ICCS 2003
Progress, prizes, & Community-centric Computing

Melbourne

June 3, 2003

Performance, Grids, and Communities

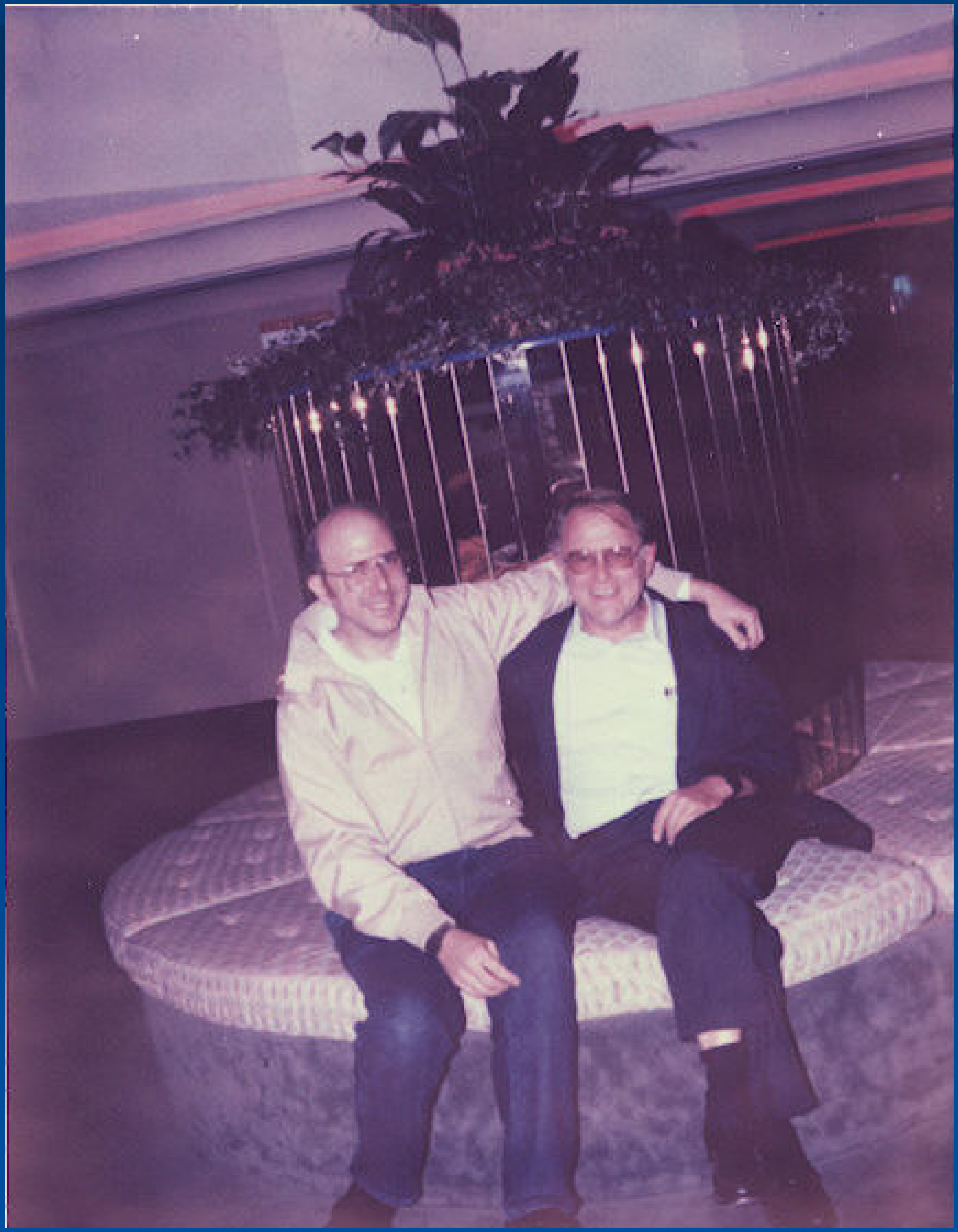
- Quest for parallelism
- Bell Prize winners past, present, and
- Future implications (or what do you bet on)

- Grids: web services are the challenge…
  not teragrids with ∞bw, 0 latency, & 0 cost

- Technology trends leading to
- Community Centric Computing versus centers

A *brief, simplified* history of HPC

1. *Cray* formula *smPv* evolves for *Fortran.* 60-02 (US:60-90)
2. 1978: VAXen threaten computer centers…
3. NSF response: Lax Report. Create 7-Cray centers 1982 –
4. 1982: The Japanese are coming: Japan's 5th Generation.)
5. SCI: DARPA search for parallelism with "killer" micros
6. Scalability found: "bet the farm" on micros clusters
   Users "adapt": MPI, lcd programming model found. >95
   Result: EVERYONE gets to re-write their code!!
7. Beowulf Clusters form by adopting PCs and Linus' Linux to create the cluster standard! (In spite of funders.)>1995
8. "Do-it-yourself" Beowulfs negate computer centers since everything is a cluster and shared power is nil! >2000.
9. ASCI: DOE's petaflops clusters => "arms" race continues!
10. High speed nets enable peer2peer & Grid or Teragrid
11. *Atkins Report: Spend $1.1B/year, form more and larger centers and connect them as a single center…*
12. 1997-2002: *SOMEONE* tell Fujitsu & NEC to get "in step"!
13. *2004: The Japanese came! GW Bush super response!*



Steve Squires & Gordon Bell at our "Cray" at the start of DARPA's SCI program c1984.

<u>20 years later: Clusters of Killer micros become the single standard</u>

- One Instruction Stream SISD
  - Hardwired, Minimal (MISC) *701, PDP-8, 8080*
  - Reduced, extensive pipeling (RISC) *801, MIPS, Sparc*
  - Complete/Complex (CISC) *360/370, VAX, 68K, 80x86*
  - Language-based (microprogrammed) *Symbolics,TI*
- Single Instruction Stream Multiple Data Operations (SIMD)
  - Fixed function units (Array Processors) *FPS, Analogic, CSPI* and Signal Processing chips *TI, Motorola*
  - Extra-Long Instruction-word *Multiflow*
  - Multiple, parallel execution units *CDC 6600*
  - Massive data parallelism *DAP, MPP, Connection Machine, GF11*
  - Pipelined, parallel execution *CDC 7600, 360/91*
  - Systolic Chip Cells (programmed pipelines) *WARP cell*
  - Supercomputers (Vector) *TI ASC, STAR, Cray 1, SX-2*
    - mini-super & micro-super *Convex C-1*
    - personal supers, one processor e.g. based on *Intel 80860*
- Multiprocessors MIMD (shared memory with micro- and multi-tasking)
  - Supercomputers (multi, vector proc.) *Cray XMP, ETA-10, SX3*
    - minisuper *Alliant, Convex C-2*
    - Graphic Super *Ardent*
  - 2, 4, 6 Processor Mainframes *IBM& BUNCH*
  - Functional Multi's *Multibus,VME-based micros*
  - The "Multi" (4-30) *Arete, DEC, Encore, Sequent, etc.*
  - Large "Multi" (>100) *RP3, E&S, Ultramax, Kendall Square*
  - Fault-Tolerant "Multi" *Stratus*
- Multicomputers MIMD (interconnected compu[ters,] no shared memory, communic[ation] via message passing)
  - High Availability *Tandem, Parallel, Teradata (tree)*
  - High Performance *Ncube, Ametek, Intel, Transputer, TF1*
  - LAN Clusters *Apollo, DEC, IBM PC, SUN environments*
  - Dataflow computers *Manchester and MIT Research computers*
  - Multiple cell, systolic arrays *WARP*

