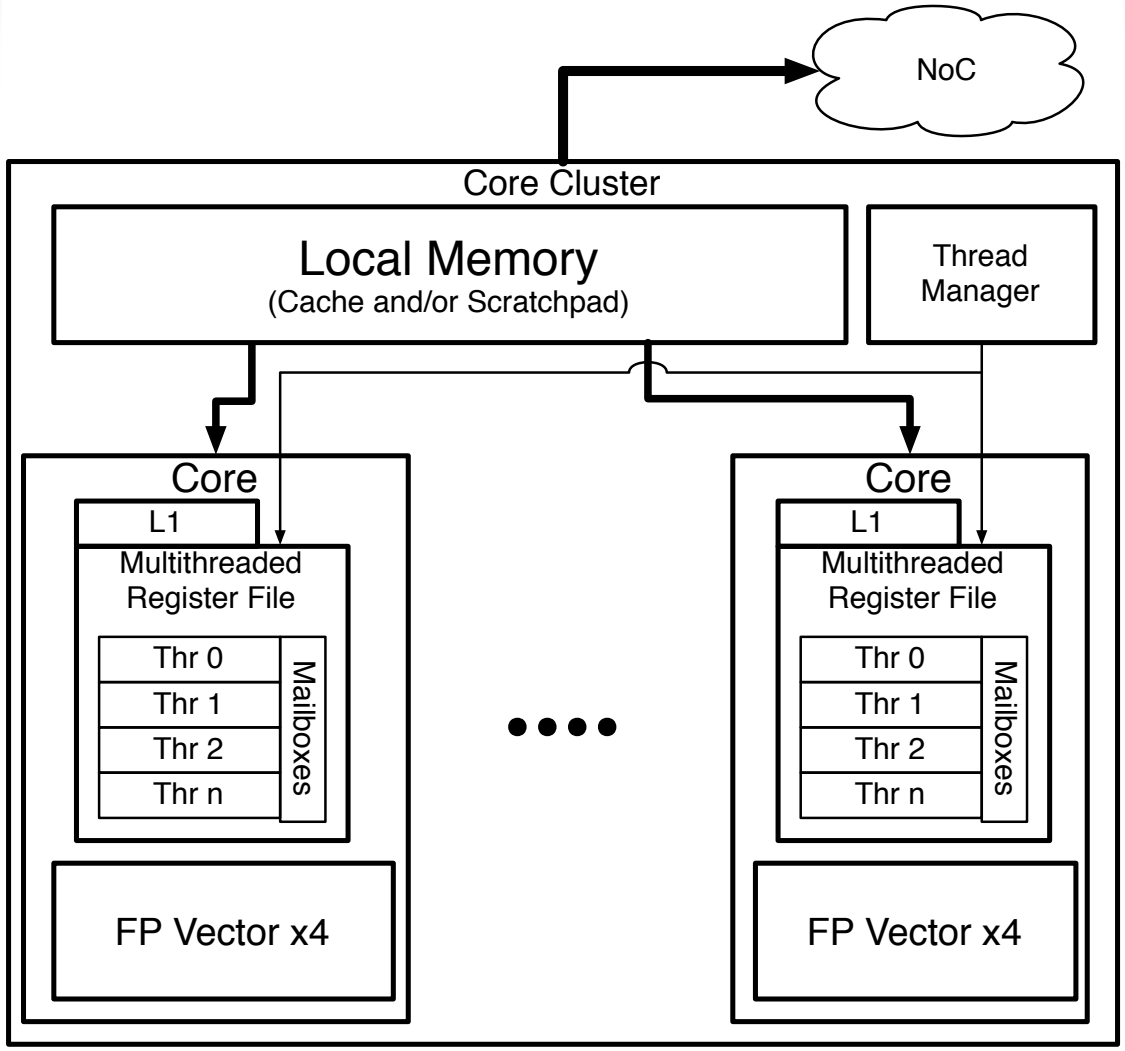
[1] W. Zortman, *et al*. to be published
[2] C.T. DeRose, *et al*. CLEO (2010)
[3] W. Zortman, *et al*. CLEO (2010)
[4] Zheng, *et al*. Opt. Exp. **18** pp 204-211 (2009)

Total power consumption: **~1 pJ/bit**

Node Architecture (Continued)

