

Trends in Scientific Discovery Engines

Mark Stalzer

Center for Advanced Computing Research

California Institute of Technology

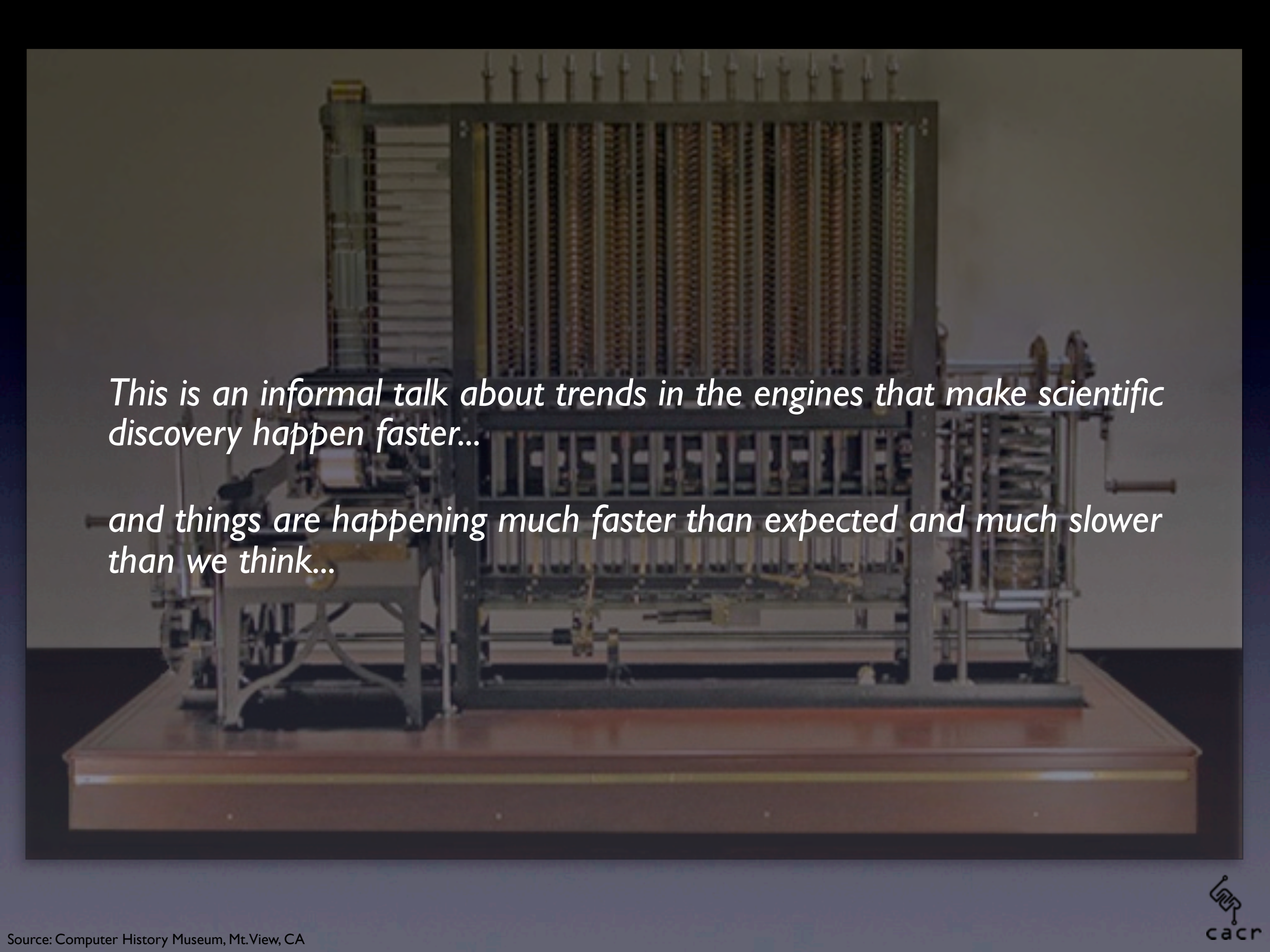
stalzer@caltech.edu

www.cacr.caltech.edu

AstroInformatics, September 2012

Redmond WA





This is an informal talk about trends in the engines that make scientific discovery happen faster...

and things are happening much faster than expected and much slower than we think...

Some Truths about Computing


- It must be COTS (cheaply manufacturable IP)
 - ▶ Cray-I was COTS: 6 dual in-out ECL gates per chip
 - ▶ Drivers: missile guidance and virtual missile guidance
- Advanced computing systems are all about **power** and packaging
 - ▶ Batteries and 100 MW power plants are expensive
 - ▶ Apple iPad and Cray-I
- Advanced computers are hard to program
 - ▶ But they can be easy to use
 - ▶ It only takes a few good abstractions

Trends in Simulation Engines:

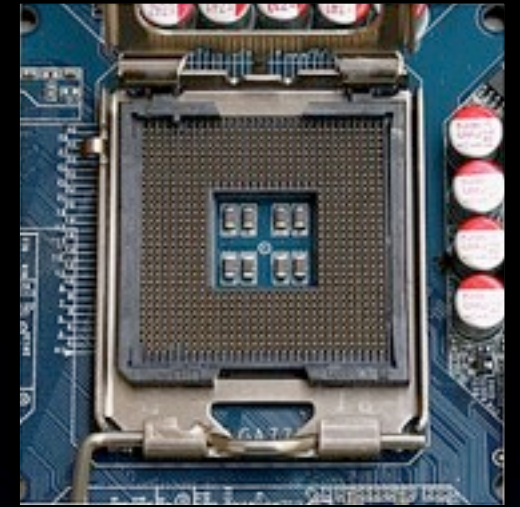
Exascale or Bust

(or computing without data)

Top 10 Supercomputers

Rank	Site	Computer/Year Vendor	Cores	R _{max}	R _{peak}	Power
1	DOE/NNSA/LLNL United States	Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom / 2011 IBM	1572864	16324.75	20132.66	7890.0
						
8	Germany	16C 1.60GHz, Custom / 2012 IBM	131072	1360.39	1677.72	657.3
9	CEA/TGCC-GENCI France	Curie thin nodes - Bullx B510, Xeon E5- 2680 8C 2.700GHz, Infiniband QDR / 2012 Bull	77184	1359.00	1667.17	2251.0
10	National Supercomputing Centre in Shenzhen (NSCS) China	Nebulae - Dawning TC3600 Blade System, Xeon X5650 6C 2.66GHz, Infiniband QDR, NVIDIA 2050 / 2010 Dawning	120640	1271.00	2984.30	2580.0