1987 Interview July 1987 as first CISE AD

- Kicked off parallel processing initiative with 3 paths
  - Vector processing *was totally ignored*
  - Message passing multicomputers including distributed workstations and clusters
  - smPs (multis) -- main line for programmability
  - *SIMDs might be low-hanging fruit*
- Kicked off Gordon Bell Prize
- Goal: common applications parallelism
  - 10x by 1992; 100x by 1997

IEEE Software launches annual Gordon Bell Award

Editor-in-Chief Ted Lewis has announced the First Annual Gordon Bell Award for the most improved speedup for parallel-processing applications. The two $1000 awards will be presented to the person or team that demonstrates the greatest speedup on a multiple-instruction, multiple-data parallel processor.

One award will be for most speedup on a general-purpose (multiapplication) MIMD processor, the other for most speedup on a special-purpose MIMD processor. Speedup can be accomplished by hardware or software improvements, or by a combination of the two.

To qualify for the 1987 awards, candidates must submit documentation of their results by Dec. 1. The winners will be announced in the March 1988 issue. This year's judges are Alan Karp of IBM's Palo Alto Scientific Center, Jack Dongarra of Argonne National Laboratory, and Ken Kennedy of Rice University.

For a complete set of rules, definitions, and submission guidelines, write to the Gordon Bell Award, *IEEE Software*, 10662 Los Vaqueros Cir., Los Alamitos, CA 90720.

Gordon Bell Prize announced
Computer July 1987

"In Dec. 1995 computers with 1,000 processors will do most of the scientific processing."

Danny Hillis
1990 (1 paper *or 1 company*)

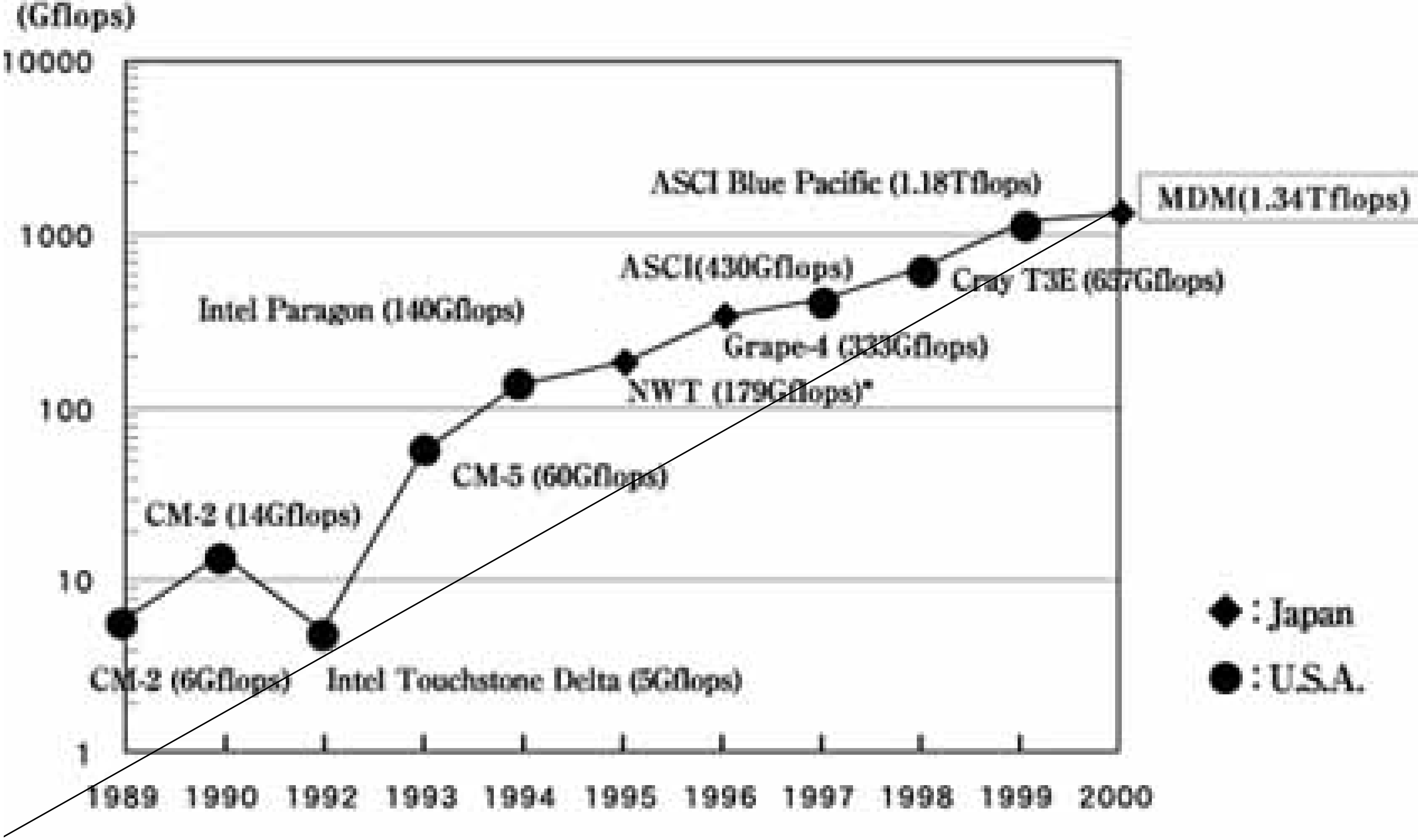
The Bell-Hillis Bet
Massive Parallelism in 1995

