OSC Statewide Users Group Distinguished Lecture Series and
Ralph Regula School of Computational Science Lecture Series

Supercomputers and Clusters and Grids, Oh My!

Jack Dongarra
University of Tennessee
and
Oak Ridge National Laboratory

Take a Journey Through the World of High Performance Computing

Apologies to Frank Baum author of “Wizard of Oz”…

Dorothy: “Do you suppose we'll meet any wild animals?”

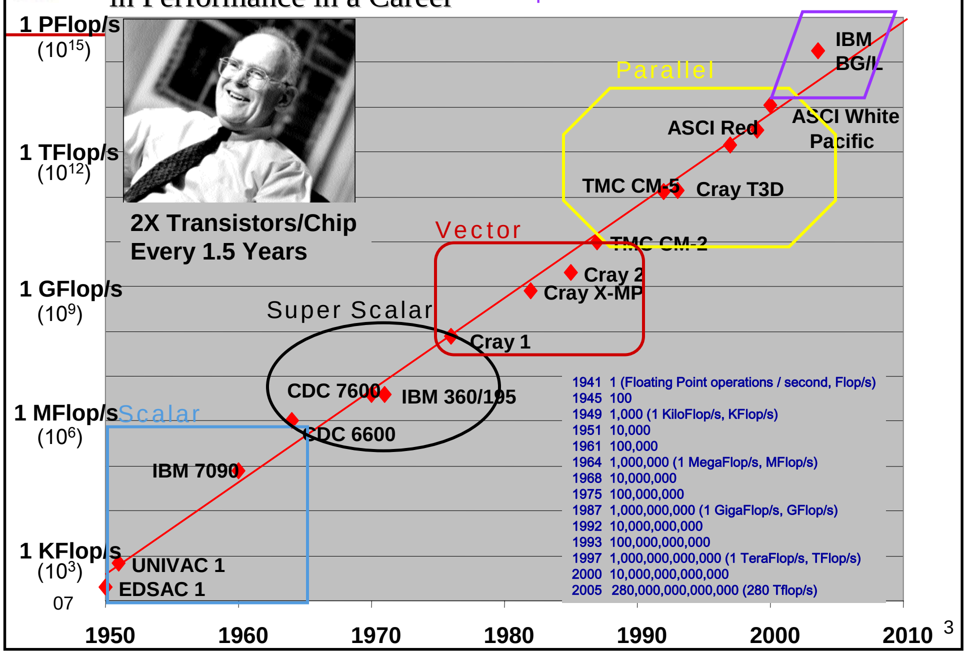
Tinman: “We might.”

Scarecrow: “Animals that ... that eat straw?”

Tinman: “Some. But mostly lions, and tigers, and bears.”

All: Supercomputers and clusters and grids, oh my!
Supercomputers and clusters and grids, oh my!

A Growth-Factor of a Billion in Performance in a Career



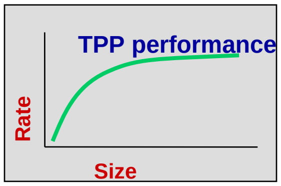
1941	1 (Floating Point operations / second, Flop/s)
1945	100
1949	1,000 (1 KiloFlop/s, KFlop/s)
1951	10,000
1961	100,000
1964	1,000,000 (1 MegaFlop/s, MFlop/s)
1968	10,000,000
1975	100,000,000
1987	1,000,000,000 (1 GigaFlop/s, GFlop/s)
1992	10,000,000,000
1993	100,000,000,000
1997	1,000,000,000,000 (1 TeraFlop/s, TFlop/s)
2000	10,000,000,000,000
2005	280,000,000,000,000 (280 Tflop/s)

TOP500 superCOMPUTER

H. Meuer, H. Simon, E. Strohmaier, & JD

- Listing of the 500 most powerful Computers in the World
- Yardstick: Rmax from LINPACK MPP

  Ax=b, dense problem



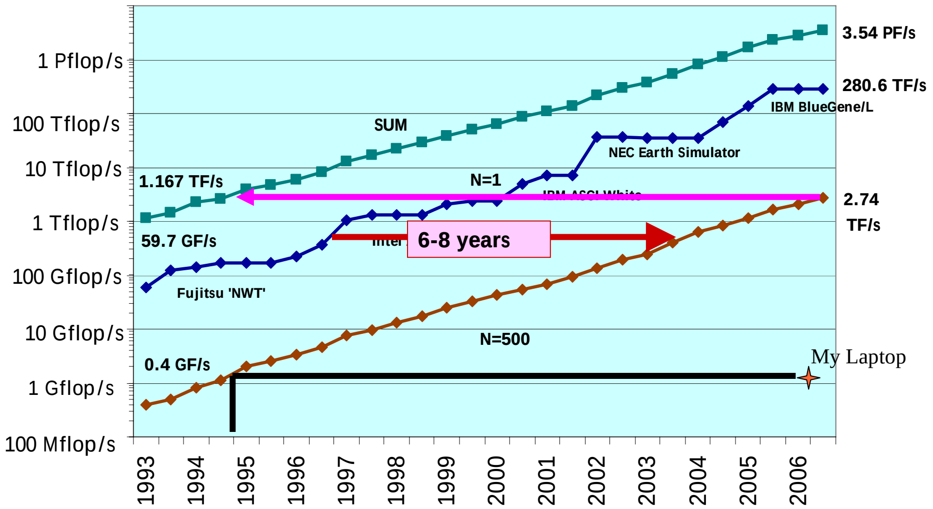
- Updated twice a year

  SC‘xy in the States in November

  Meeting in Germany in June

- All data available from **www.top500.org**

Performance Development; Top500

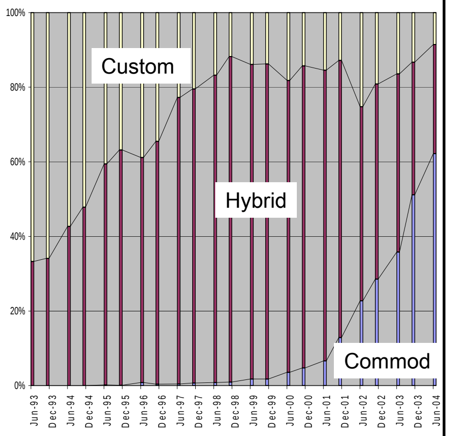


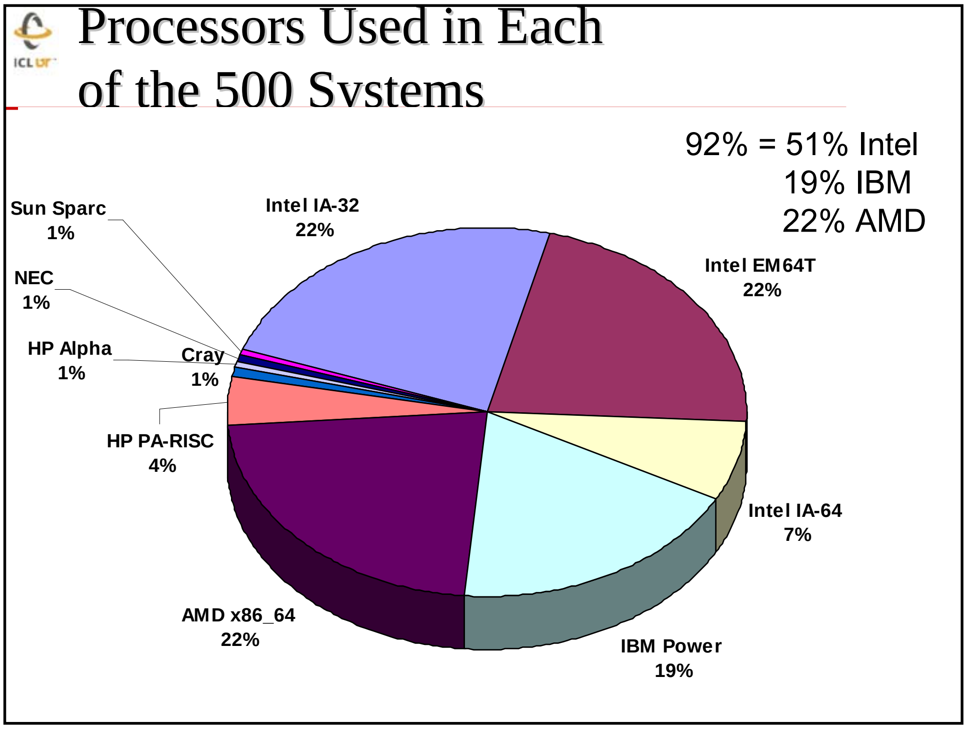
Architecture/Systems Continuum

Tightly Coupled

- Custom processor with custom interconnect
  - Cray X1
  - NEC SX-8
  - IBM Regatta
  - IBM Blue Gene/L
- Commodity processor with custom interconnect
  - SGI Altix
    - Intel Itanium 2
  - Cray XT3
    - AMD Opteron
- Commodity processor with commodity interconnect
  - Clusters
    - Pentium, Itanium, Opteron, Alpha
    - GigE, Infiniband, Myrinet, Quadrics
  - NEC TX7
  - IBM eServer
  - Dawning