Socket Parallelism: Top500 2002 vs. 2012



Optimized for dot products:

```
complex s = 0;
```

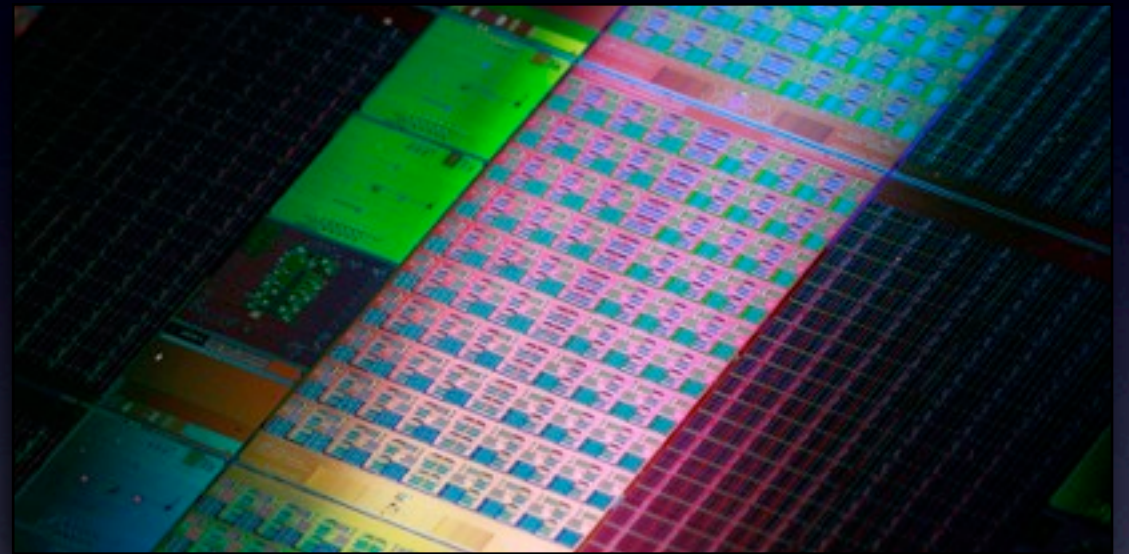
```
for (complex* lp = xp + n; s += *xp++ * *yp++; xp < lp) ;
```

	ASCI White #2	Sequoia #1	GAIN
Rmax (Tflops)	7.226	16,325	2,260x
Processor	IBM Power 3	IBM BQC 16C	
Clock (Ghz)	0.375	1.6	4.27x
#Sockets	8,192	98,304	12x
Socket Parallelism	1	64	64x (3,280)

Half of parallelism increase is on-socket, and getting “worse”...

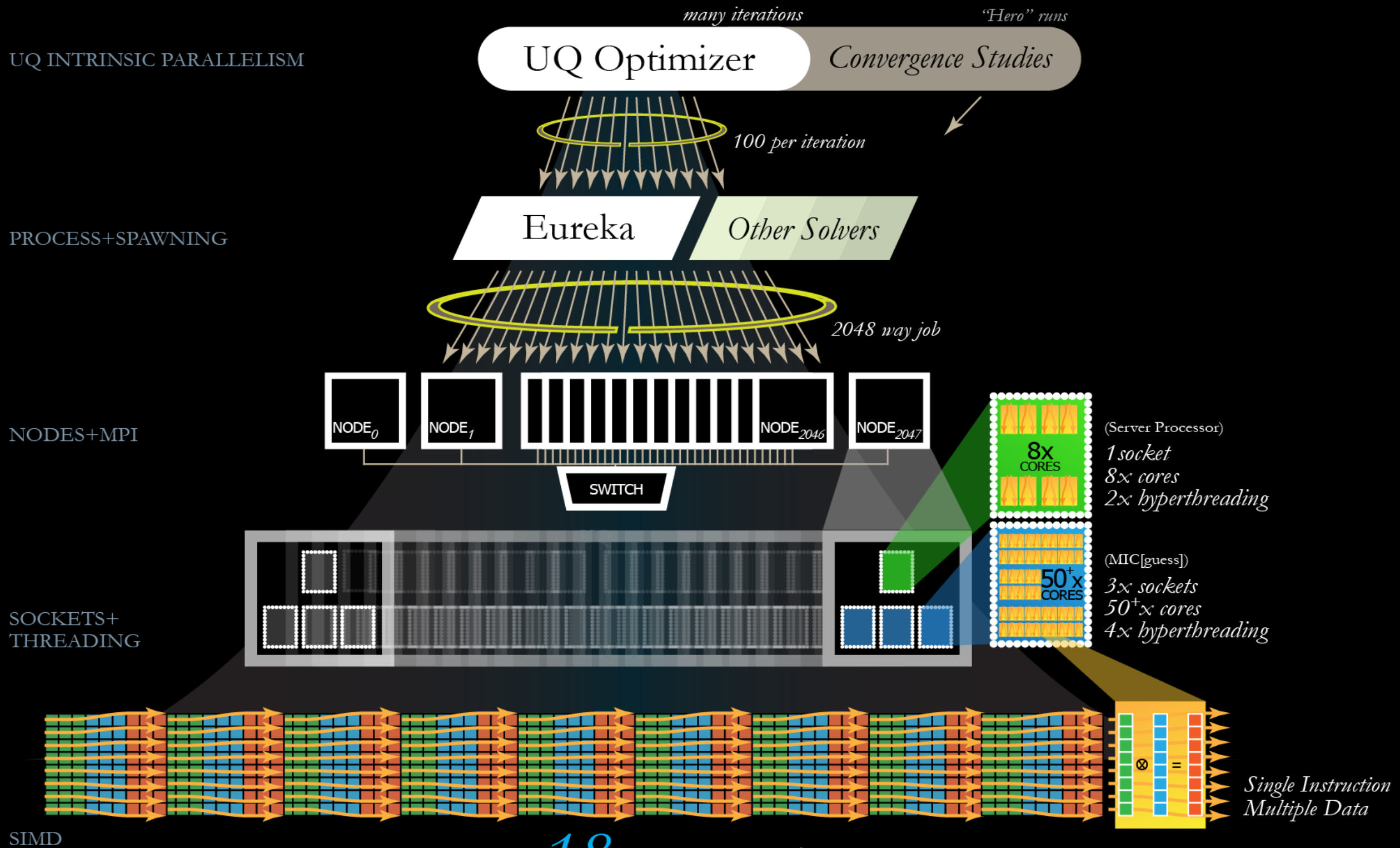
Intel Many Integrated Core (MIC)

- 50+ IA64 cores (Pentium-like)
 - ▶ 8 (double) SIMD
 - ▶ 200+ hardware threads
 - ▶ Tflops per socket
- Will scale to $O(1,000)$ threads ~2 yrs
 - ▶ Cache coherent in socket
 - ▶ ASCI White performance!
- Qualitatively different programming model
 - ▶ MPI between sockets
 - ▶ Massive **general** threading in a socket: must rewrite codes
- *Example: Image processing parallelism*