Applications

Revenue

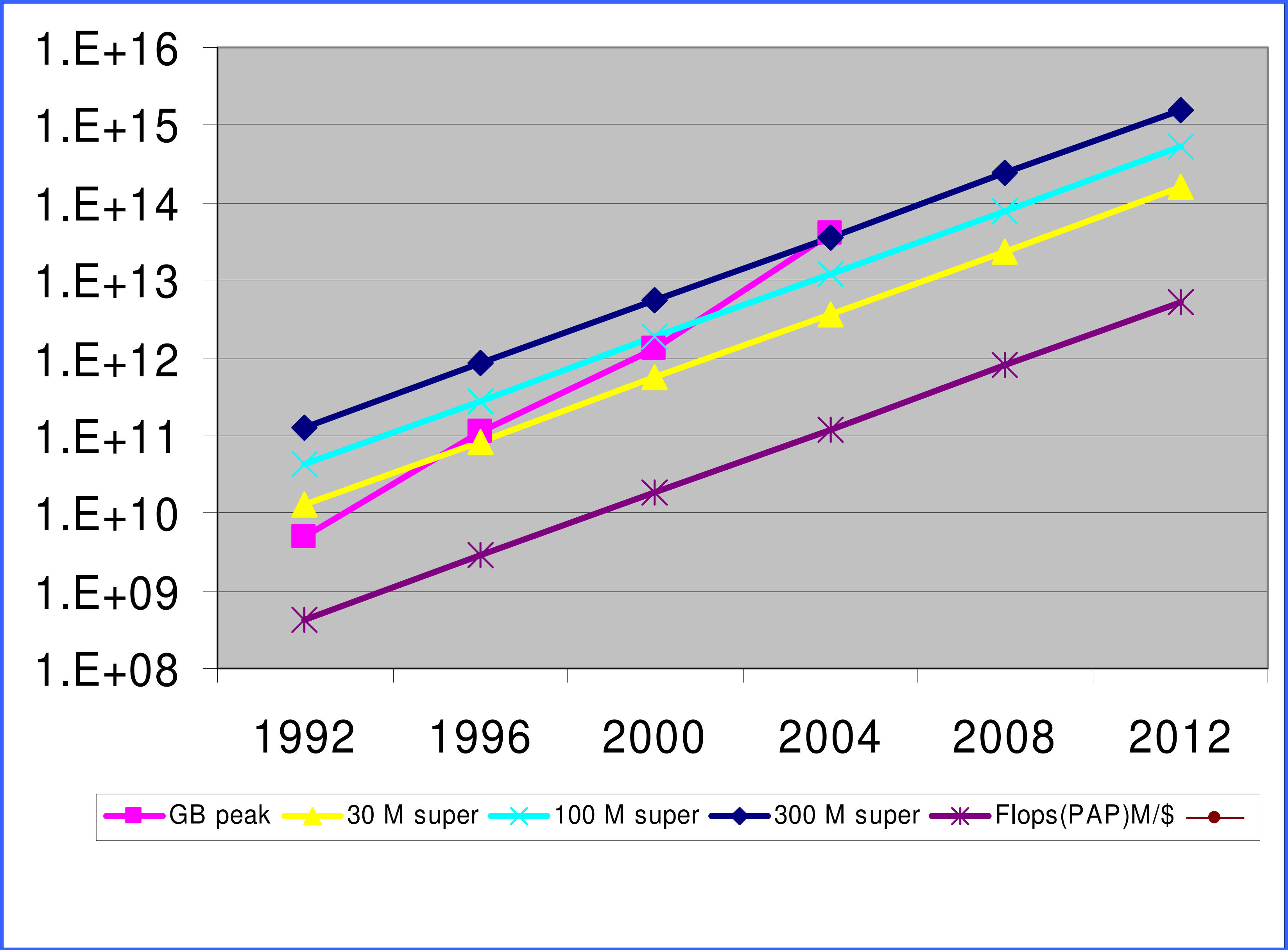
Petaflops / mo.

Trend of computing speed at Gordon Bell Prizes



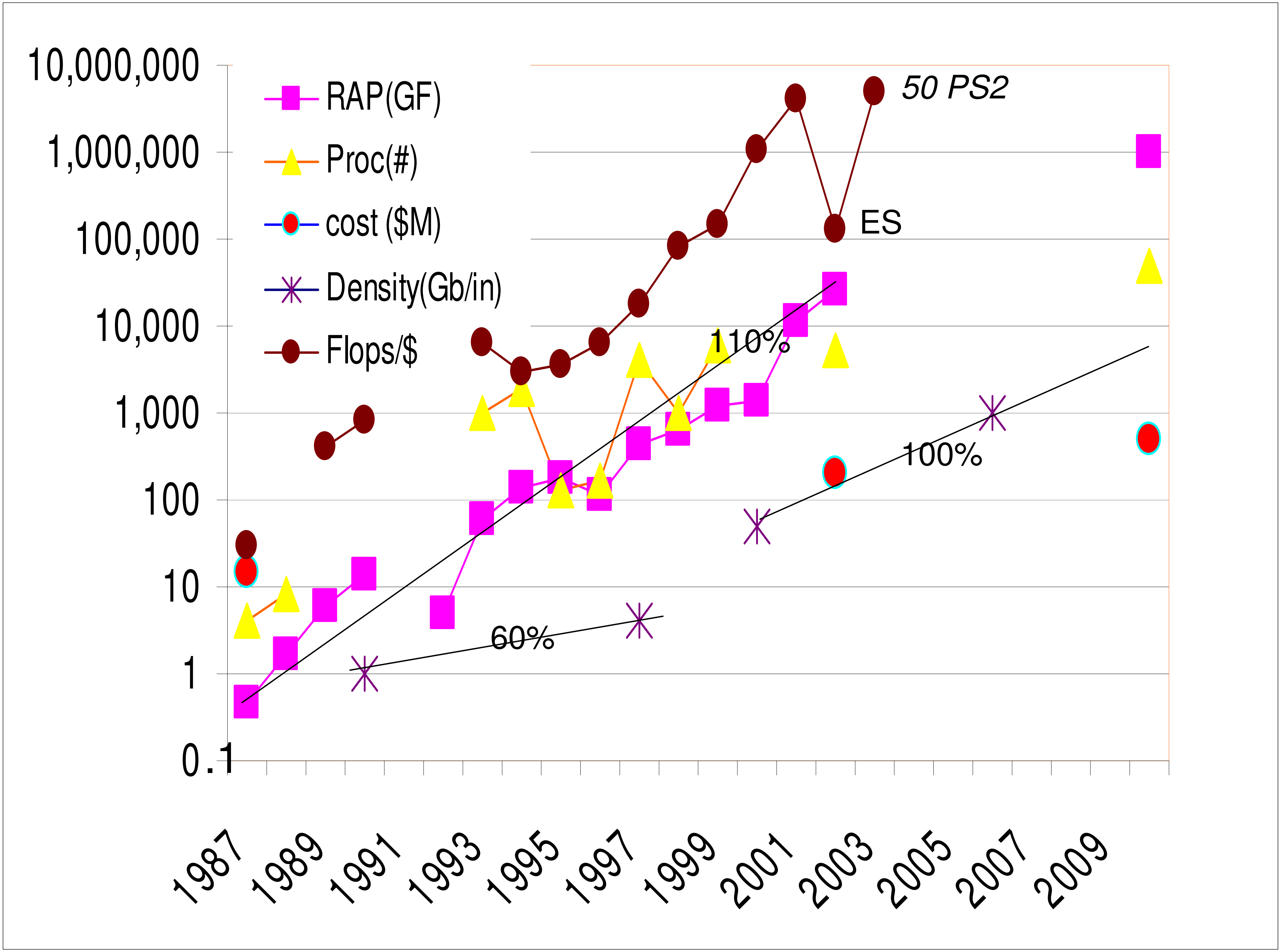
*NAL of STA

Perf (PAP) = c x 1.6**(t-1992); *c = 128 GF/$300M*
‘94 prediction: c = 128 GF/$30M



1987-2002 Bell Prize Performance Gain

- **26.58TF/0.000450TF = 59,000 in 15 years**
  $\underline{= 2.08^{15}}$
- **Cost increase \$15 M >> \$300 M? *say 20x***
- **Inflation was 1.57 X, so**
  effective spending increase 20/1.57 =12.73
- **59,000/12.73 = 4639 X**
  $= \underline{1.76^{15}}$
- **Price-performance 89-2002:**
  \$2500/MFlops > \$0.25/MFlops = 10^4
  $= \underline{2.04}^{13}$ ***\$1K/4GFlops PC = \$0.25/MFlops***