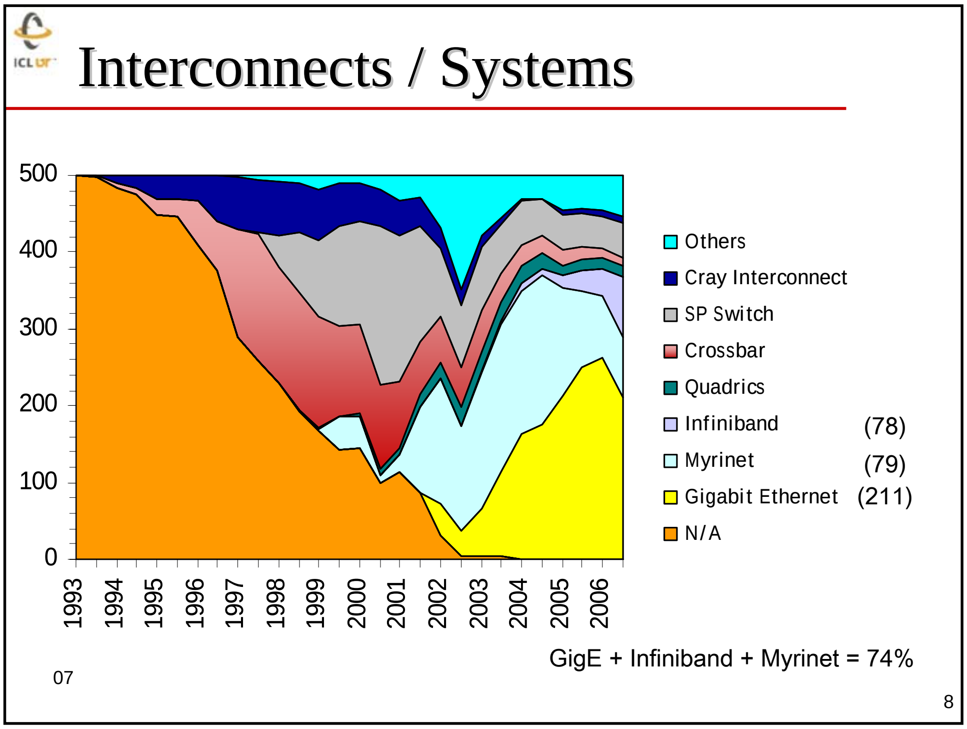
Loosely Coupled





Processors Used in Each of the 500 Svstems

92% = 51% Intel
19% IBM
22% AMD

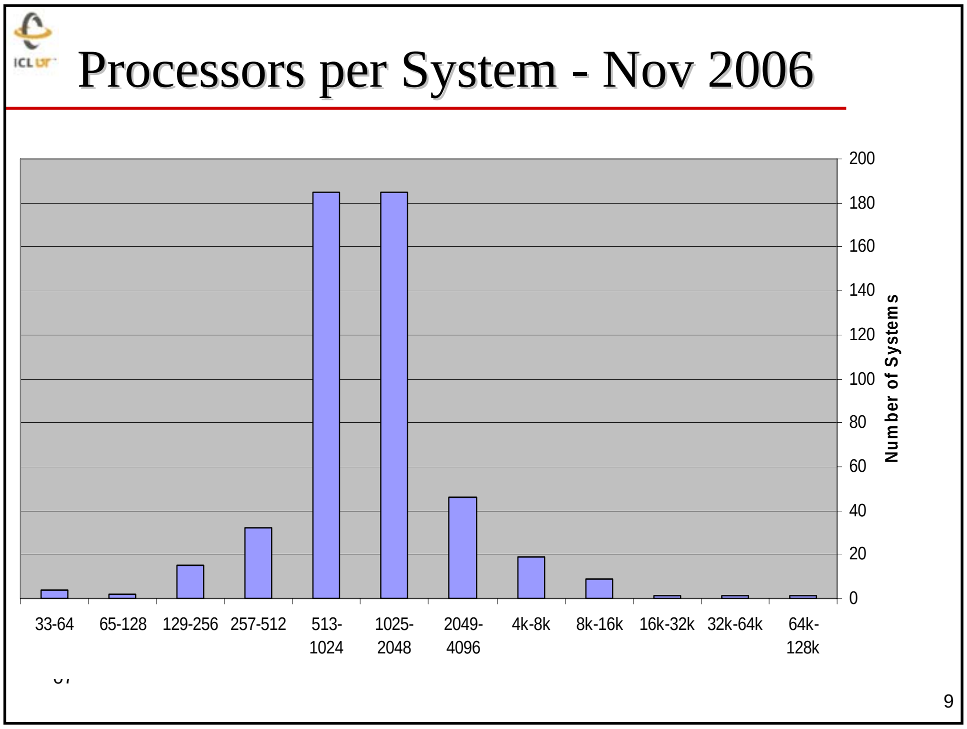


Interconnects / Systems

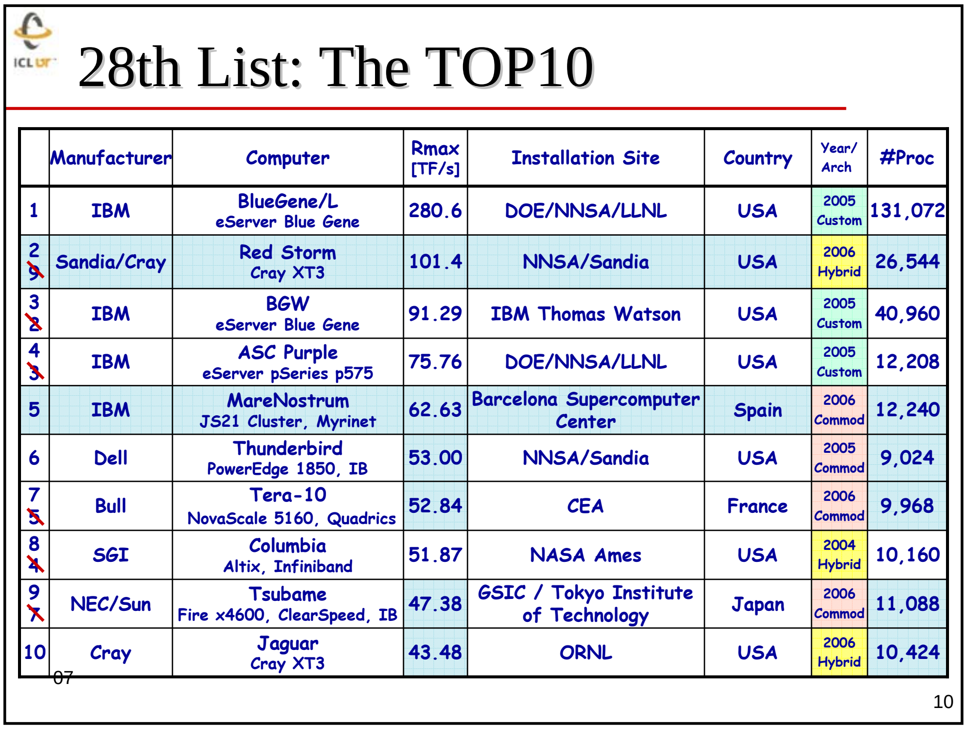
GigE + Infiniband + Myrinet = 74%

07

8

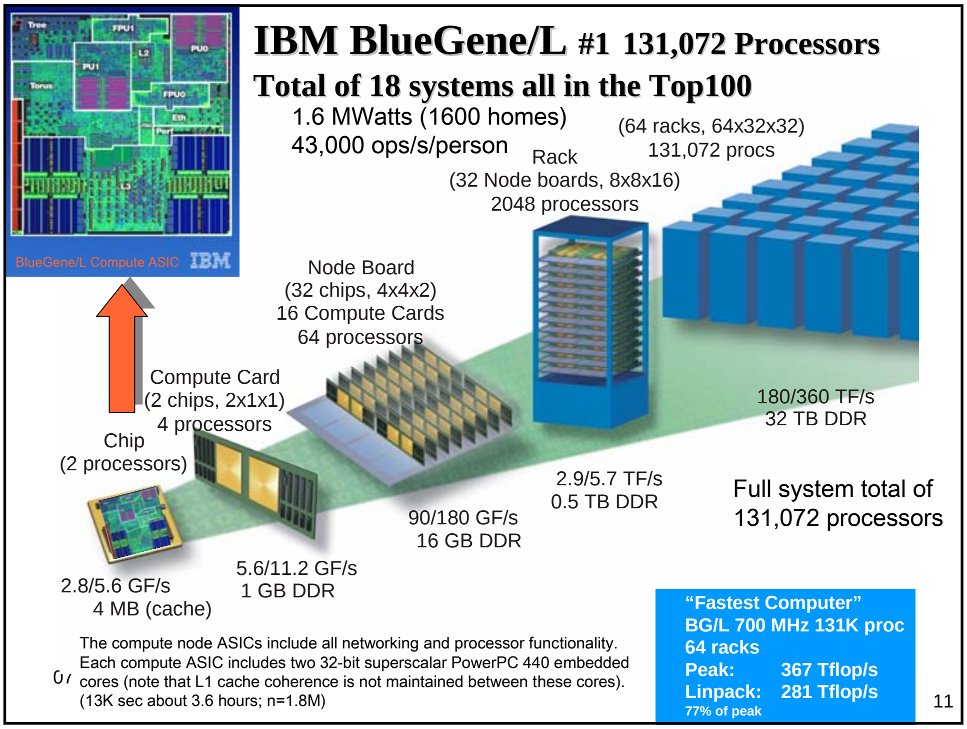


Processors per System - Nov 2006



28th List: The TOP10

	Manufacturer	Computer	Rmax [TF/s]	Installation Site	Country	Year/ Arch	#Proc
1	IBM	BlueGene/L eServer Blue Gene	280.6	DOE/NNSA/LLNL	USA	2005 Custom	131,072
2 ~~9~~	Sandia/Cray	Red Storm Cray XT3	101.4	NNSA/Sandia	USA	2006 Hybrid	26,544
3 ~~2~~	IBM	BGW eServer Blue Gene	91.29	IBM Thomas Watson	USA	2005 Custom	40,960
4 ~~3~~	IBM	ASC Purple eServer pSeries p575	75.76	DOE/NNSA/LLNL	USA	2005 Custom	12,208
5	IBM	MareNostrum JS21 Cluster, Myrinet	62.63	Barcelona Supercomputer Center	Spain	2006 Commod	12,240
6	Dell	Thunderbird PowerEdge 1850, IB	53.00	NNSA/Sandia	USA	2005 Commod	9,024
7 ~~5~~	Bull	Tera-10 NovaScale 5160, Quadrics	52.84	CEA	France	2006 Commod	9,968
8 ~~4~~	SGI	Columbia Altix, Infiniband	51.87	NASA Ames	USA	2004 Hybrid	10,160
9 ~~7~~	NEC/Sun	Tsubame Fire x4600, ClearSpeed, IB	47.38	GSIC / Tokyo Institute of Technology	Japan	2006 Commod	11,088
10	Cray	Jaguar Cray XT3	43.48	ORNL	USA	2006 Hybrid	10,424



IBM BlueGene/L #1 131,072 Processors