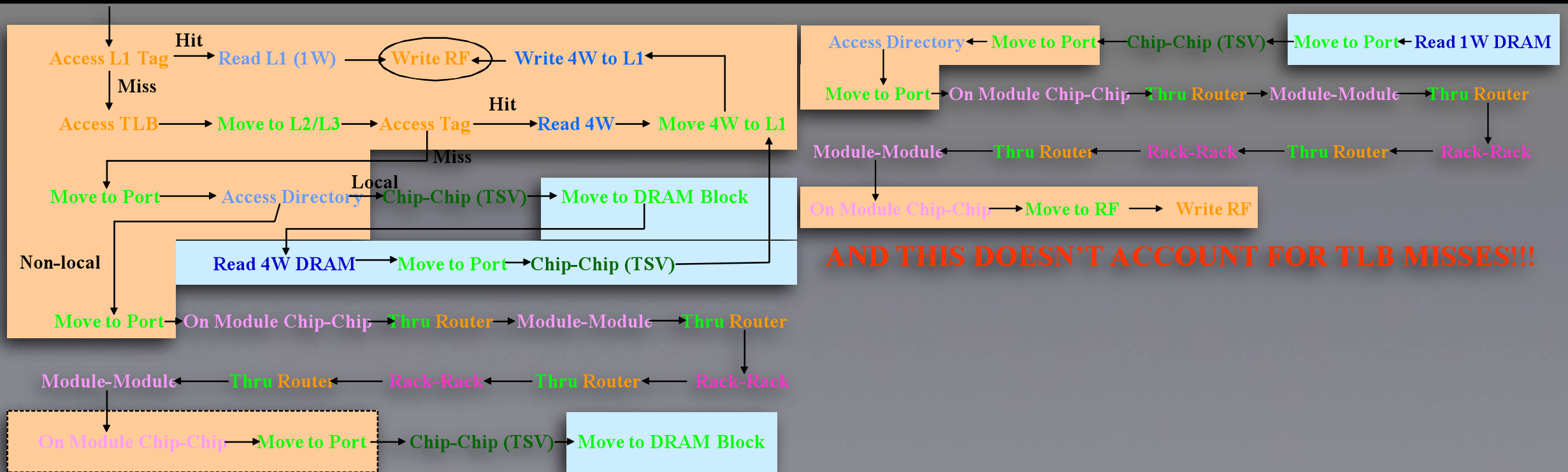
“Socket Archipelago”

UQ +Eureka and Extreme Parallelism



* Assuming full pipelines, no hyperthread contention, and a power plant

Follow the Power



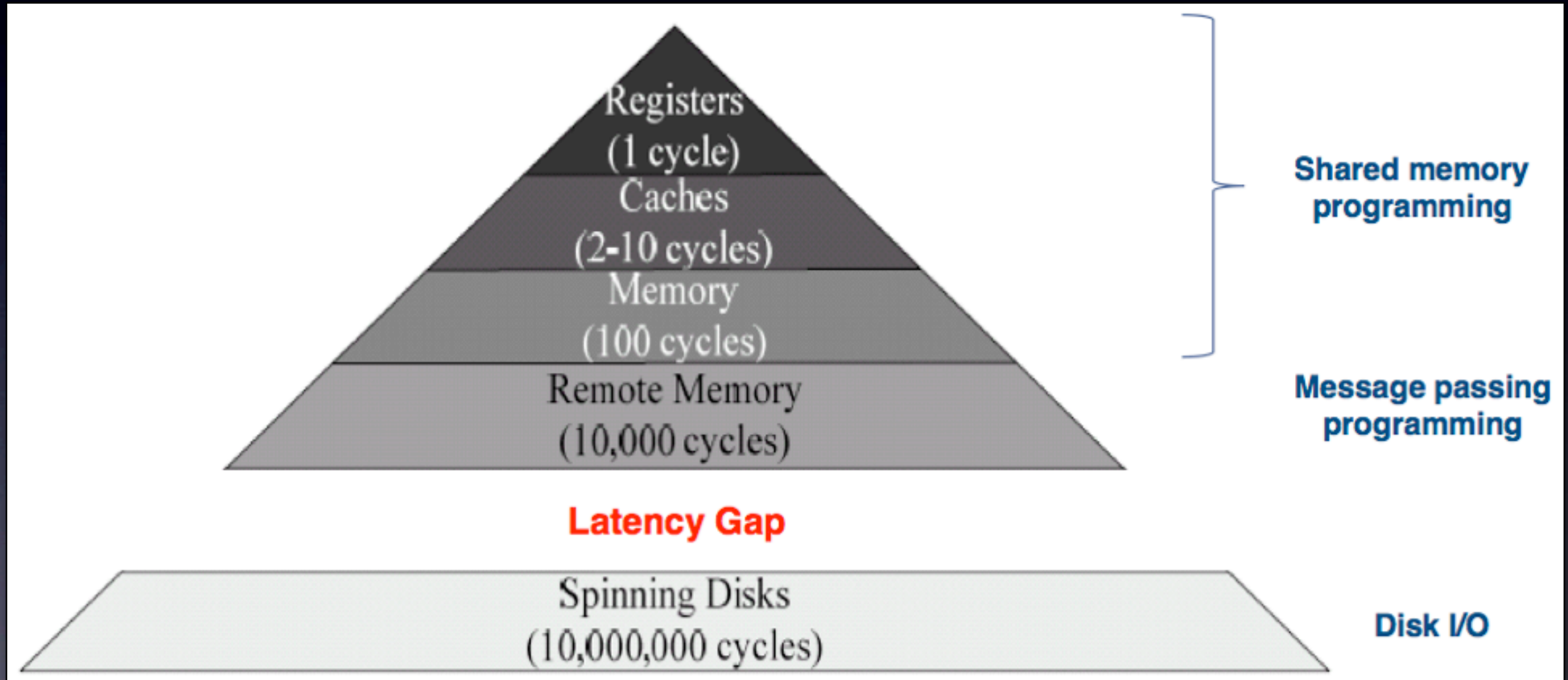
Operation	Energy (pJ/bit)
Register File Access	0.16
SRAM Access	0.23
DRAM Access	1
On-chip movement	0.0187
Thru Silicon Vias (TSV)	0.011
Chip-to-Board	2
Chip-to-optical	10
Router on-chip	2

Step	Target	pJ	#Occurrences	Total pJ	% of Total
Read Alphas	Remote	13,819	4	55,276	16.5%
Read pivot row	Remote	13,819	4	55,276	16.5%
Read 1st Y[i]	Local	1,380	88	121,400	36.3%
Read Other Y[i]s	L1	39	264	10,425	3.1%
Write Y's	L1	39	352	13,900	4.2%
Flush Y's	Local	891	88	78,380	23.4%
Total				334,656	
Ave per Flop				475	

In 2015, a flop will be ~10 pJ: It takes ~50x energy just to move the bits!

Trends in Data to Discovery Engines

Latency Gap



Several efforts at closing the latency gap using flash memories...