10,000,000
1,000,000
100,000
10,000
1,000
100
10
1
0.1
RAP(GF)
Proc(#)
cost ($M)
Density(Gb/in)
Flops/$
50 PS2
ES
110%
100%
60%
1987
1989
1991
1993
1995
1997
1999
2001
2003
2005
2007
2009

1987-2002 Bell Prize Performance Winners

- **Vector: Cray-XMP, -YMP, CM2* (2), Clustered: CM5, Intel 860 (2), Fujitsu (2), NEC (1) = 10**
- **Cluster of SMP (Constellation): IBM**
- **Cluster, single address, very fast net: Cray T3E**
- **Numa: SGI… good idea, but not universal**
- **Special purpose (2)**
- **No winner: 91**
- **By 1994, all were scalable (x,y,cm2)**
- **No x86 winners!**

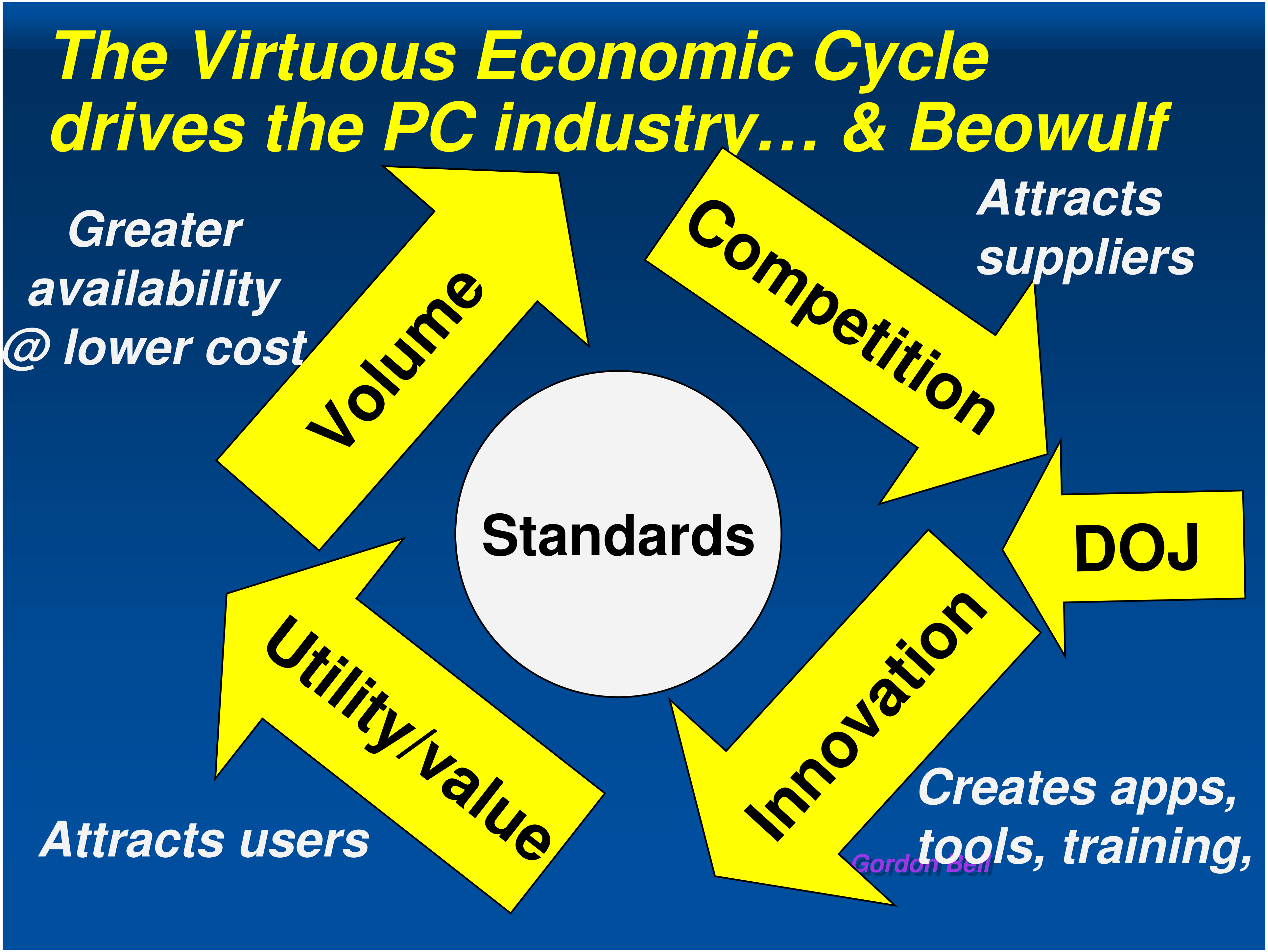
Heuristics

- Use dense matrices, or almost embarrassingly // apps
- Memory BW… you get what you pay for (4-8 Bytes/Flop)
- RAP/$ is constant. Cost of memory bandwidth is constant.
- Vectors will continue to be an essential ingredient;
  the low overhead formula to exploit the bandwidth, stupid
- SIMD a bad idea; No multi-threading yet… a bad idea?
- Fast networks or larger memories decrease inefficiency
- Specialization pays in performance/price
- 2003: 50 Sony workstations @6.5gflops for 50K is good.
- COTS aka x86 for Performance/Price BUT <u>not Perf</u>.
- Bottom Line:
  Memory BW, FLOPs, Interconnect BW <>Memory Size

Lessons from Beowulf

- **An experiment in parallel computing systems ‘92**
- **Established <u>vision</u>- low cost high end computing**
- **Demonstrated effectiveness of PC clusters for some (not all) classes of applications**
- **Provided networking software**
- **Provided cluster management tools**
- **Conveyed findings to broad community**
- **Tutorials and the book**
- **Provided design standard to rally community!**
- **<u>Standards beget: books, trained people, software … virtuous cycle that allowed apps to form</u>**
- **Industry began to form beyond a research project**

Courtesy, Thomas Sterling, Caltech.