Total of 18 systems all in the Top100

1.6 MWatts (1600 homes)
43,000 ops/s/person

(64 racks, 64x32x32)
131,072 procs

Rack
(32 Node boards, 8x8x16)
2048 processors

Node Board
(32 chips, 4x4x2)
16 Compute Cards
64 processors

Compute Card
(2 chips, 2x1x1)
4 processors

Chip
(2 processors)

180/360 TF/s
32 TB DDR

2.9/5.7 TF/s
0.5 TB DDR

90/180 GF/s
16 GB DDR

5.6/11.2 GF/s
1 GB DDR

2.8/5.6 GF/s
4 MB (cache)

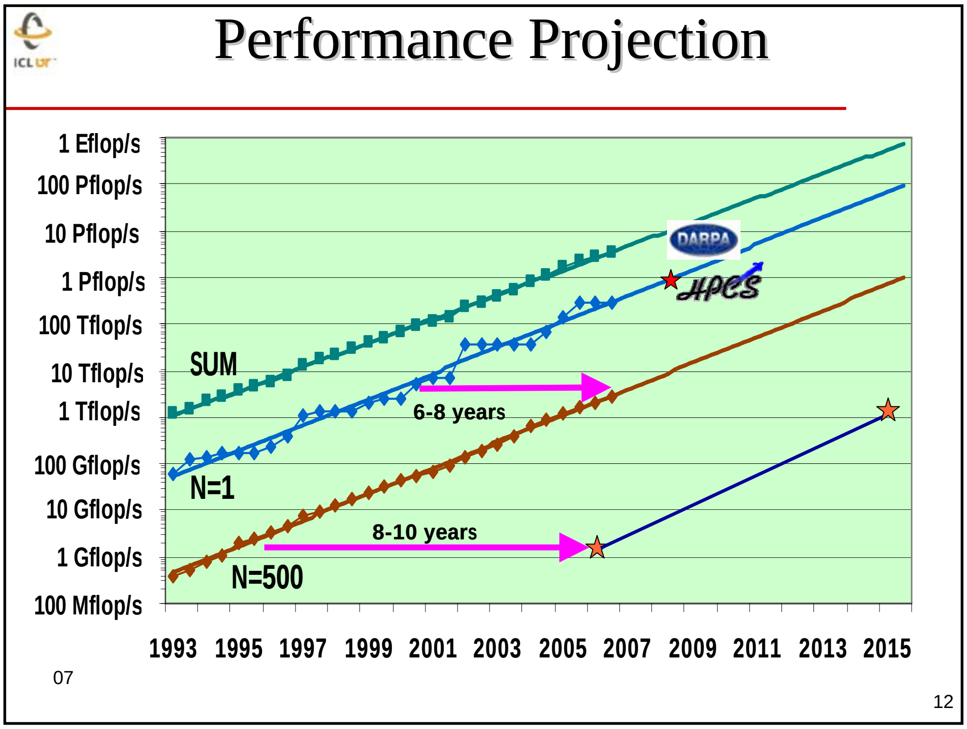
Full system total of
131,072 processors

The compute node ASICs include all networking and processor functionality. Each compute ASIC includes two 32-bit superscalar PowerPC 440 embedded cores (note that L1 cache coherence is not maintained between these cores). (13K sec about 3.6 hours; n=1.8M)

"Fastest Computer"
BG/L 700 MHz 131K proc
64 racks
Peak: 367 Tflop/s
Linpack: 281 Tflop/s
77% of peak

07

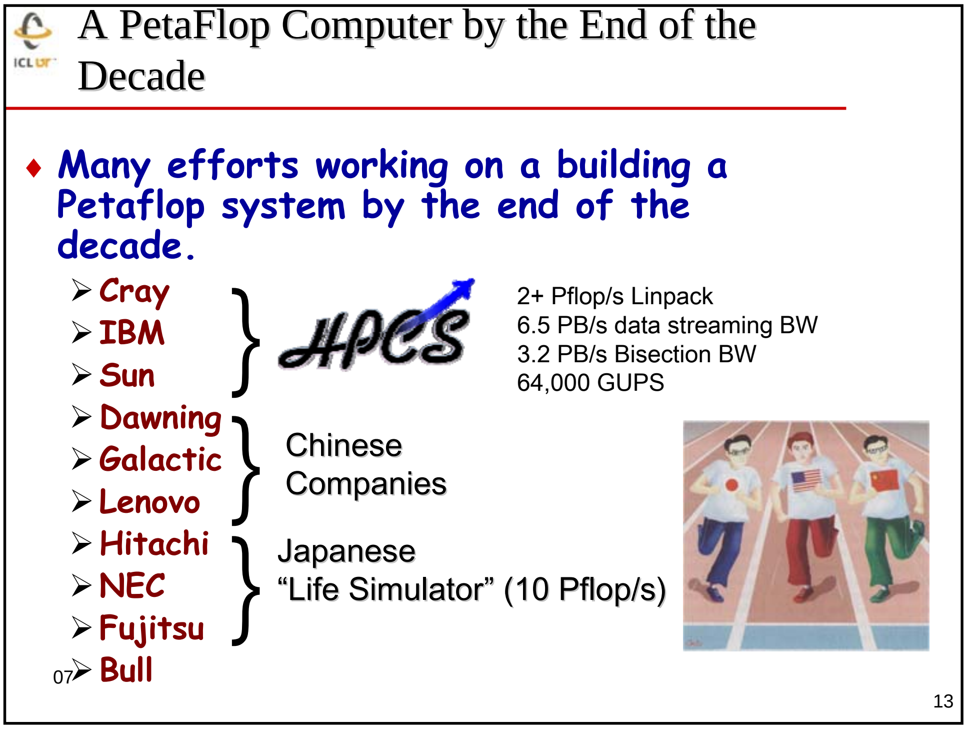
11



Performance Projection

07

12



A PetaFlop Computer by the End of the Decade

- Many efforts working on a building a Petaflop system by the end of the decade.
  - Cray
  - IBM
  - Sun
  - Dawning
  - Galactic
  - Lenovo
  - Hitachi
  - NEC
  - Fujitsu
  - Bull

Cray, IBM, Sun: HPCS

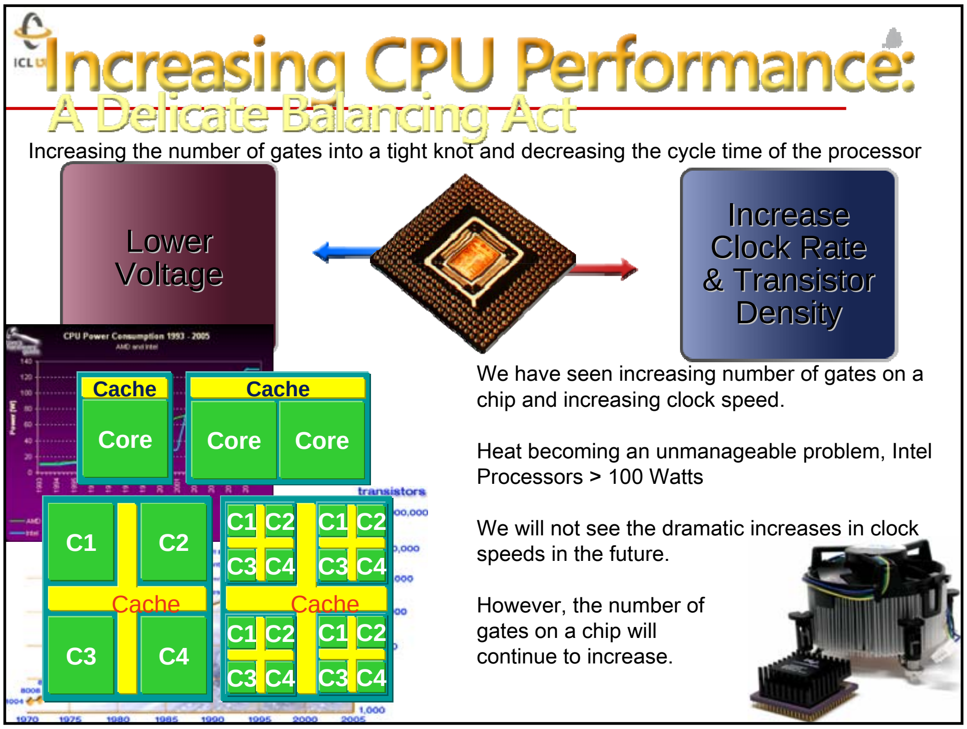
2+ Pflop/s Linpack
6.5 PB/s data streaming BW
3.2 PB/s Bisection BW
64,000 GUPS