Gordon Supernode Architecture

32 Appro Extreme-X compute nodes

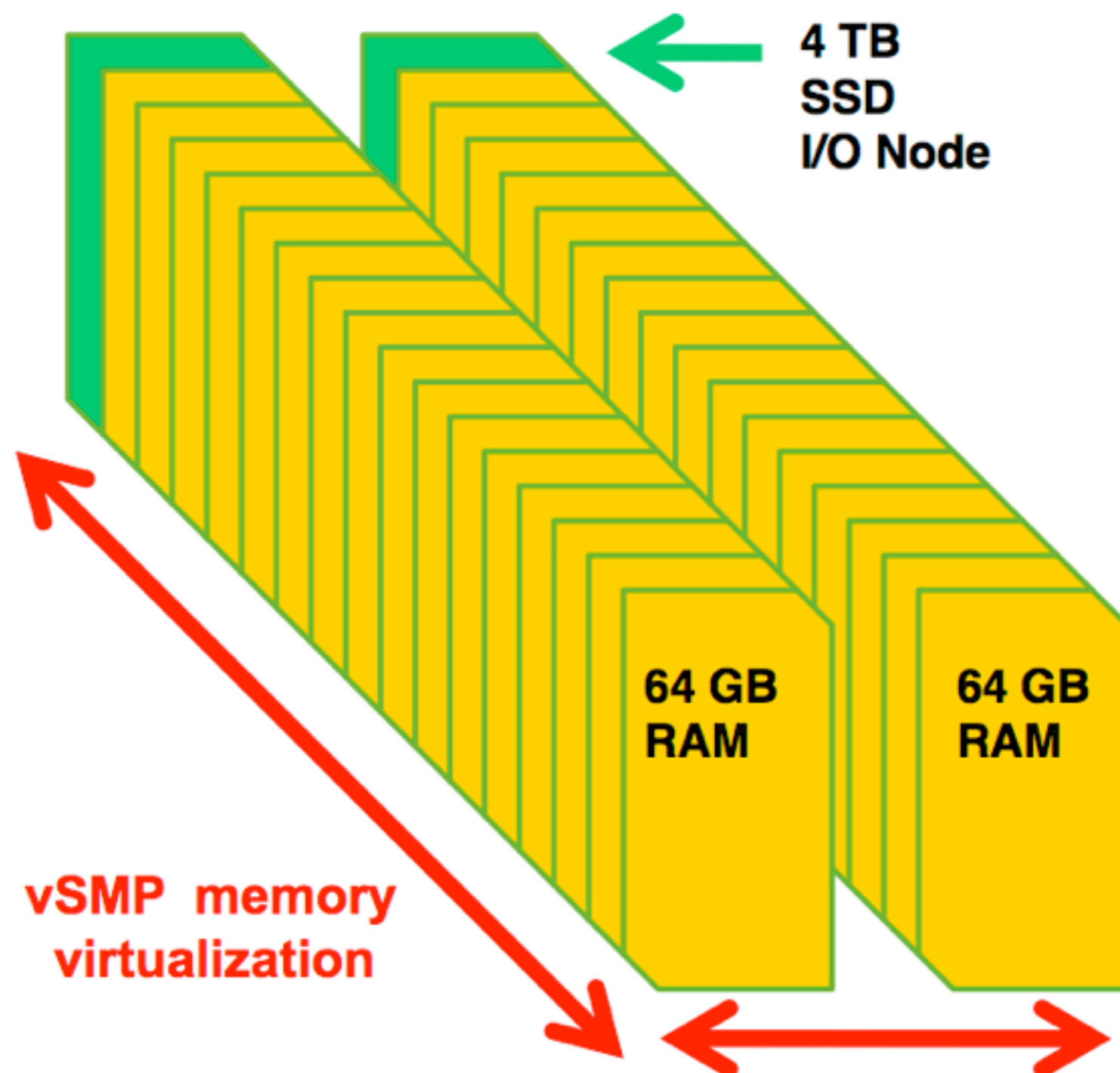
- Dual processor Intel Sandy Bridge
- 64 GB

2 Appro Extreme-X IO nodes

- Intel SSD drives
- 4 TB ea.
- 560,000 IOPS

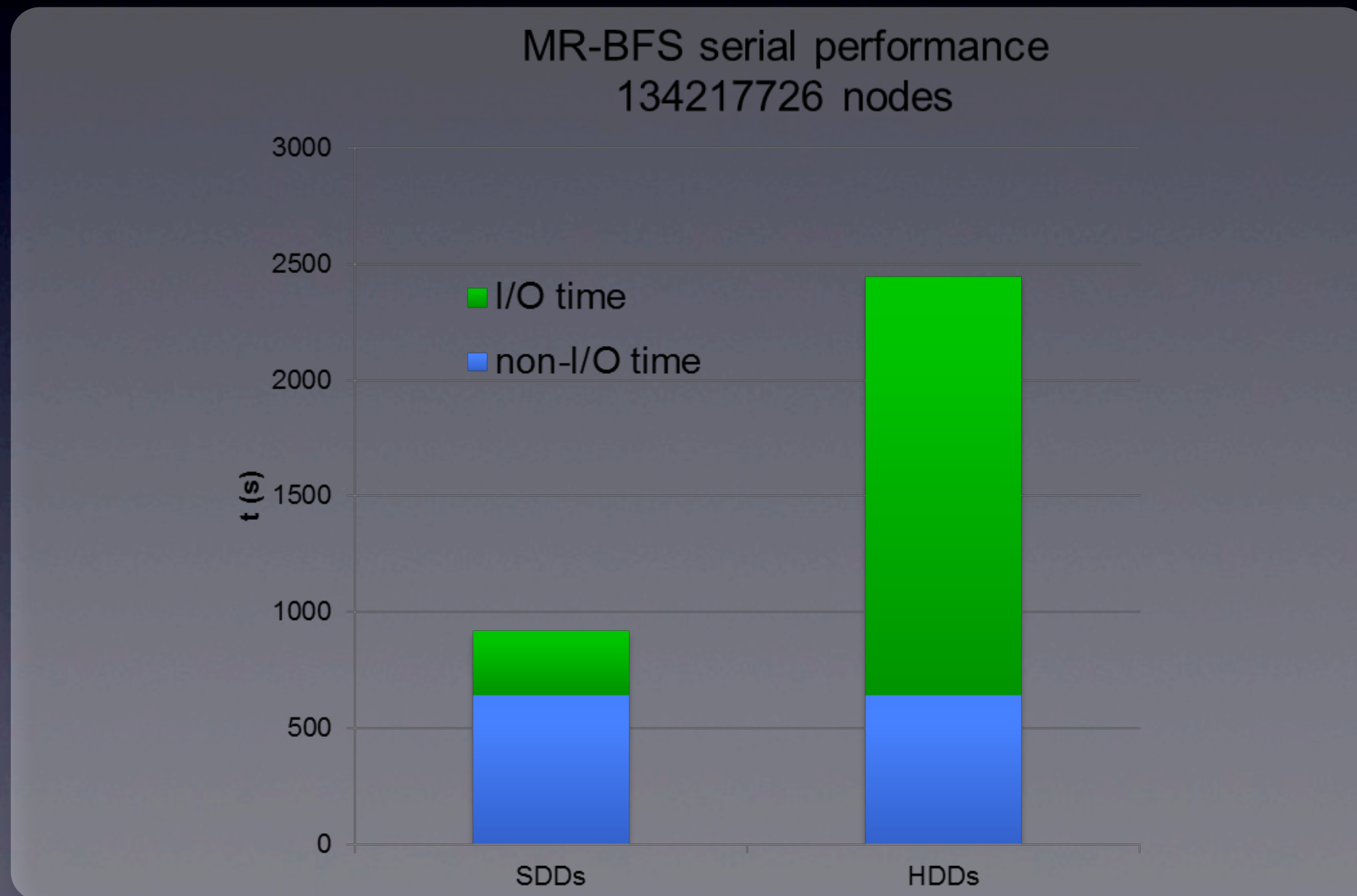
ScaleMP vSMP virtual shared memory

- 2 TB RAM aggregate
- 8 TB SSD aggregate



Full machine is 32 supernodes interconnected by dual-rail QDR IB in 3D torus.