The Virtuous Economic Cycle
drives the PC industry… & Beowulf
Greater
availability
@ lower cost
Volume
Competition
Attracts
suppliers
Standards
DOJ
Innovation
Creates apps,
tools, training,
Utility/value
Attracts users
Gordon Bell

Computer types

-------- Connectivity--------

	WAN/LAN	SAN	DSM	SM
vector	Netwrked Supers…	VPPuni	NEC mP	NEC super Cray X…T (all mPv)
Scalar-u	GRID Legion Condor Beowulf NT clusters	SP2(mP) NOW T3E	SGI DSM clusters & SGI DSM	Mainframes Multis WSs PCs

Clusters

PC+

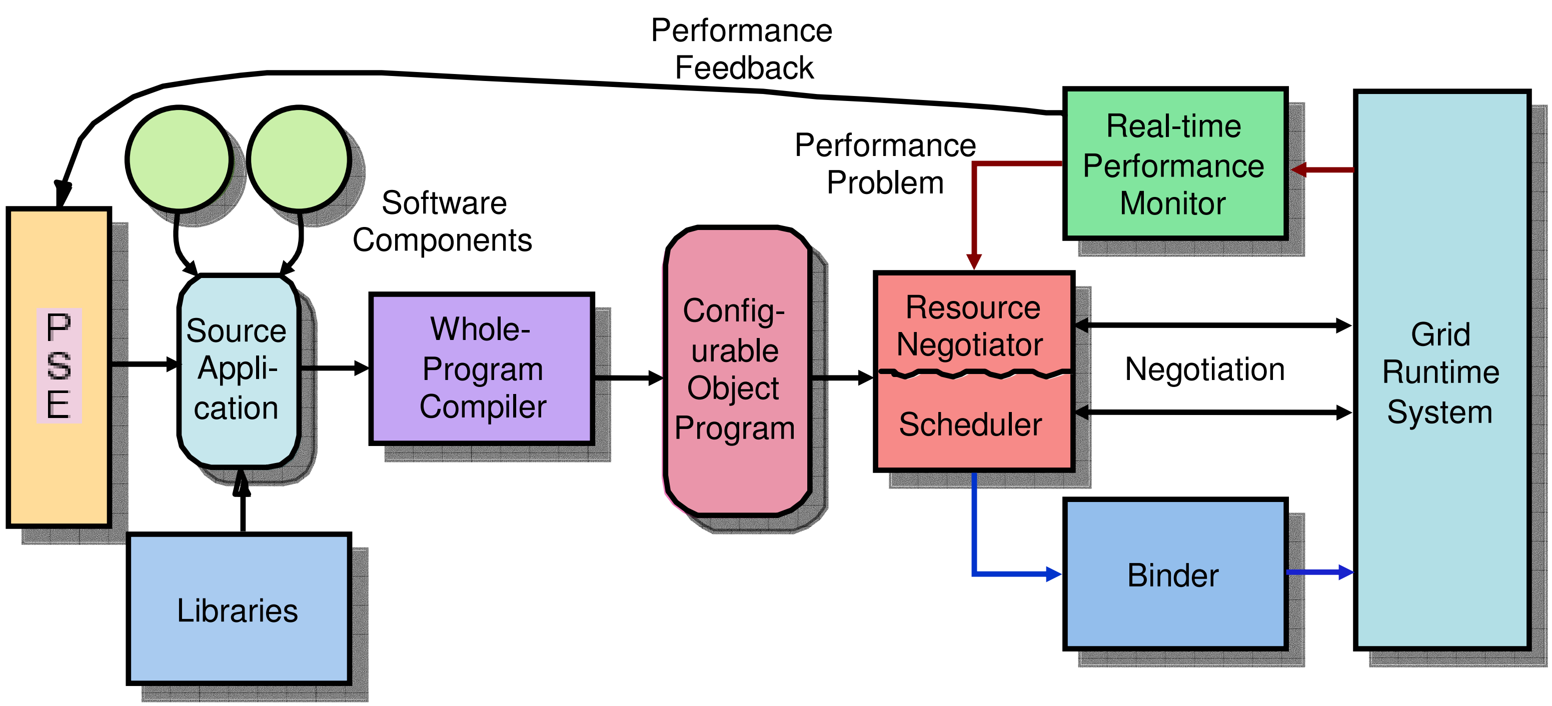
Lost in the search for parallelism

- ACRI
- Alliant
- American Supercomputer
- Ametek
- Applied Dynamics
- Astronautics
- BBN
- CDC
- Cogent
- Convex > HP
- Cray Computer
- Cray Research > SGI > Cray
- Culler-Harris
- Culler Scientific
- Cydrome
- Dana/Ardent/Stellar/Stardent
- Denelcor
- Encore
- Elexsi
- ETA Systems
- Evans and Sutherland Computer
- Exa
- Flexible
- Floating Point Systems
- Galaxy YH-1

- Goodyear Aerospace MPP
- Gould NPL
- Guiltech
- Intel Scientific Computers
- International Parallel Machines
- Kendall Square Research
- Key Computer Laboratories *searching again*
- MasPar
- Meiko
- Multiflow
- Myrias
- Numerix
- Pixar
- Parsytec
- nCube
- Prisma
- Pyramid
- Ridge
- Saxpy
- Scientific Computer Systems (SCS)
- Soviet Supercomputers
- Supertek
- Supercomputer Systems
- Suprenum
- Tera > Cray Company
- Thinking Machines
- Vitesse Electronics
- Wavetracer

Grids and Teragrids

GrADSoft Architecture



Building on Legacy Software

- Nimrod
  - Support parametric computation without programming
  - High performance distributed computing
    - Clusters (1994 – 1997)
    - The Grid (1997 -) (Added QOS through Computational Economy)
  - Nimrod/O – Optimisation on the Grid
  - Active Sheets – Spreadsheet interface
- GriddLeS
  - General Grid Applications using Legacy Software
  - Whole applications as components
  - Using no new primitives in application

Some science is hitting a wall
FTP and GREP are not adequate (Jim Gray)

You can GREP 1 GB in a minute
You can GREP 1 TB in 2 days
You can GREP 1 PB in 3 years.

1PB ~10,000 >> 1,000 disks

At some point you need
indices **to limit search**
parallel **data search and analysis**

You can FTP 1 MB in
sec.

You can FTP 1 GB / r

… 2 days and 1

3 years and 1

Goal using dbases. Make it easy to
Publish: Record structured data
Find data anywhere in the network
Get the subset you need!
Explore datasets interactively