Dawning, Galactic, Lenovo: Chinese Companies

Hitachi, NEC, Fujitsu: Japanese "Life Simulator" (10 Pflop/s)

07

13



Increasing CPU Performance: A Delicate Balancing Act

Increasing the number of gates into a tight knot and decreasing the cycle time of the processor

Lower Voltage
Increase Clock Rate & Transistor Density
CPU Power Consumption 1993 - 2005
AMD and Intel
Power (W)
140
120
100
80
60
40
20
0
AMD
Intel
Cache
Core
Cache
Core
Core
C1
C2
Cache
C3
C4
C1
C2
C3
C4
C1
C2
C3
C4
Cache
C1
C2
C3
C4
C1
C2
C3
C4
transistors
1,000
1970
1975
1980
1985
1990
1995
2000
2005

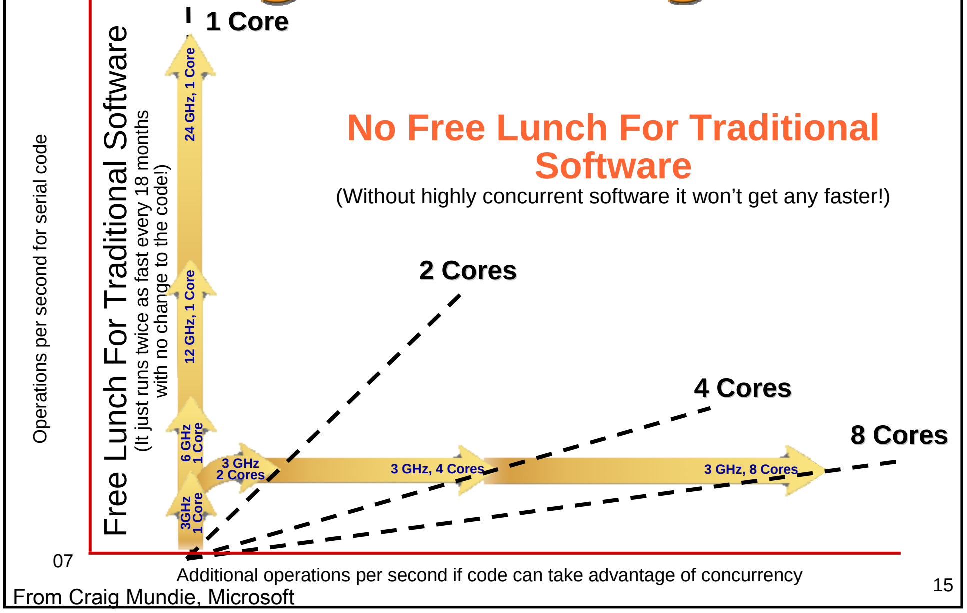
We have seen increasing number of gates on a chip and increasing clock speed.

Heat becoming an unmanageable problem, Intel Processors > 100 Watts

We will not see the dramatic increases in clock speeds in the future.

However, the number of gates on a chip will continue to increase.

Change Is Coming

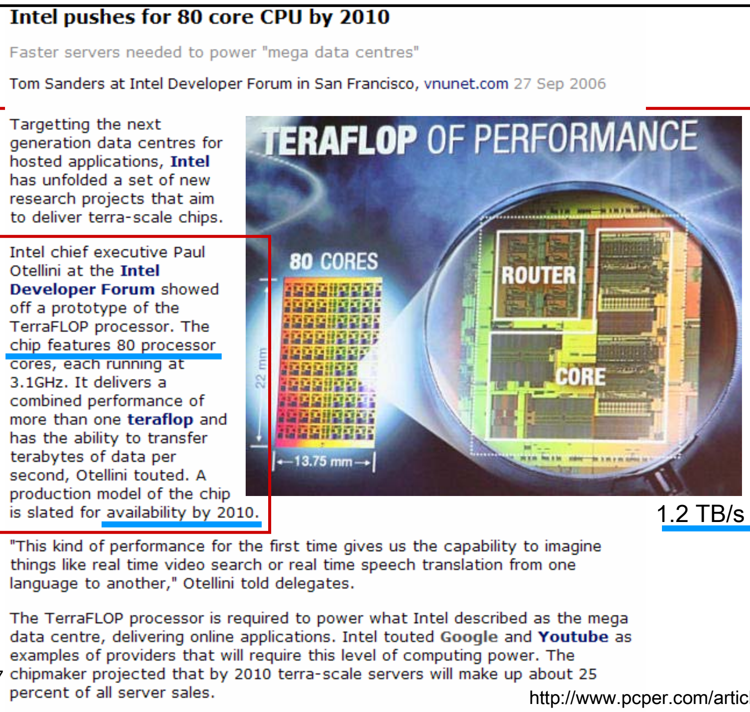


No Free Lunch For Traditional Software

(Without highly concurrent software it won't get any faster!)

07

From Craig Mundie, Microsoft



Intel pushes for 80 core CPU by 2010

Faster servers needed to power "mega data centres"

Tom Sanders at Intel Developer Forum in San Francisco, vnunet.com 27 Sep 2006

Targetting the next generation data centres for hosted applications, **Intel** has unfolded a set of new research projects that aim to deliver terra-scale chips.

Intel chief executive Paul Otellini at the **Intel Developer Forum** showed off a prototype of the TerraFLOP processor. The chip features 80 processor cores, each running at 3.1GHz. It delivers a combined performance of more than one **teraflop** and has the ability to transfer terabytes of data per second, Otellini touted. A production model of the chip is slated for availability by 2010.

1.2 TB/s memory BW

"This kind of performance for the first time gives us the capability to imagine things like real time video search or real time speech translation from one language to another," Otellini told delegates.

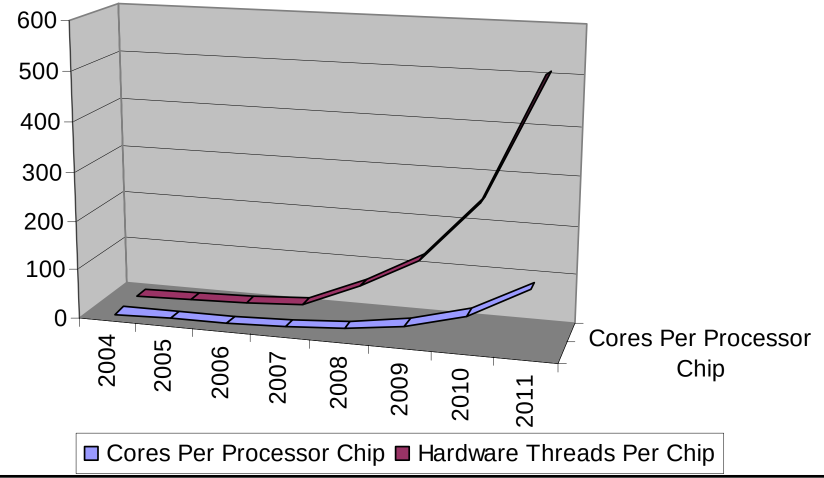
The TerraFLOP processor is required to power what Intel described as the mega data centre, delivering online applications. Intel touted **Google** and **Youtube** as examples of providers that will require this level of computing power. The chipmaker projected that by 2010 terra-scale servers will make up about 25 percent of all server sales.

07

http://www.pcper.com/article.php?aid=302

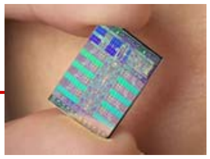
CPU Desktop Trends 2004-2011

- Relative processing power will continue to double every 18 months
- 5 years from now: 128 cores/chip w/512 logical processes per chip



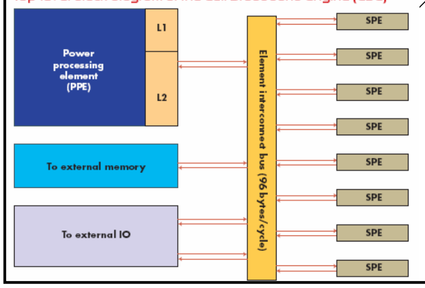
07

And Along Came the PlayStation 3



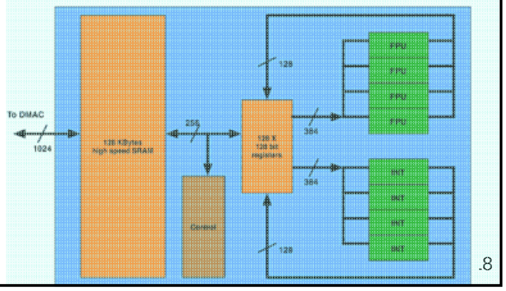
- The PlayStation 3's CPU based on a "Cell" processor
- Each Cell contains 8 APUs.
  - An SPE is a self contained vector processor which acts independently from the others.
  - 4 floating point units capable of a total of 25 Gflop/s (5 Gflop/s each @ 3.2 GHz)
  - 204 Gflop/s peak! 32 bit floating point; 64 bit floating point at 15 Gflop/s.
  - IEEE format, but only rounds toward zero in 32 bit, overflow set to largest
    - According to IBM, the SPE's double precision unit is fully IEEE854 compliant.