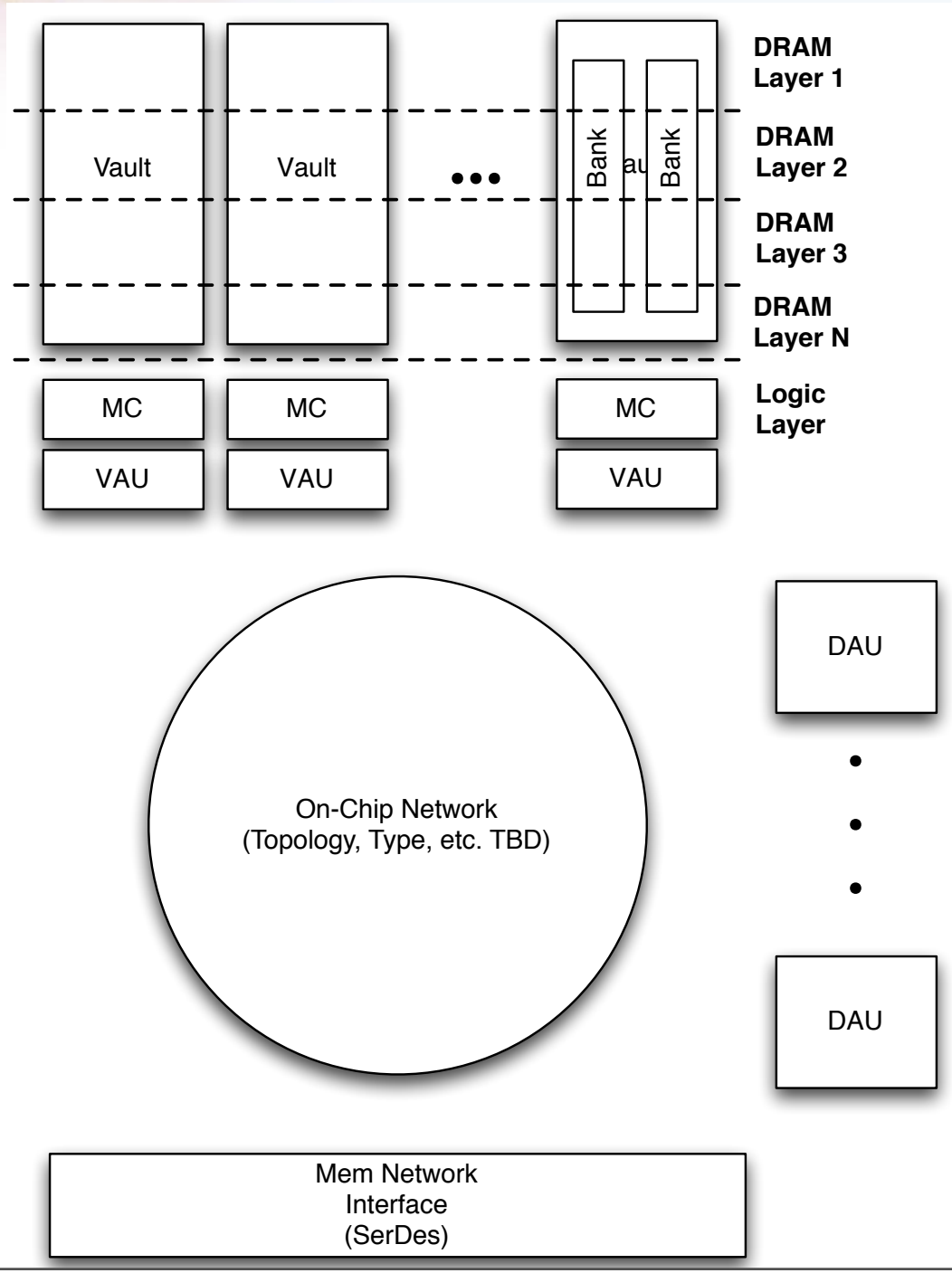


Processor (P)



- X-caliber more concerned with data movement
- Hybrid CPU-like and GPU-like architecture
- Heavily Threaded and Vectored
- Client of the Memory Network
- Owns only cache/ scratchpad memory

Memory System (M)



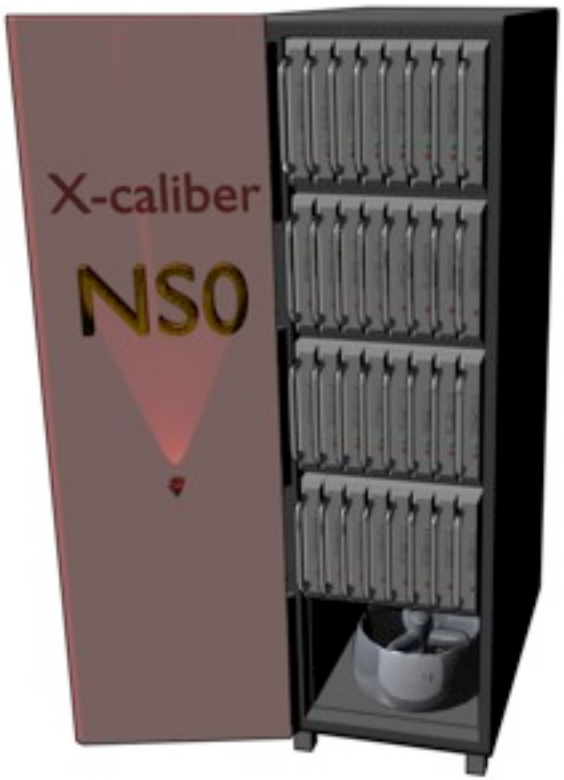
- Two computation Units
  - Right next to the DRAM vault memory controller (VAU)
  - To aggregate between DRAM vaults (DAU)
- "Memory Network" Centric
- Homenode for all addresses
  - Owns the "address"
  - Owns the "data"
  - Owns the "state" of the data
  - Can build "coherency"-like protocols via local operations
  - Can support PGAS-like operations
  - Can manage thread state locally