<u>Database becomes the file system!!!</u>

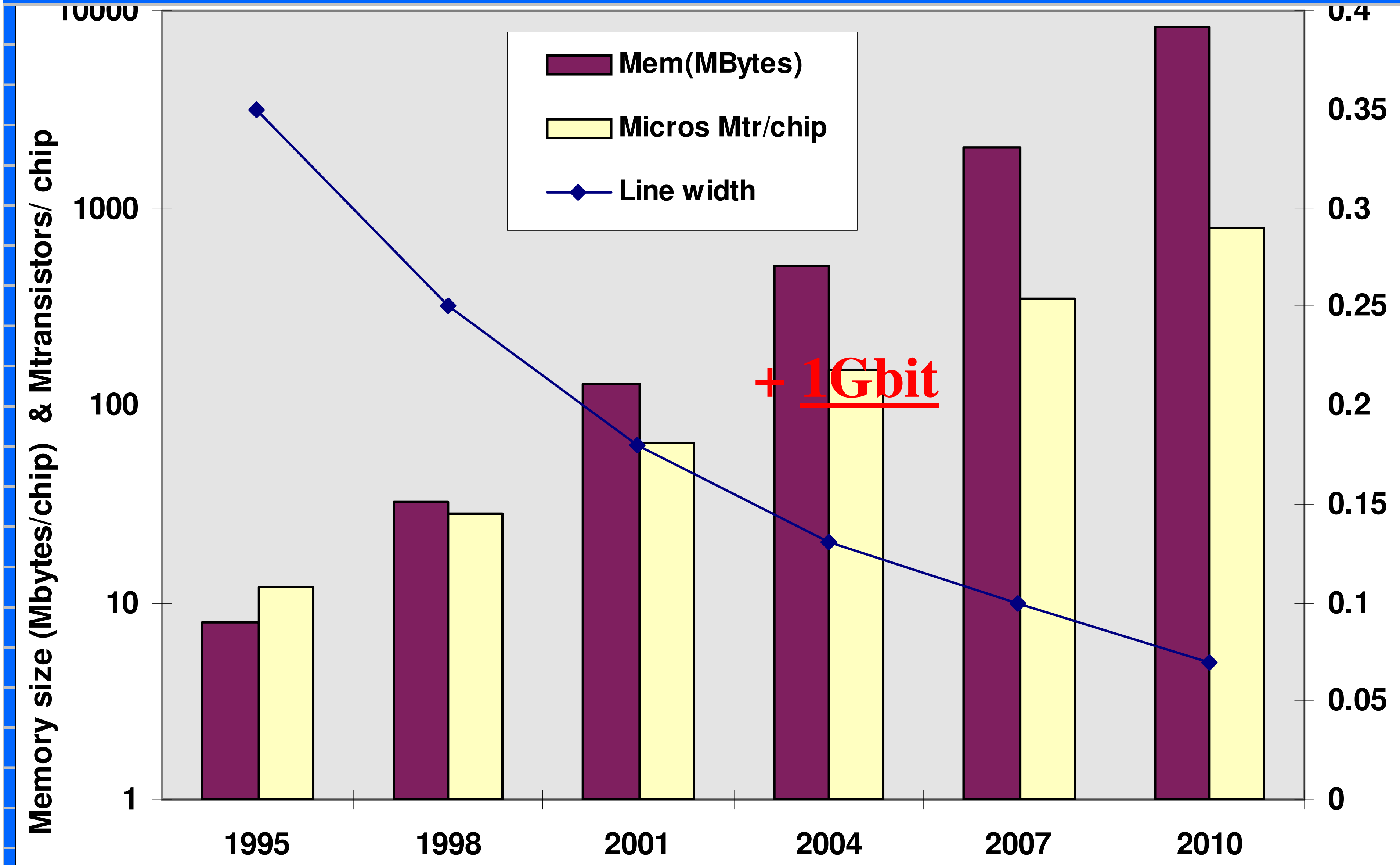
What can be learned from Sky Server?

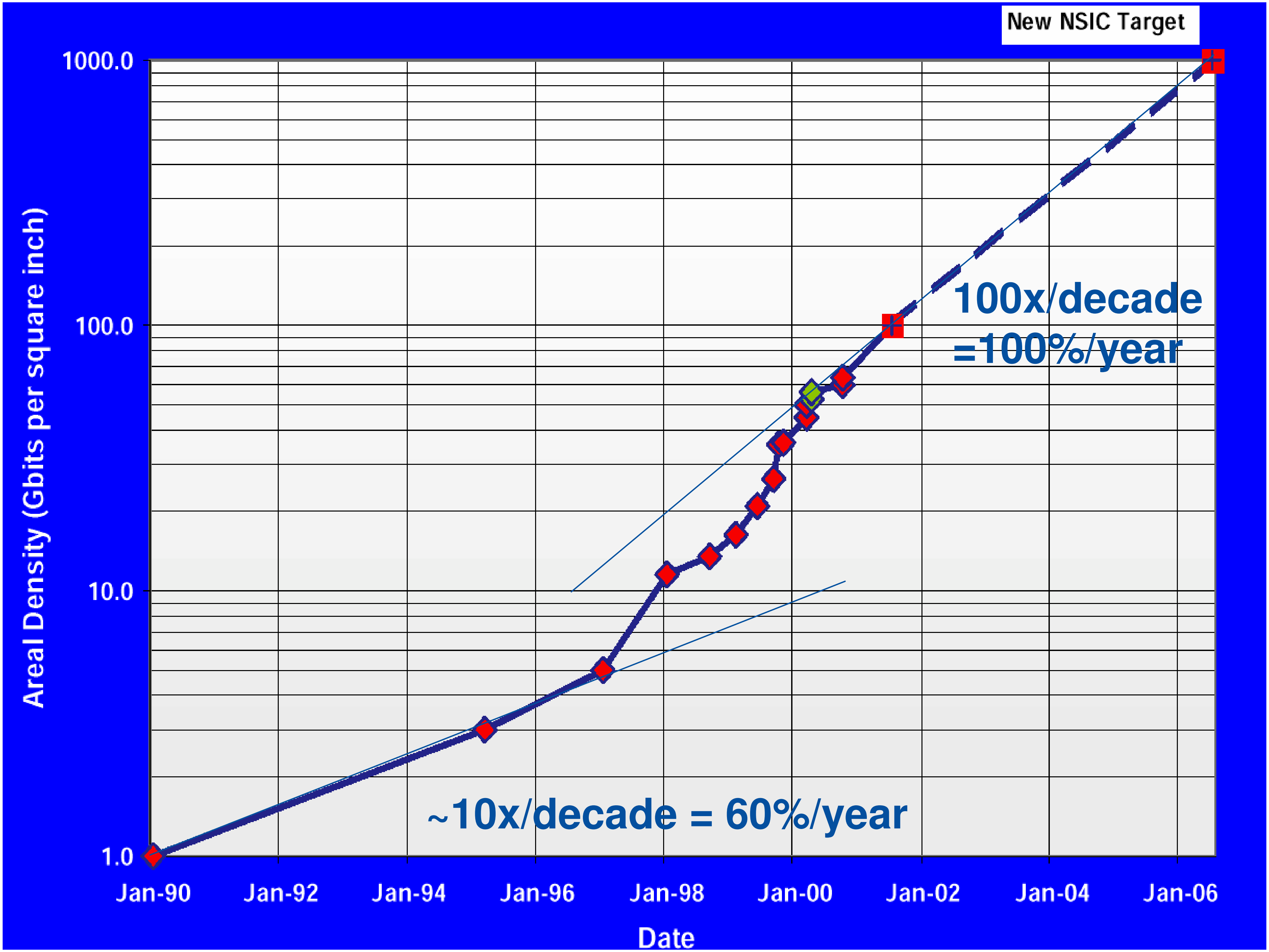
- **It's about data, not about harvesting flops**
- **1-2 hr. query programs versus 1 wk programs based on grep**
- **10 minute runs versus 3 day compute & searches**
- **Database viewpoint. 100x speed-ups**
  - **Avoid costly re-computation and searches**
  - **Use indices and PARALLEL I/O. Read / Write >>1.**
  - **Parallelism is automatic, transparent, and just depends on the number of computers/disks.**
- **Limited experience and talent to use dbases**

Technology: peta-bytes, -flops, -bps
We get no technology before its time

- Moore's Law 2004-2012: 40X
- <u>The big surprise: 64 bit micro with 2-4 processors 8-32 GByte memories</u>
- 2004: O(100) processors = 300 GF PAP, $100K
  - 3 TF/M, not diseconomy of scale for large systems
  - 1 PF => 330M, but 330K processors; other paths
- Storage 1-10 TB disks; 100-1000 disks
- Networking cost is between 0 and unaffordable!
- Cost of disks < cost to transfer its contents!!!
- Internet II killer app – NOT teragrid
  - Access Grid, new methods of communication
  - Response time to provide web services