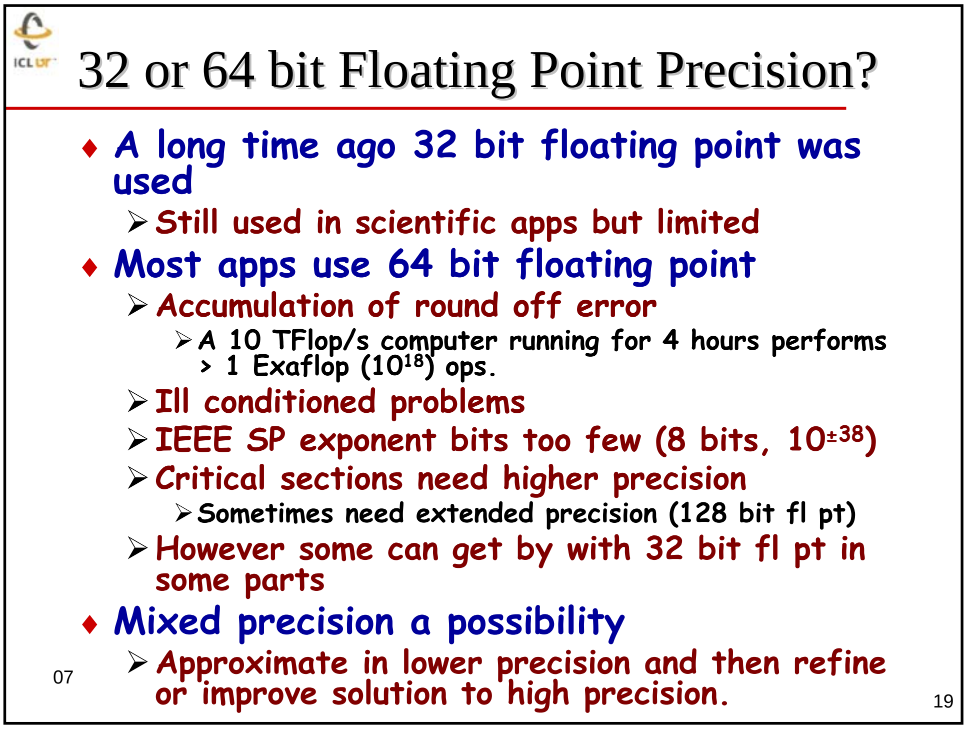
Top-level block diagram of the Cell Broadband Engine (CBE)


Power processing element (PPE)
L1
L2
To external memory
To external IO
Element interconnect bus (96 bytes/cycle)
SPE
SPE
SPE
SPE
SPE
SPE
SPE
SPE

Cell APU Architecture

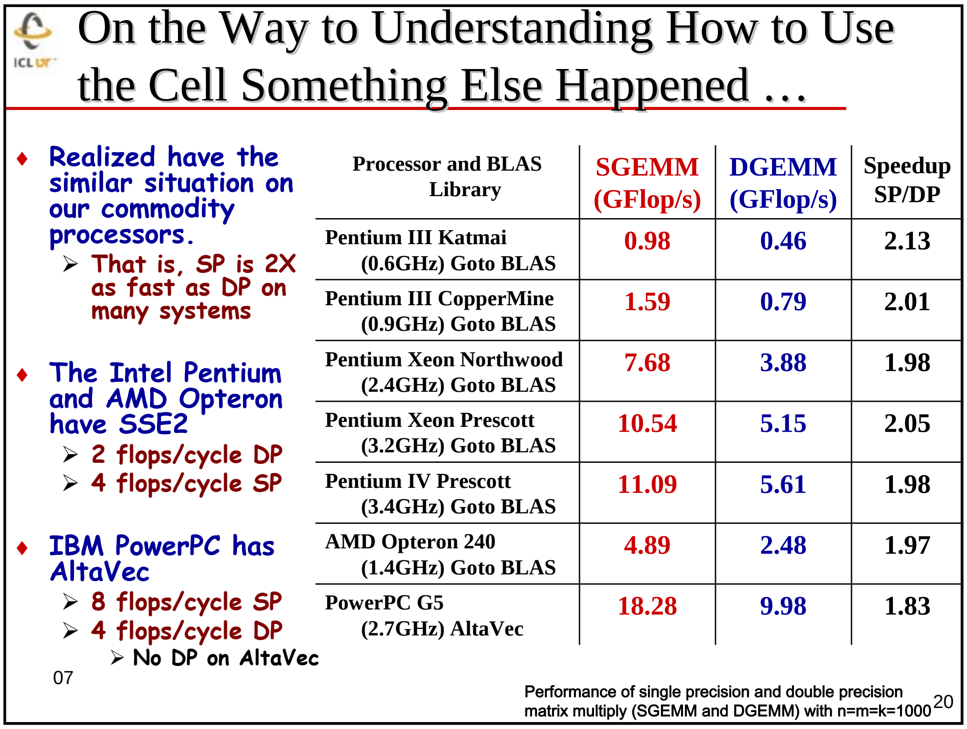
Each APU is an independent vector CPU capable of 32 GFLOPs or 32 GOPs.





32 or 64 bit Floating Point Precision?

- A long time ago 32 bit floating point was used
  - Still used in scientific apps but limited
- Most apps use 64 bit floating point
  - Accumulation of round off error
    - A 10 TFlop/s computer running for 4 hours performs > 1 Exaflop (10^{18}) ops.
  - Ill conditioned problems
  - IEEE SP exponent bits too few (8 bits, $10^{\pm 38}$)
  - Critical sections need higher precision
    - Sometimes need extended precision (128 bit fl pt)
  - However some can get by with 32 bit fl pt in some parts
- Mixed precision a possibility
  - Approximate in lower precision and then refine or improve solution to high precision.

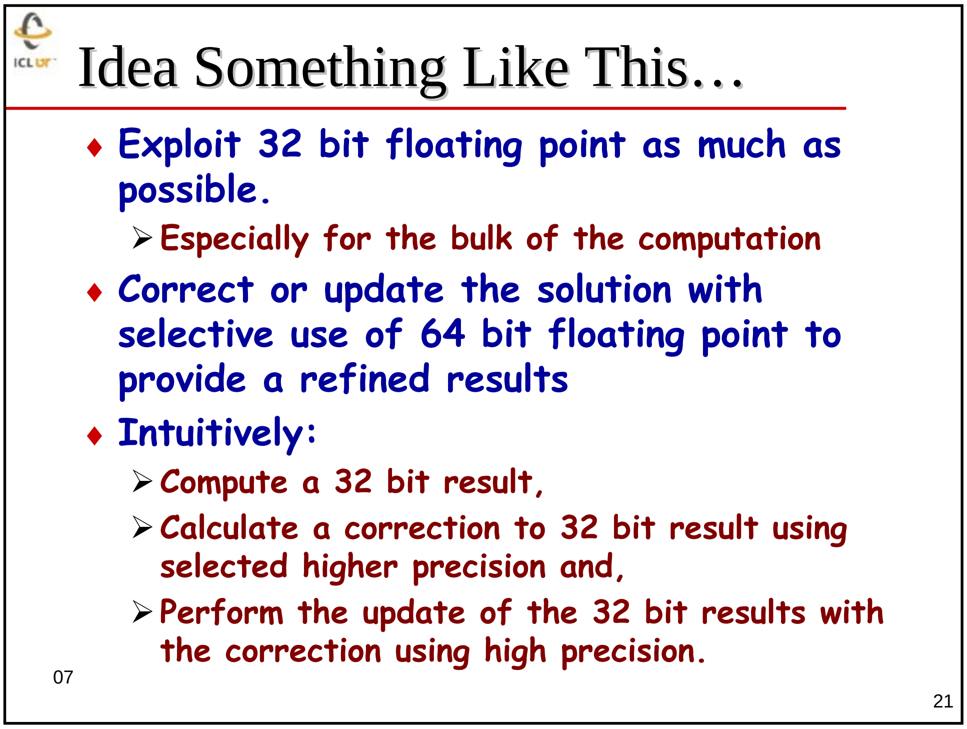


On the Way to Understanding How to Use the Cell Something Else Happened …

- Realized have the similar situation on our commodity processors.
  - That is, SP is 2X as fast as DP on many systems
- The Intel Pentium and AMD Opteron have SSE2
  - 2 flops/cycle DP
  - 4 flops/cycle SP
- IBM PowerPC has AltaVec
  - 8 flops/cycle SP
  - 4 flops/cycle DP
    - No DP on AltaVec