Gordon BFS Performance



6.5x Improvement - Available now through XSEDE (www.xsede.org).

Amdahl-Balanced Blades



- Gene Amdahl's Laws for I/O & memory (1965, 2007):
 - ▶ A bit of seq. I/O per sec. per instruction per sec. (*Amdahl #*)
 - ▶ Mbytes / MIPS ~ 1 (*Memory ratio*)
 - ▶ One I/O operations per 50,000 instructions (*IOPS ratio*)
- Simulation codes may have an Amdahl # of 10^{-5} ; data intensive apps may need ~ 1
- Szalay, Bell, Huang, Terzis, White (Hotpower-09):

Table 2: Performance, power, and cost characteristics of various data-intensive architectures.

	CPU [GHz]	Mem [GB]	SeqIO [GB/s]	RandIO [kIOPS]	Disk [TB]	Power [W]	Cost [\$]	Relative Power	Amdahl numbers		
									Seq	Mem	Rand
GrayWulf	21.3	24	1.500	6.0	22.5	1,150	19,253	1.000	0.56	1.13	0.014
ASUS	1.6	2	0.124	4.6	0.25	19	820	0.017	0.62	1.25	0.144
Intel	3.2	2	0.500	10.4	0.50	28	1,177	0.024	1.25	0.63	0.156
Zotac	3.2	4	0.500	10.4	0.50	30	1,189	0.026	1.25	1.25	0.163
AxiomTek	1.6	2	0.120	4.0	0.25	15	995	0.013	0.60	1.25	0.125
Alix 3C2	0.5	0.5	0.025	N/A	0.008	4	225	0.003	0.40	1.00	

Cyberbricks

- 36-node Amdahl cluster at 1,200 W total!
 - ▶ N330 dual core Atom, 16 GPU cores, 4 GB
- Aggregate disk space of ~43 TB
 - ▶ About 1 SSD 120 GB per core (~8 TB)
 - ▶ 35 TB of spinning disk
- Blazing I/O performance: 18 GB/s
- Amdahl # = 1 for under \$30 K
- Using the GPUs for data mining:
 - ▶ 6.4 B multidimensional regressions in 5 min over 1.2 TB
 - ▶ Ported RF module from R to C#/CUDA



Calxeda/HP Moonshot



“The EnergyCore is a single chip with a Cortex-A9 ARM processor running between 1.1GHz and 1.4GHz. The chip includes 4MB of cache, an 80-Gigabit fabric switch and a management engine for power optimization. Servers with the chip, 4GB of memory and a large-capacity solid-state drive [SATA] draw 5 watts of power. Besides using a low-power ARM processor, Calxeda has cut down chip power consumption by integrating key server components.”

- Agam Shah, Computerworld, Nov 11, 2011

Gordon and Cyberbricks are real

(and 6-10x is great but still constrained by I/O architecture)

What if we do “make believe” computer architecture?

Power Miser Devices: Apple A5 to CI

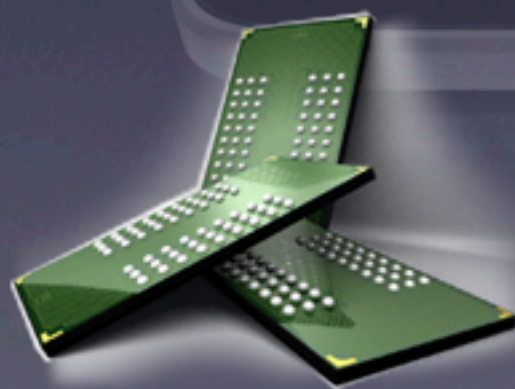
- A5



- ▶ SoC/PoP
- ▶ ARM/GPU/USB 2.0/Flash cntrl.
- ▶ 512 MB
- ▶ 10 Gflops? at 1W?
- ▶ 64 GB NAND Flash