National Semiconductor Technology Roadmap (size)





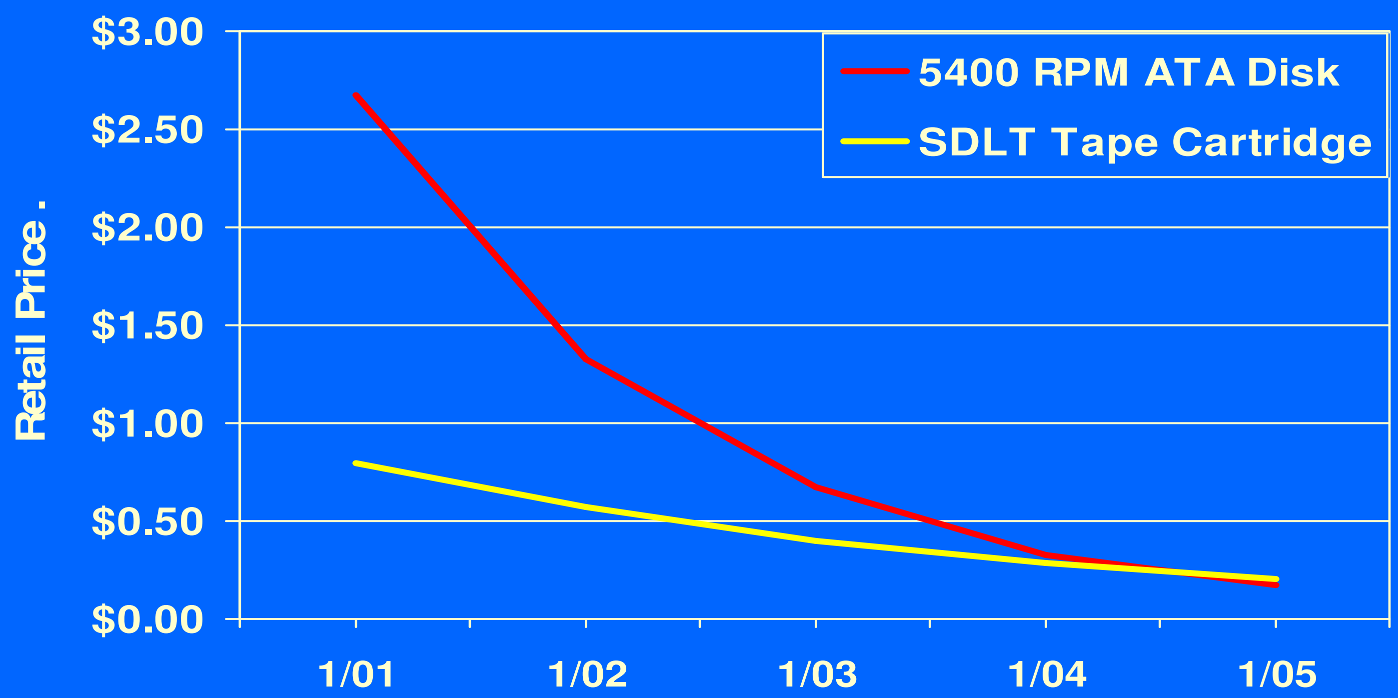
New NSIC Target
1000.0
100.0
10.0
1.0
Areal Density (Gbits per square inch)
100x/decade
=100%/year
~10x/decade = 60%/year
Jan-90
Jan-92
Jan-94
Jan-96
Jan-98
Jan-00
Jan-02
Jan-04
Jan-06
Date

Disk Density Explosion

- Magnetic disk recording density (bits per mm^2) grew at 25% per year from 1975 until 1989.
- Since 1989 it has grown at 60-70% per year
- Since 1998 it has grown at >100% per year
  - This rate will continue into 2003
- Factors causing accelerated growth:
  - Improvements in head and media technology
  - Improvements in signal processing electronics
  - *Lower head flying heights*

Courtesy Richie Lary

Disk / Tape Cost Convergence

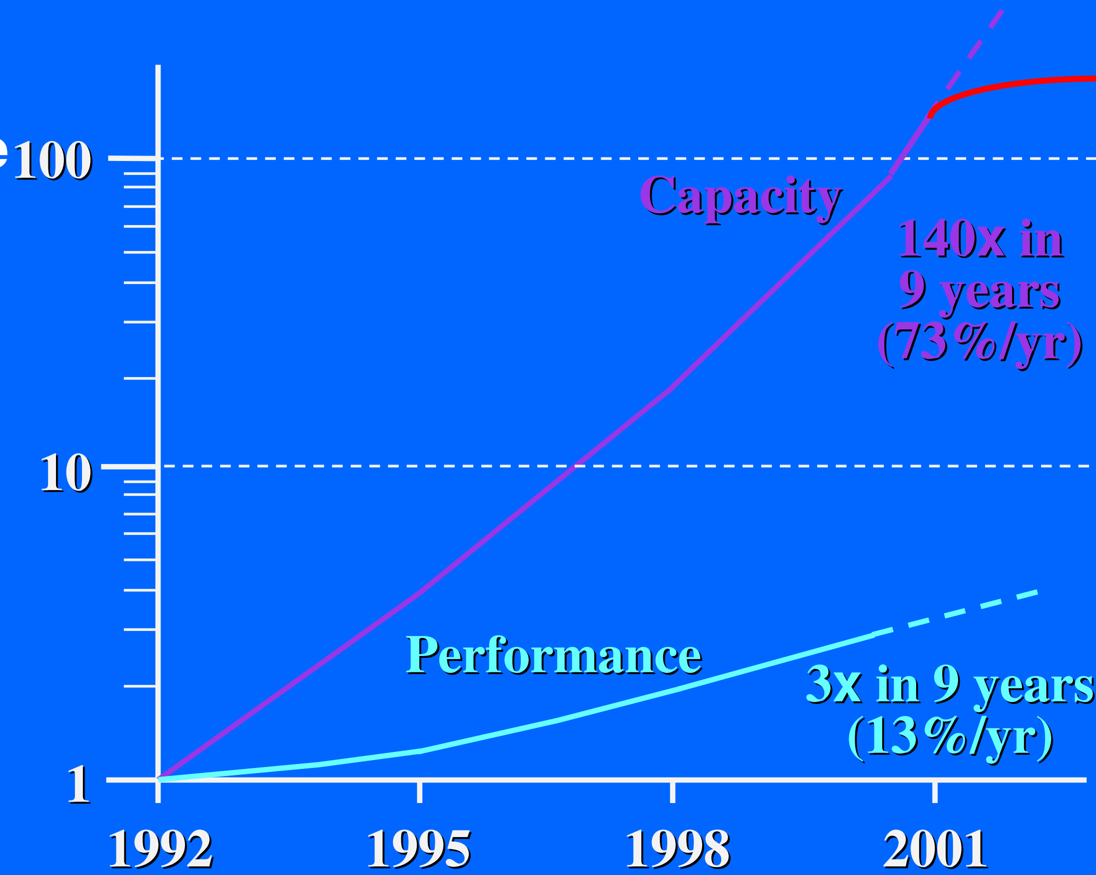


- 3½" ATA disk could cost less than SDLT <u>cartridge</u> in 2004.
  - <u>If</u> disk manufacturers maintain 3½", multi-platter form factor
- Volumetric density of disk will exceed tape in 2001.
- "Big Box of ATA Disks" could be cheaper than a tape library of equivalent size in 2001