Sandia National Laboratories

Sprinting

- Every major component of the system has the capability to "sprint" by operating outside it's nominal power envelope
  - Processor: Increases the clock rate from 1.5 GHz to 2.5 GHz
    - Can be applied to half the cores and allow "ping-ponging"
  - Memory: Additional memory links (increasing concurrency and bandwidth) can be powered up in sprint mode
  - Network: Sprint on injection bandwidth from 512 GB/sec on the NIC to 1 TB/s
- Decisions about when to sprint are made dynamically by the runtime and OS

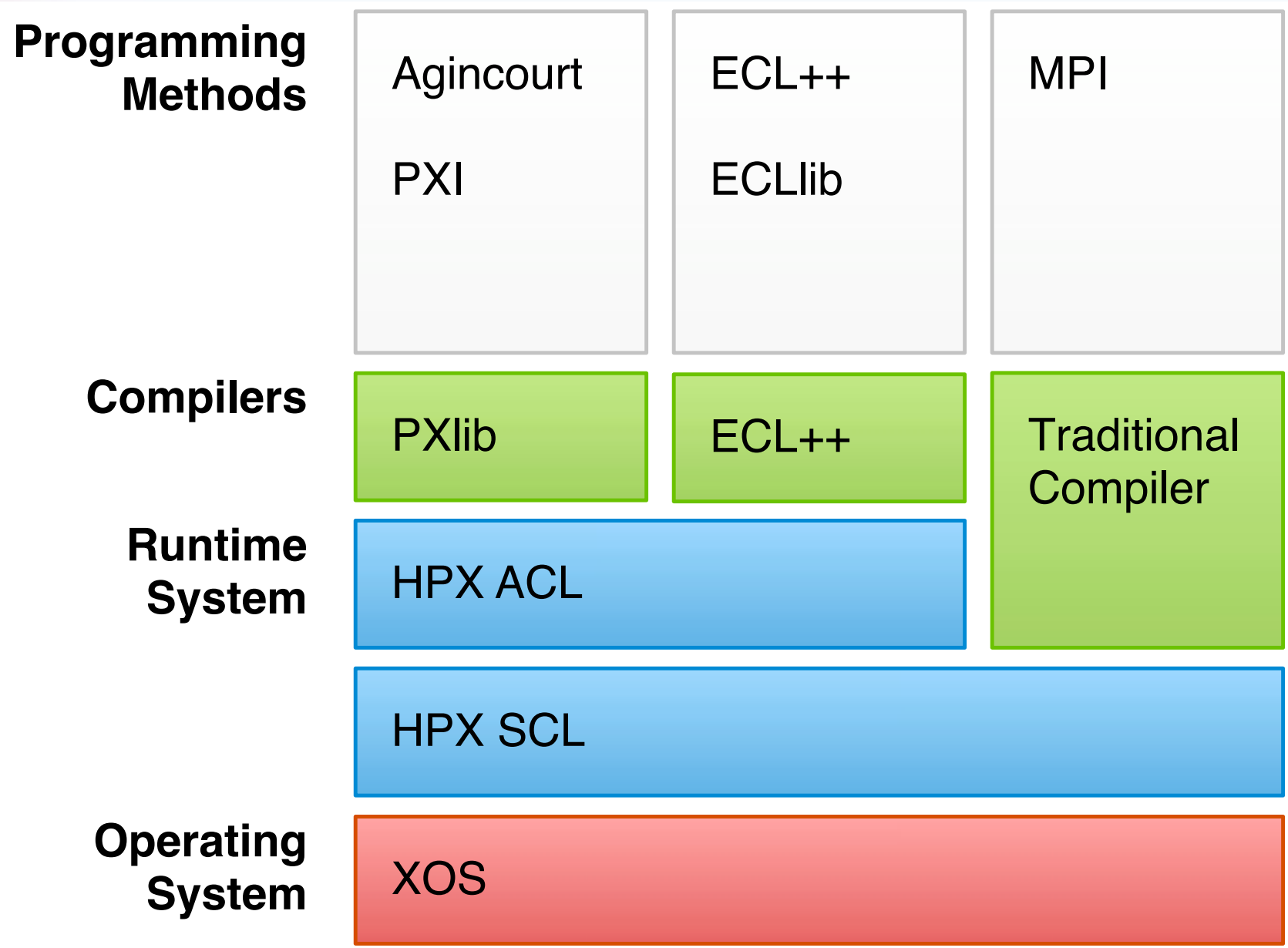
Target Scales



- **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

X-caliber Software Stack



Graph500: We can't optimize what we don't measure

- To announced at ISC'10, first list at SC10
  - submission open until 11/8
- Three kernels under discussion
  - 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
- 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

Thank You!