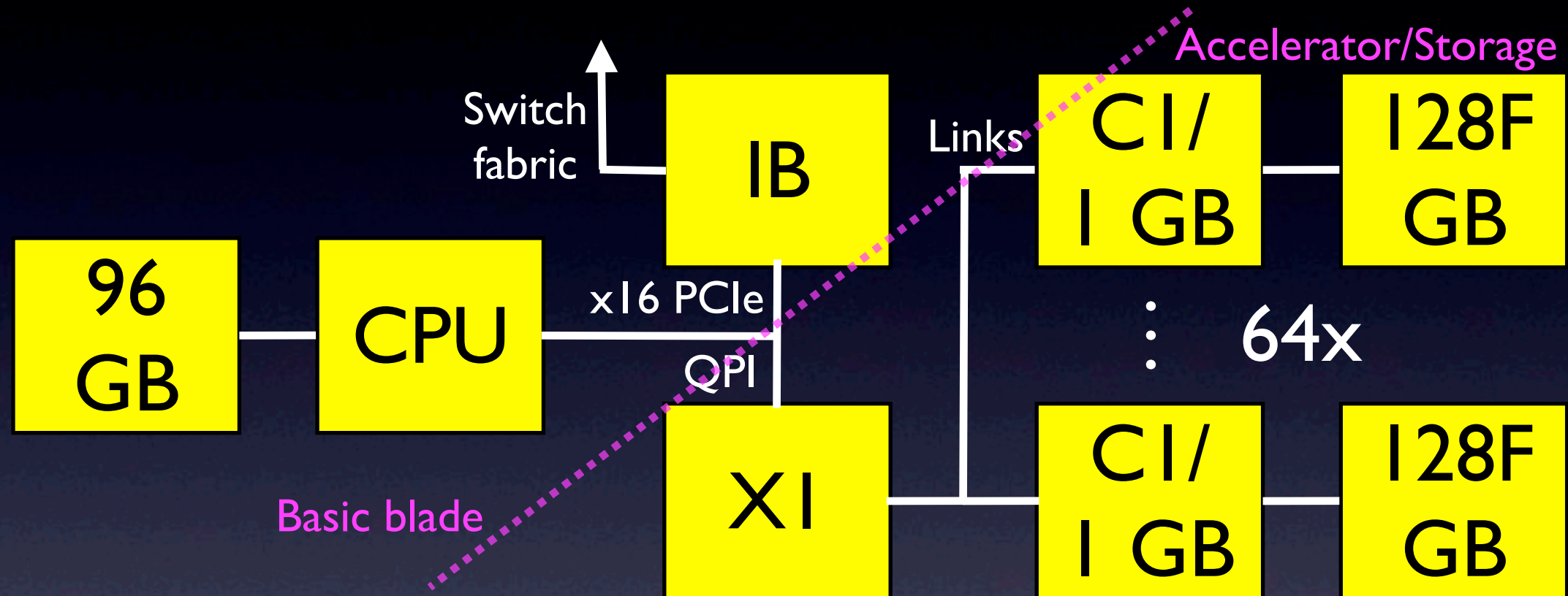


- CI



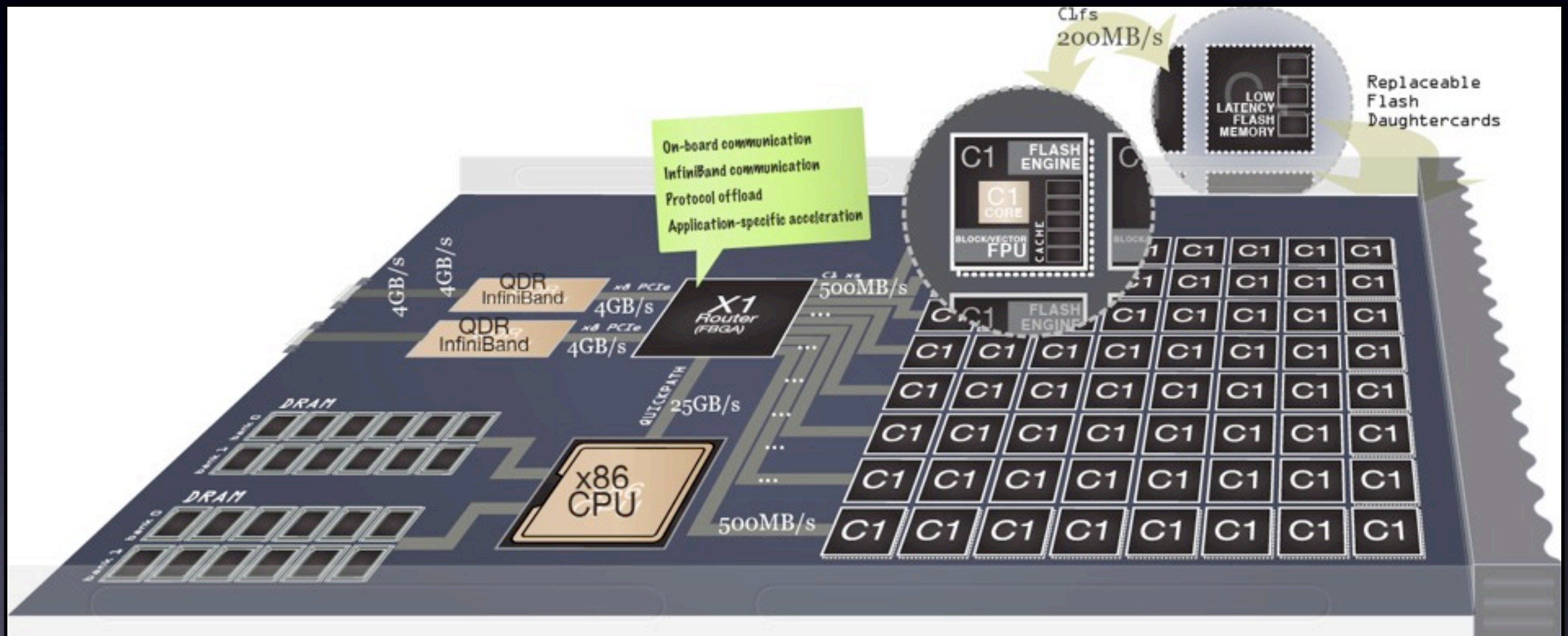
- ▶ Proc./Accel./Link/Flash engine
- ▶ 1 GB LDDR2 (64 bit wide) PoP
- ▶ 64 Gflops at 6 W + 2 W (1 Ghz)
- ▶ 128 GB NAND Flash RAID
- ▶ All existing IP; need <1 year

A More Extreme Approach: FlashBlades



- “XI” is an FPGA switch for CI array & QPI to CPU & PCIe to IB
- The CPU orchestrates *abstractions*; to the CPU the array looks like:
 - ▶ A ~6 TB, 25 GB/s (burst), 50 us, || disk (file system, triple stores)
 - ▶ A ~4 TFlops accelerator (OpenCL with embedded triple stores)
- This all fits on a standard blade (2 sides) and uses commodity IP
 - ▶ Draws about 600 W and is 100x faster on disk operations

FlashBlade Packaging



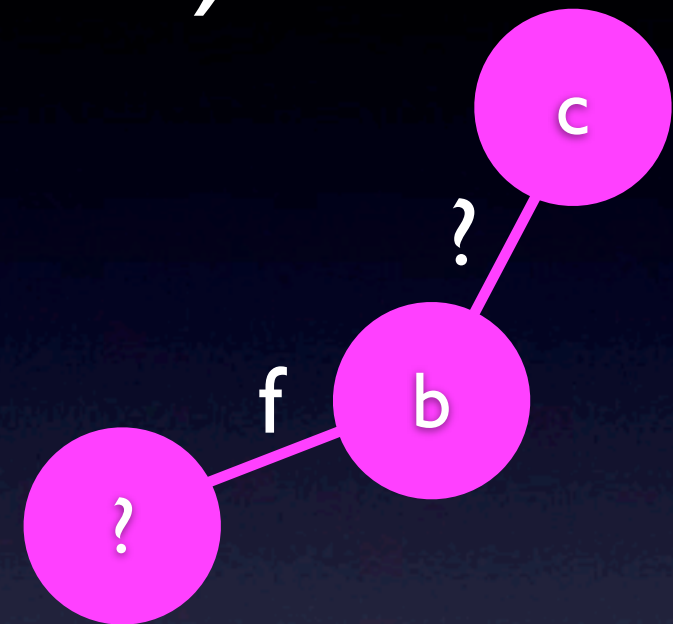
- Stalzer, FlashBlades: System Architecture and Applications, Workshop on Architectures and Systems for Big Data (ASBD), June 2012, Portland OR

Implications for Data to Discovery

- **HUGE** data processing capability: a single server can read (and “process”) its entire contents in about 10 minutes (200 MB/s)
 - ▶ A same size disk array would take 100x to read (2 TB disks)
 - ▶ 100x faster at random access (50 us vs. 5 ms)
 - ▶ Balanced I/O and computation
- *This is qualitatively new, what could we do with it?*
- Want applications with #reads >> #writes
 - ▶ One rack (0.5 PB) could handle LSST processing for a year?
 - ▶ Good for LHC/CMS data analysis (comparing data to Monte-Carlo simulations of the standard model)
 - ▶ Analytics, search, intelligence, video processing, tomography, ...
 - ▶ Metadata server for massive archival disk arrays