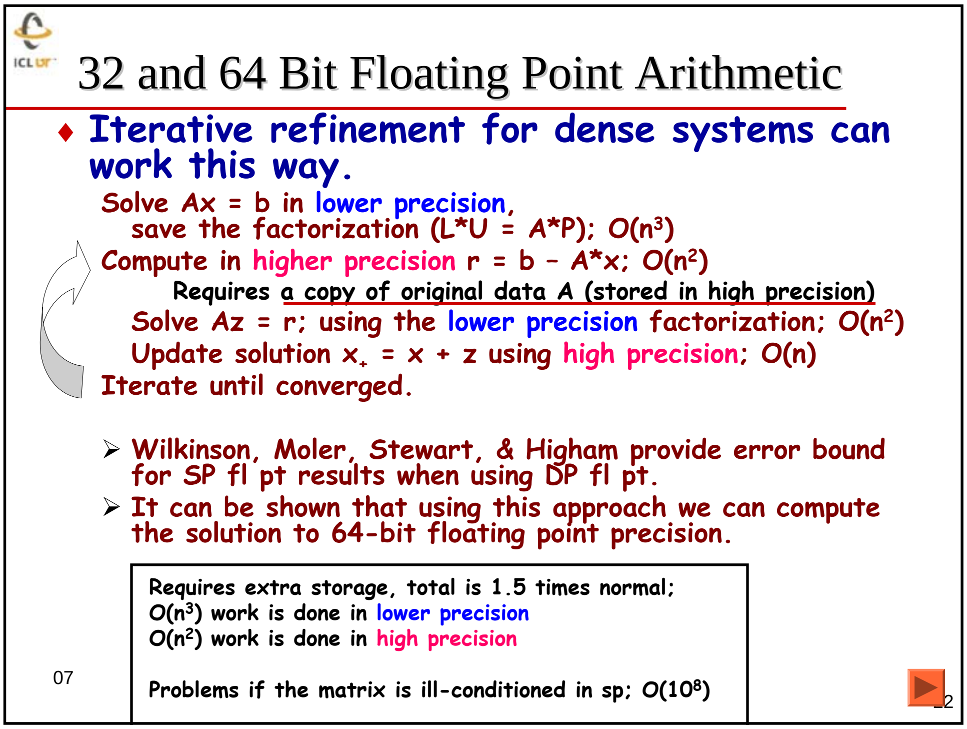
Processor and BLAS Library	SGEMM (GFlop/s)	DGEMM (GFlop/s)	Speedup SP/DP
Pentium III Katmai (0.6GHz) Goto BLAS	0.98	0.46	2.13
Pentium III CopperMine (0.9GHz) Goto BLAS	1.59	0.79	2.01
Pentium Xeon Northwood (2.4GHz) Goto BLAS	7.68	3.88	1.98
Pentium Xeon Prescott (3.2GHz) Goto BLAS	10.54	5.15	2.05
Pentium IV Prescott (3.4GHz) Goto BLAS	11.09	5.61	1.98
AMD Opteron 240 (1.4GHz) Goto BLAS	4.89	2.48	1.97
PowerPC G5 (2.7GHz) AltaVec	18.28	9.98	1.83

Performance of single precision and double precision matrix multiply (SGEMM and DGEMM) with n=m=k=1000



Idea Something Like This…

- Exploit 32 bit floating point as much as possible.
  - Especially for the bulk of the computation
- Correct or update the solution with selective use of 64 bit floating point to provide a refined results
- Intuitively:
  - Compute a 32 bit result,
  - Calculate a correction to 32 bit result using selected higher precision and,
  - Perform the update of the 32 bit results with the correction using high precision.



32 and 64 Bit Floating Point Arithmetic

- Iterative refinement for dense systems can work this way.

Solve Ax = b in lower precision,
  save the factorization (L*U = A*P); $O(n^3)$
Compute in higher precision r = b – A*x; $O(n^2)$
  Requires a copy of original data A (stored in high precision)
  Solve Az = r; using the lower precision factorization; $O(n^2)$
  Update solution $x_+ = x + z$ using high precision; $O(n)$
Iterate until converged.

- Wilkinson, Moler, Stewart, & Higham provide error bound for SP fl pt results when using DP fl pt.
- It can be shown that using this approach we can compute the solution to 64-bit floating point precision.

Requires extra storage, total is 1.5 times normal;
$O(n^3)$ work is done in lower precision
$O(n^2)$ work is done in high precision

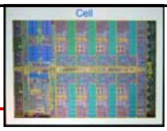
Problems if the matrix is ill-conditioned in sp; $O(10^8)$

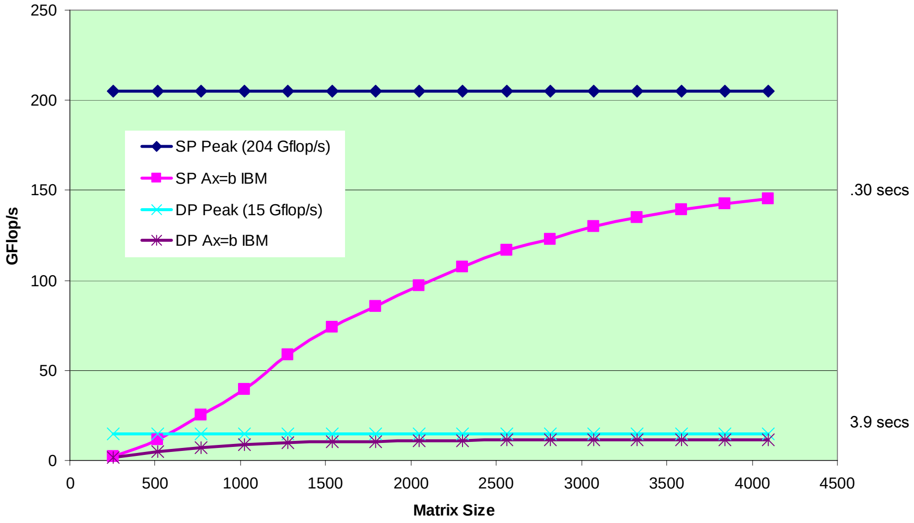
Speedups for Ax = b (Ratio of Times)

Architecture (BLAS)	*n*	DGEMM /SGEMM	DP Solve /SP Solve	DP Solve /Iter Ref	# iter
Intel Pentium III Coppermine (Goto)	3500	2.10	2.24	1.92	4
Intel Pentium IV Prescott (Goto)	4000	2.00	1.86	1.57	5
AMD Opteron (Goto)	4000	1.98	1.93	1.53	5
Sun UltraSPARC IIe (Sunperf)	3000	1.45	1.79	1.58	4
IBM Power PC G5 (2.7 GHz) (VecLib)	5000	2.29	2.05	1.24	5
Cray X1 (libsci)	4000	1.68	1.57	1.32	7
Compaq Alpha EV6 (CXML)	3000	0.99	1.08	1.01	4
IBM SP Power3 (ESSL)	3000	1.03	1.13	1.00	3
SGI Octane (ATLAS)	2000	1.08	1.13	0.91	4

Architecture (BLAS-MPI)	# procs	*n*	DP Solve /SP Solve	DP Solve /Iter Ref	# iter
AMD Opteron (Goto – OpenMPI MX)	32	22627	1.85	1.79	6
AMD Opteron (Goto – OpenMPI MX)	64	32000	1.90	1.83	6