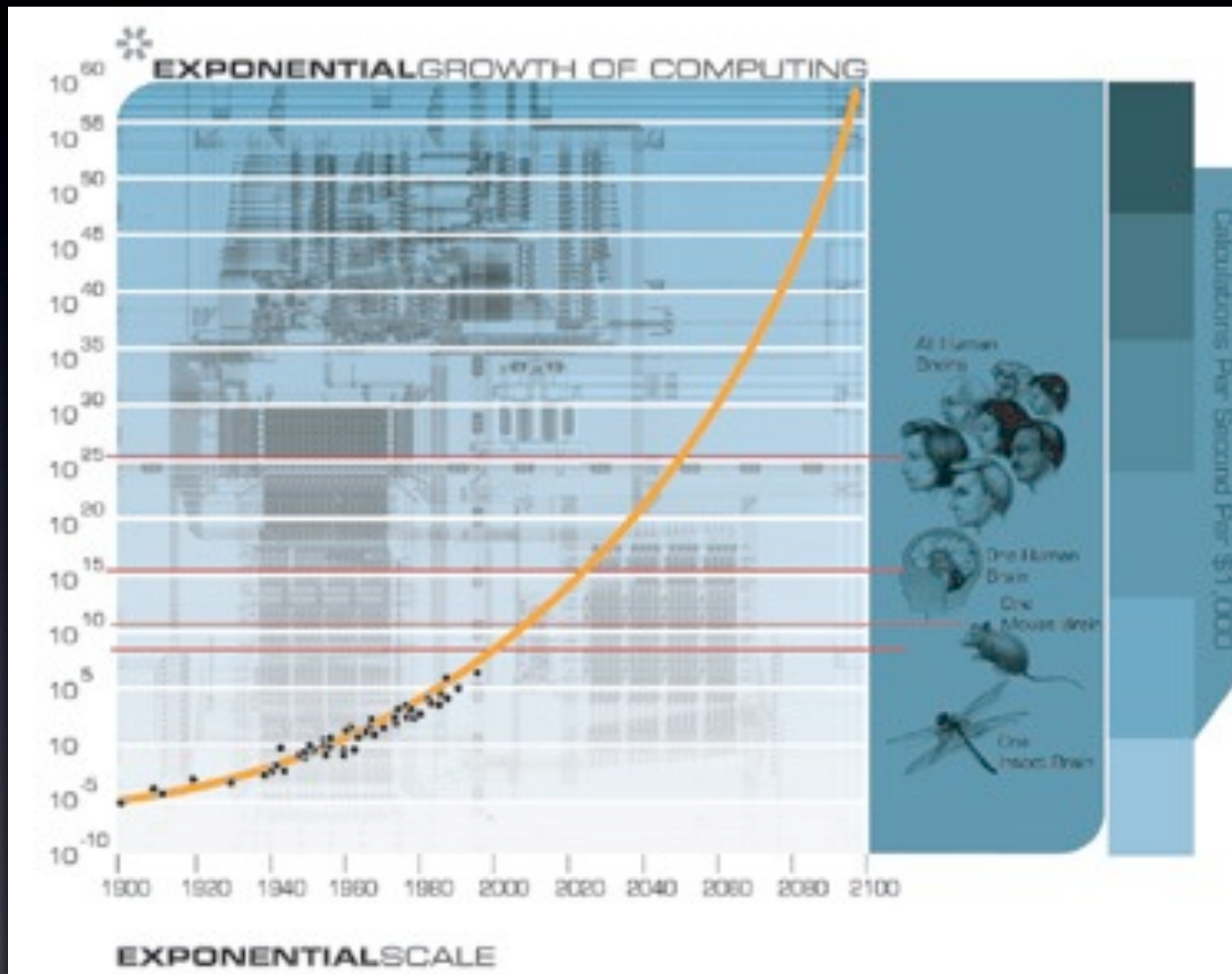
Triple Stores (a Storage Metric?)

- Triple stores
 - ▶ (object, attribute, value)
 - ▶ Graphs: (a, f, b)
- Query: “select (?, f, b) and (b, ?, c)”
- Implementation
 - ▶ Hash tables for partial triples (x, y, ?)
 - ▶ Performance order dependent; need working memory
- Use distributed hash tables across CI array
- Flash byte random access mode latency: ~1 us vs. 5,000 us
- **1,000x**
- Inferencing performance and mutation rate?



Another Big Data Implementation Technology

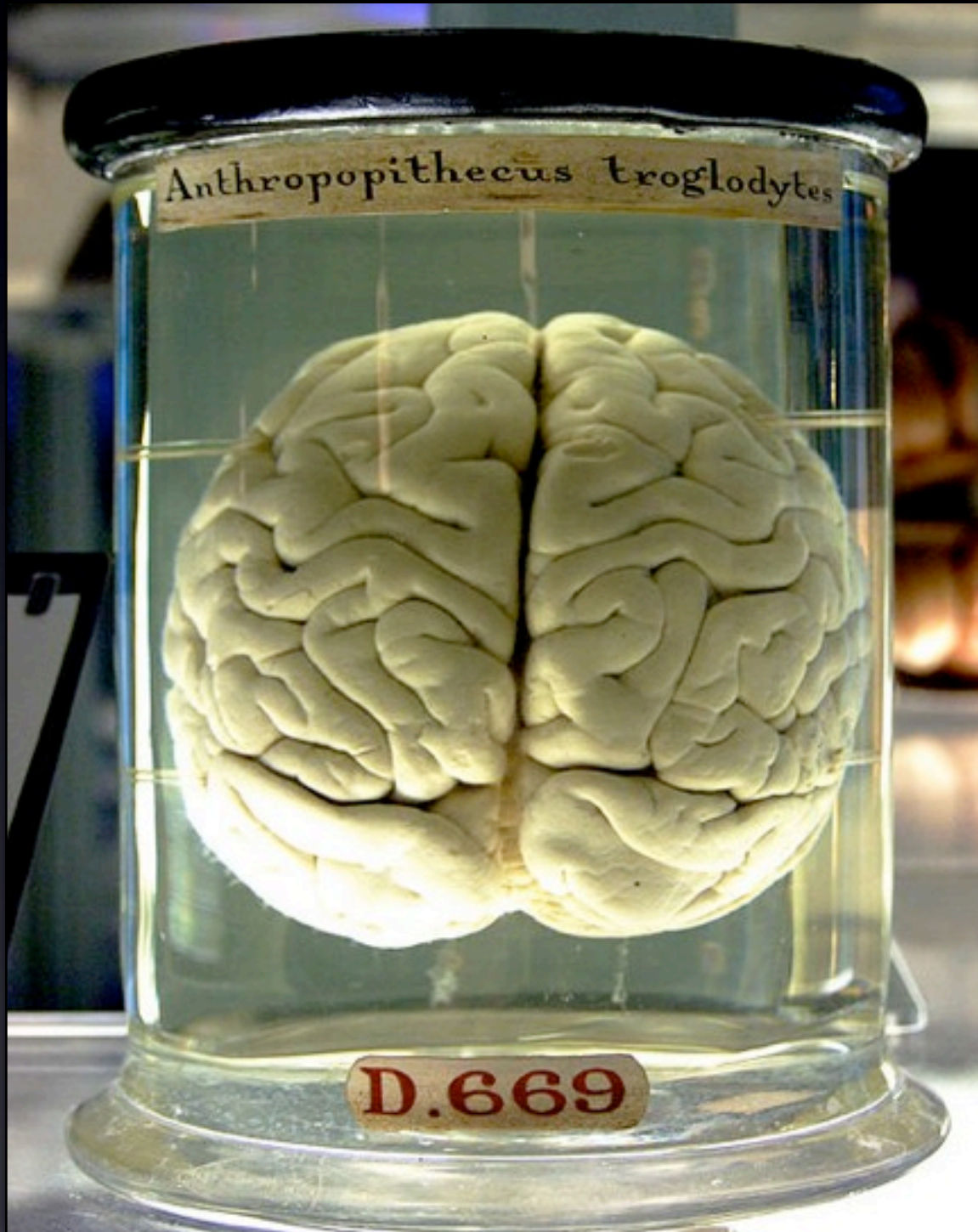
It Only Takes 10^{16} Ops (?)



A 10^{16} Flops Engine

- Quantity: Need about 200 servers (14 blades at 7U each)
 - ▶ Big IB switch fabric too (168 IB ports/rack)
 - ▶ Volume (need lots of air): 2,000 ft³
- Flash memory: ~17 PB
 - ▶ SDSC runs 18 PB of tape storage (1,000's of scientific data sets)
 - ▶ Constraint: no more than 1,000-2,000 writes/chip/day?
 - ▶ Data ingestion and reflective analysis by engine
 - ▶ Checkpoints in <10 s
- About 10 Pflops at ~2 MW (does not include cooling)
- Cost unknown due to rapidly dropping flash costs, new packaging, and other economies of scale

Comparison to a Natural Big Data Engine



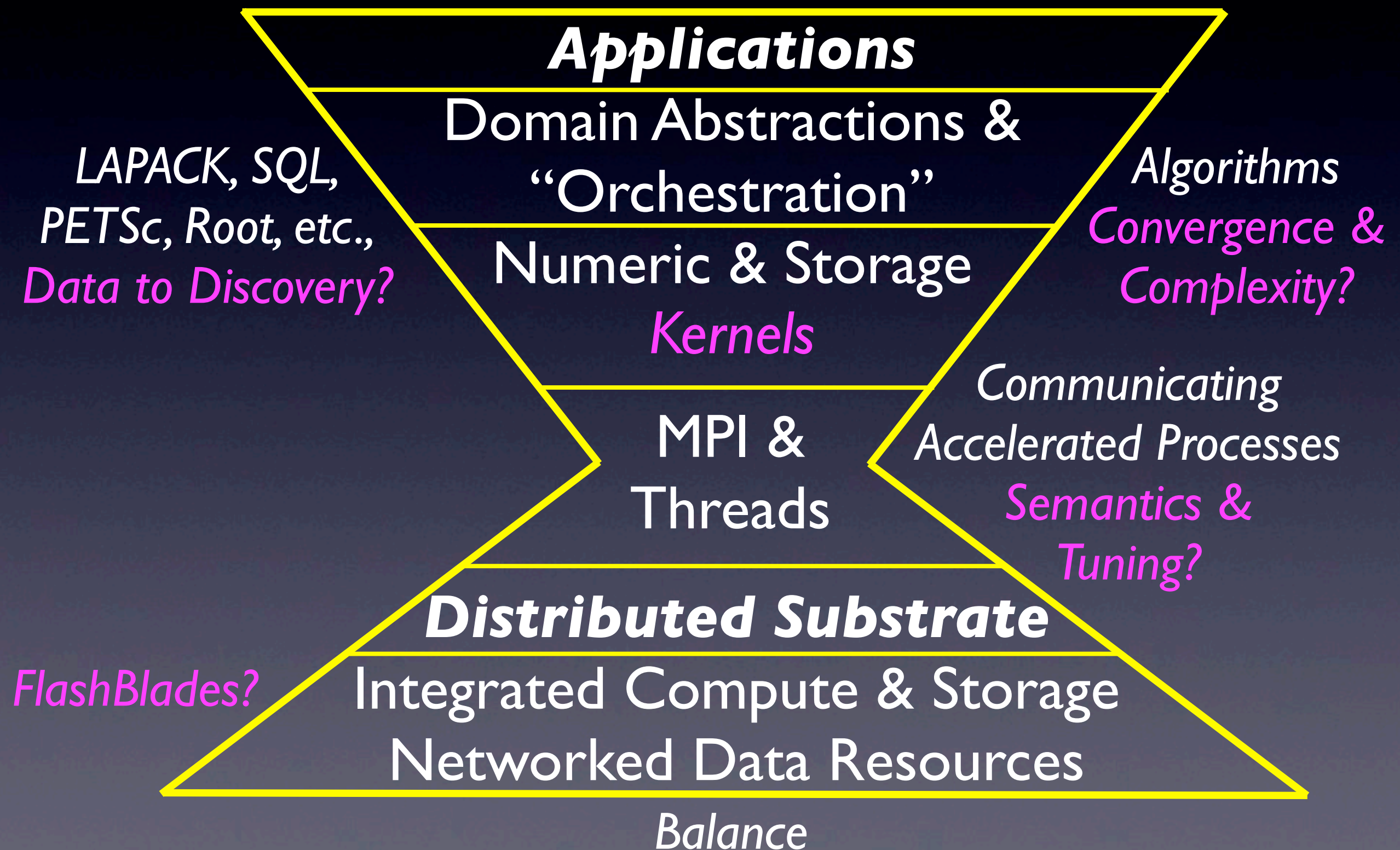
- Operations: 10 Pops (1x)
- Memory: 1 PB (0.06x & forgets)
- Bandwidth: 1 PB/s? (0.5x - 12x - 30x)
- Packaging: 0.25 ft³ (8,000x)
- Power: 25 W (80,000x)!
- *Where's the algorithm?*

Socket Archipelago (2017)

	Cluster	Socket
Parallelism	100,000	10,000
Rel. Latency	~1,000	1

- Parallelism is becoming dramatically bimodal
 - ▶ MPI (or process) is essentially island level parallelism
 - ▶ Threads are tribe level parallelism and much much faster
 - ▶ What if all threads want to talk with another island at once?
- Must have very large L2 cache on socket
- Non-volatile storage must be integrated too (stacking)
- *Analogy: cortical columns*

Engine Software Architecture



Concluding Remarks

- We are not stuck with “clusters”: COTS is also IP not just Fry’s.
- What is the Top500-like benchmark for data?
 - ▶ *Metrics drive development*
- Massively threaded programming is the future, but can be partially hidden by libraries (kernels).
- *Think in terms of what can be done with a shrinking 10^{16} ops system*