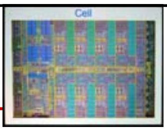
IBM Cell 3.2 GHz, Ax = b

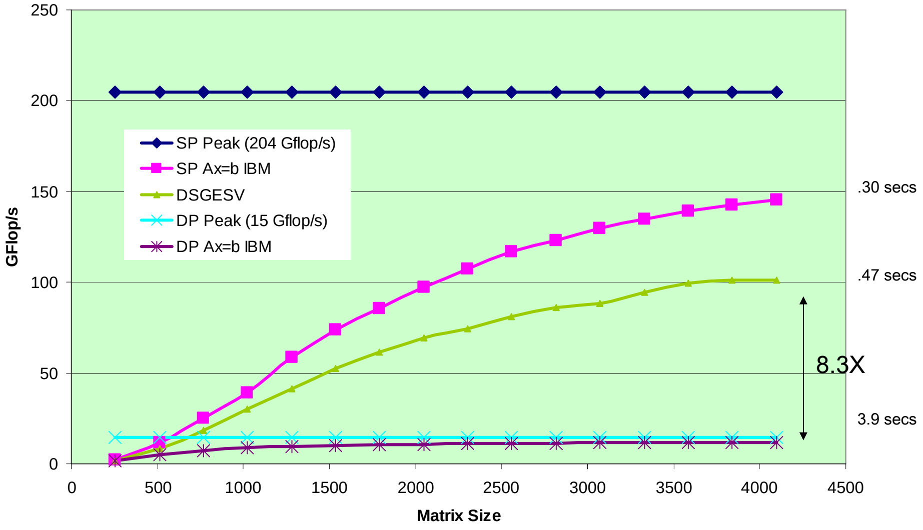




07

IBM Cell 3.2 GHz, Ax = b



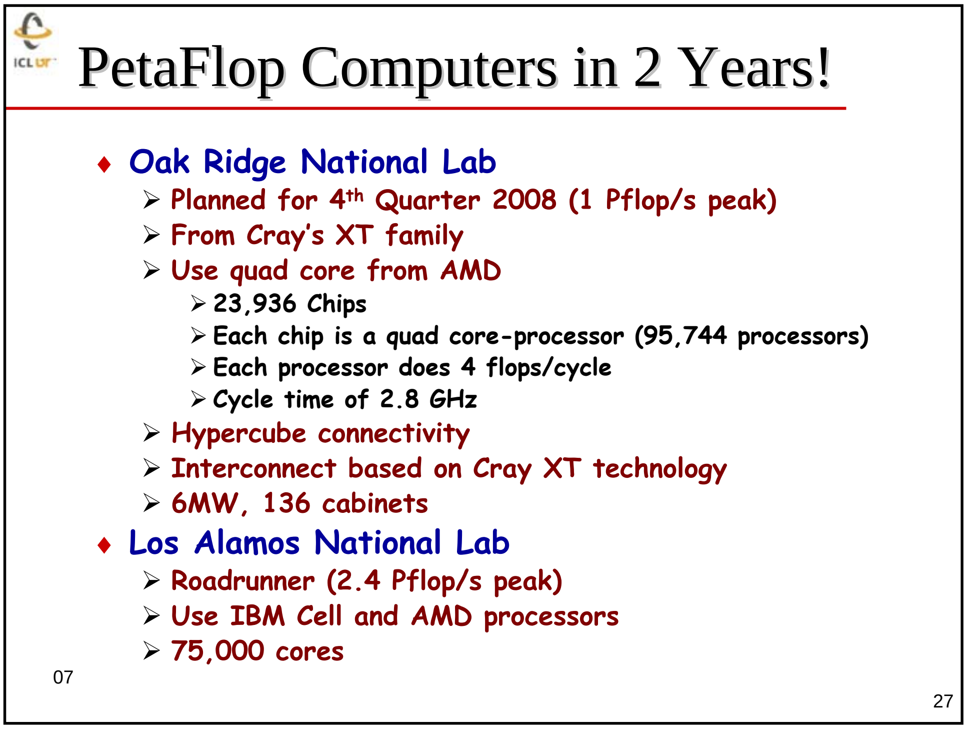


07

Refinement Technique Using Single/Double Precision

- **Linear Systems**
  - LU dense (in current release of LAPACK) and sparse
  - Cholesky
  - QR Factorization
- **Eigenvalue**
  - Symmetric eigenvalue problem
  - SVD
  - Same idea as with dense systems,
    - Reduce to tridiagonal/bi-diagonal in lower precision, retain original data and improve with iterative technique using the lower precision to solve systems and use higher precision to calculate residual with original data.
    - O(n^2) per value/vector
- **Iterative Linear System**
  - Relaxed GMRES
  - Inner/outer iteration scheme

07 See webpage for tech report which discusses this.

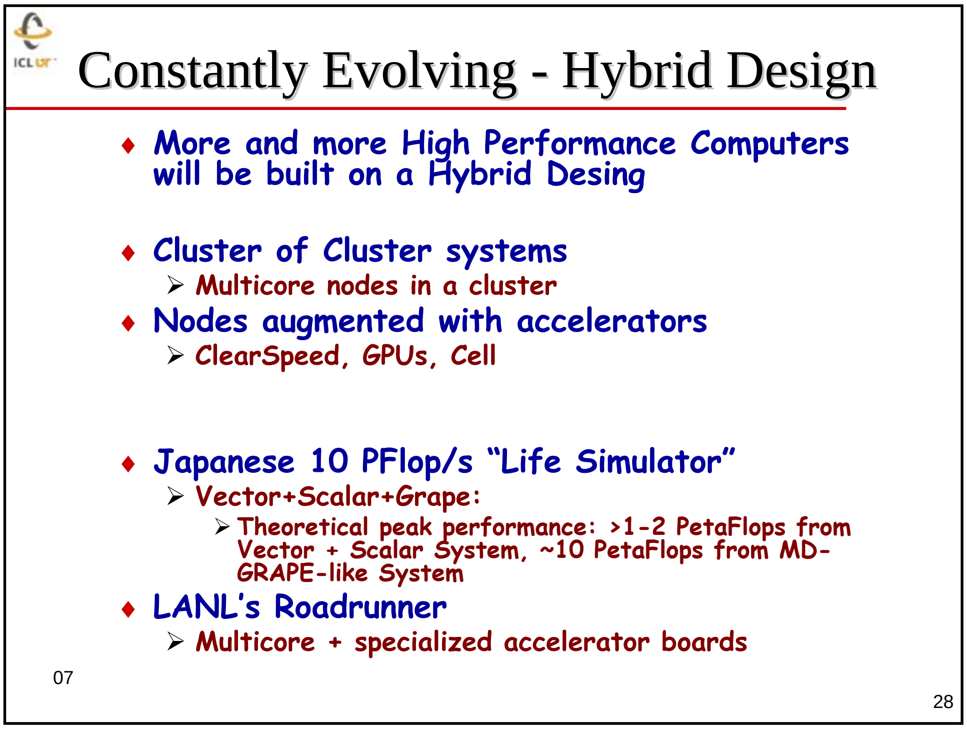


PetaFlop Computers in 2 Years!

- Oak Ridge National Lab
  - Planned for 4th Quarter 2008 (1 Pflop/s peak)
  - From Cray's XT family
  - Use quad core from AMD
    - 23,936 Chips
    - Each chip is a quad core-processor (95,744 processors)
    - Each processor does 4 flops/cycle
    - Cycle time of 2.8 GHz
  - Hypercube connectivity
  - Interconnect based on Cray XT technology
  - 6MW, 136 cabinets
- Los Alamos National Lab
  - Roadrunner (2.4 Pflop/s peak)
  - Use IBM Cell and AMD processors
  - 75,000 cores

07

27

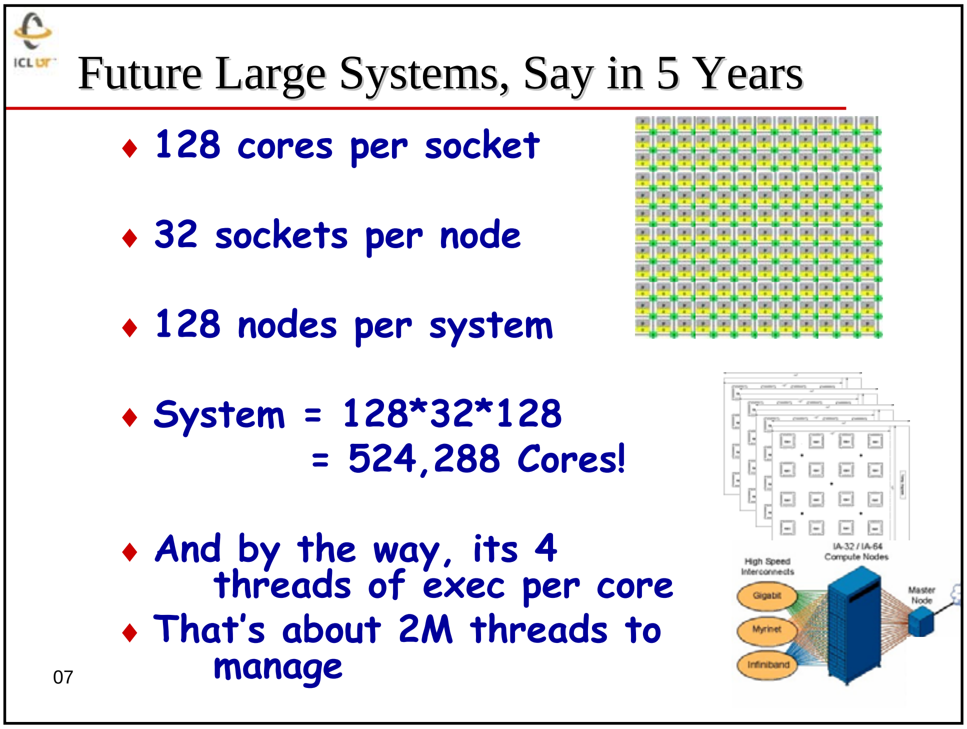


Constantly Evolving - Hybrid Design

- More and more High Performance Computers will be built on a Hybrid Desing
- Cluster of Cluster systems
  - Multicore nodes in a cluster
- Nodes augmented with accelerators
  - ClearSpeed, GPUs, Cell
- Japanese 10 PFlop/s "Life Simulator"
  - Vector+Scalar+Grape:
    - Theoretical peak performance: >1-2 PetaFlops from Vector + Scalar System, ~10 PetaFlops from MD-GRAPE-like System
- LANL's Roadrunner
  - Multicore + specialized accelerator boards

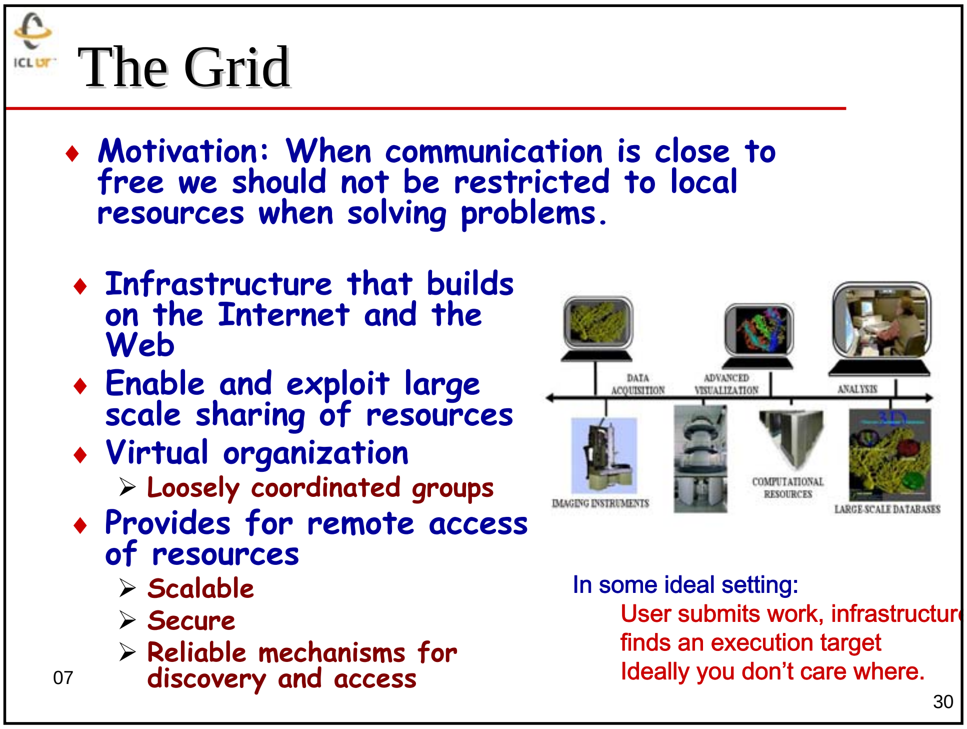
07

28



Future Large Systems, Say in 5 Years

- 128 cores per socket
- 32 sockets per node
- 128 nodes per system
- System = 128*32*128
  = 524,288 Cores!
- And by the way, its 4 threads of exec per core
- That's about 2M threads to manage



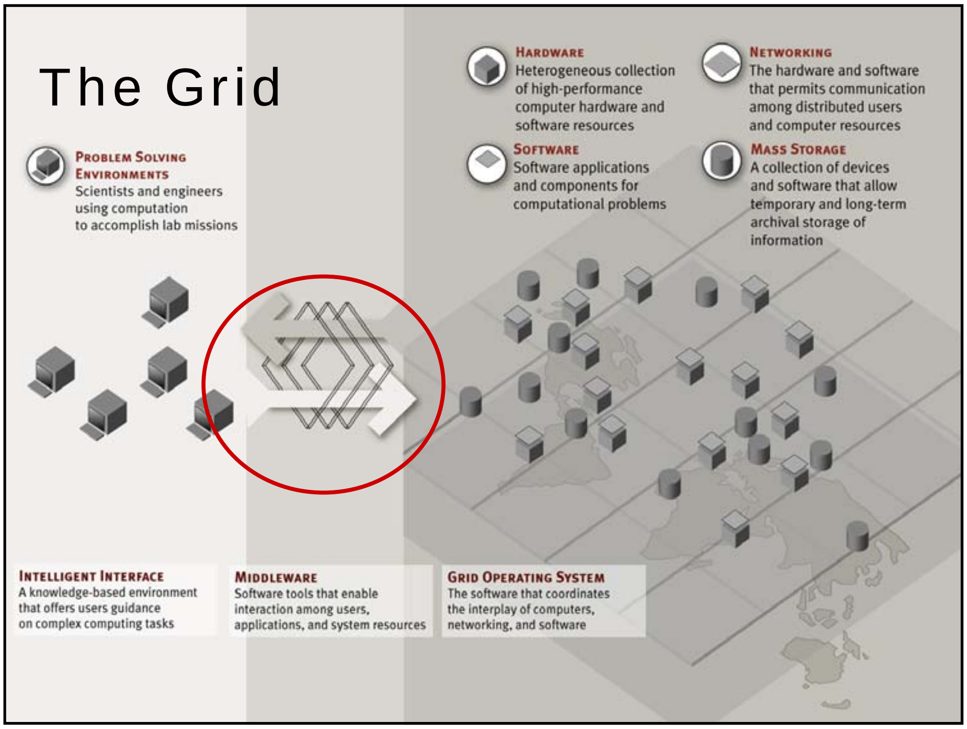
The Grid

- Motivation: When communication is close to free we should not be restricted to local resources when solving problems.

- Infrastructure that builds on the Internet and the Web
- Enable and exploit large scale sharing of resources
- Virtual organization
  - Loosely coordinated groups
- Provides for remote access of resources
  - Scalable
  - Secure
  - Reliable mechanisms for discovery and access

In some ideal setting:

User submits work, infrastructur
finds an execution target
Ideally you don't care where.



The Grid

Hardware
Heterogeneous collection of high-performance computer hardware and software resources

Networking
The hardware and software that permits communication among distributed users and computer resources

Software
Software applications and components for computational problems

Mass Storage
A collection of devices and software that allow temporary and long-term archival storage of information

Problem Solving Environments
Scientists and engineers using computation to accomplish lab missions

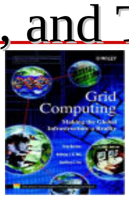
Intelligent Interface
A knowledge-based environment that offers users guidance on complex computing tasks

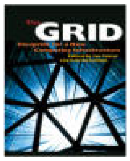
Middleware
Software tools that enable interaction among users, applications, and system resources

Grid Operating System
The software that coordinates the interplay of computers, networking, and software

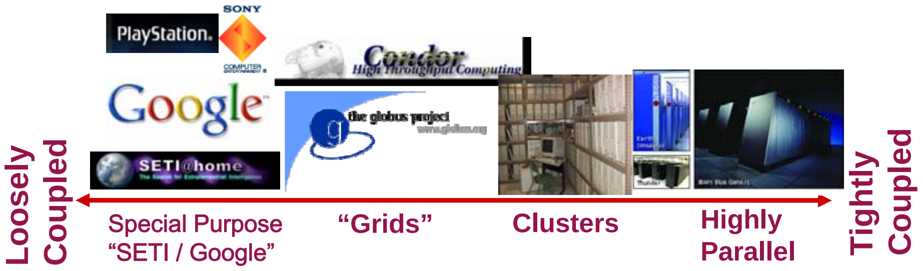
The Grid:
The Good, The Bad, and The Ugly

- **Good:**
  - Vision;
  - Community;
  - Developed functional software;
- **Bad:**
  - Oversold the grid concept;
  - Still too hard to use;
  - Underestimated the technical difficulties;
  - Point solution to apps
- **Ugly:**
  - *Authentication and security*
  - *Gap between hype and reality*





The Computing Continuum



Loosely Coupled

Special Purpose
"SETI / Google"

"Grids"

Clusters

Highly Parallel

Tightly Coupled

- Each strikes a different balance
  - computation/communication coupling
- Implications for execution efficiency
- *Applications for diverse needs*
  - *computing is only one part of the story!*

Grids vs. Capability vs. Cluster Computing

- Not an "either/or" question
  - Each addresses different needs
  - Each are part of an integrated solution
- Grid strengths
  - Coupling necessarily distributed resources
    - instruments, software, hardware, archives, and people
  - Eliminating time and space barriers
    - remote resource access and capacity computing
  - Grids are not a cheap substitute for capability HPC
- Highest performance computing strengths
  - Supporting foundational computations
    - terascale and petascale "nation scale" problems
  - Engaging tightly coupled computations and teams
- Clusters
  - Low cost, group solution
  - Potential hidden costs
- Key is easy access to resources in a transparent way

Future Directions and Issues

- ♦ Petaflops in 2 years not 4
- ♦ Multicore
  - ➢ Disruptive (think similar to what happened with distributed memory in the 90's)
  - ➢ Today 4 core/chip, 64 by end of decade, perhaps 1K in 2012
- ♦ Heterogeneous/Hybrid computing is returning
  - ➢ IBM Cell, GPUs, FPGAs, …
- ♦ Use of mixed precision for speed and delivery of full precision accuracy
  - ➢ IBM Cell, GPUs, FPGAs
- ♦ Fault Tolerance
  - ➢ Hundreds of thousands of processors
- ♦ Self adaptively in the software and algorithms
  - ➢ ATLAS like adaptation
- ♦ New languages
  - ➢ UPC, CAF, X10, Chapel, Fortress

Real Crisis With HPC Is With The Software

- ♦ Our ability to configure a hardware system capable of 1 PetaFlop (10^{15} ops/s) is without question just a matter of time and $$.

- ♦ A supercomputer application and software are usually much more long-lived than a hardware
  - ➢ Hardware life typically five years at most…. Apps 20-30 years
  - ➢ Fortran and C are the main programming models (still!!)

- ♦ The REAL CHALLENGE is Software
  - ➢ Programming hasn't changed since the 70's
  - ➢ HUGE manpower investment
    - ➢ MPI… is that all there is?
  - ➢ Often requires HERO programming
  - ➢ Investments in the entire software stack is required (OS, libs, etc.)

- ♦ Software is a major cost component of modern technologies.
  - ➢ The tradition in HPC system procurement is to assume that the software is free… SOFTWARE COSTS (over and over)

- ♦ What's needed is a long-term, balanced investment in the HPC Ecosystem: hardware, software, algorithms and applications.

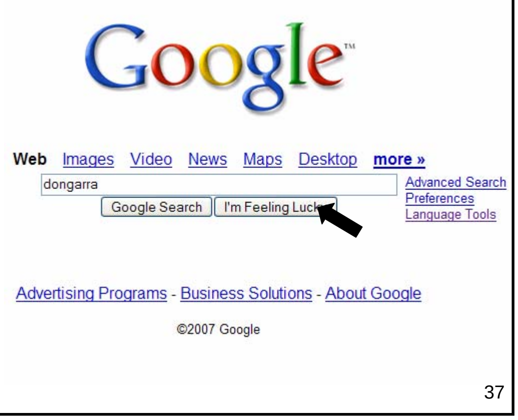
Collaborators / Support

- **Top500 Team**
  - Erich Strohmaier, NERSC
  - Hans Meuer, Mannheim
  - Horst Simon, NERSC

http://www.top500.org/

- **NetSolve**
  - Asim YarKhan, UTK
  - Keith Seymour, UTK
  - Zhiao Shi, UTK

Office of Science
U.S. DEPARTMENT OF ENERGY



dongarra

©2007 Google