Courtesy of Richard Lary

Disk Capacity / Performance Imbalance

- Capacity growth outpacing performance growth
- Difference must be made up by better caching and load balancing
- Actual disk capacity may be capped by market (red line); shift to smaller disks (already happening for high speed disks)



Courtesy of Richard Lary

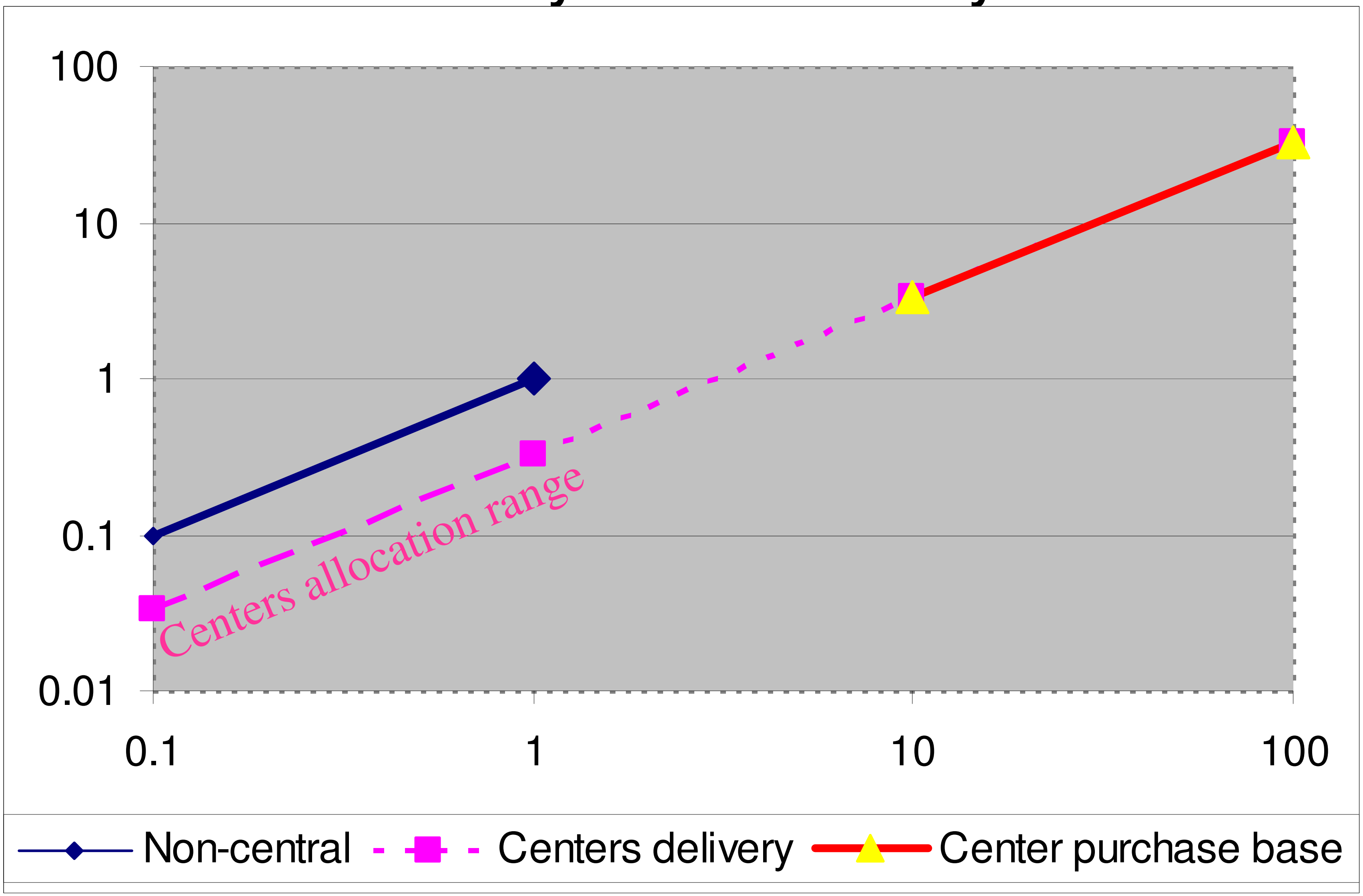
Review the bidding

- 1984: "The Japanese are coming to create the 5th Generation".
  - CMOS and killer Micros. Build // machines.
  - 40+ computers were built & failed based on CMOS and/or micros
  - No attention to software or apps. "State computers" needed.
- 1994: Parallelism and Grand Challenges
  - Converge to Linux Clusters (Constellations >1 Proc.) & MPI
  - No noteworthy middleware software to aid apps or replace Fortran
  - Grand Challenges: the *forgotten* Washington slogan.
- 2004: Teragrid, a massive computer Or just a massive project?
  - Massive review and re-architecture of centers and their function.
  - Science becomes community (app/data/instrument) centric (Calera, CERN, Fermi, NCAR)
- 2004: The Japanese have come.
  GW Bush: "The US will regain supercomputing leadership."
  - Clusters to reach a <$300M Petaflop will evolve by 2010-2014

Centers: The role going forward

- The US builds scalable clusters, NOT supercomputers
  - Scalables are 1 to n *commodity* PCs that anyone can assemble.
  - Unlike the "Crays" all clusters are equal. Use allocated in small clusters.
  - Problem parallelism *sans* ∞// has been elusive (limited to 100-1,000)
  - No advantage of having a computer larger than a //able program
- User computation can be acquired and managed effectively.
  - Computation is divvied up in small clusters e.g. 128-1,000 nodes that individual groups can acquire and manage effectively
- The basic hardware evolves, doesn't especially favor centers
  - 64-bit architecture. 512Mb x 32/dimm = 8GB >>16GB Systems (Centers machine become quickly obsolete, by memory / balance rules.)
  - 3 year timeframe: 1 TB disks at $0.20/TB
  - Last mile communication costs not decreasing to favor centers or grids.

Performance(TF) vs. cost($M) of non-central and centrally distributed systems



Community re-Centric Computing
Time for a major change --from batch to web-service

- Community Centric: “web service”
- Community is responsible
  - Planned & budget as resources
  - Responsible for its infrastructure
  - Apps are from community
  - Computing is integral to work
- In sync with technologies
  - 1-3 Tflops/$M; 1-3 PBytes/$M to buy smallish Tflops & PBytes.
- New scalables are “centers” fast
  - Community can afford
  - Dedicated to a community
  - Program, data & database centric
  - May be aligned with instruments or other community activities
- Output = web service; Can communities become communities *to supply services*?

- Centers Centric: “batch processing”
- Center is responsible
  - Computing is “free” to users
  - Provides a vast service array for all
  - Runs & supports all apps
  - Computing grant disconnected fm work
- Counter to technologies directions
  - More costly. Large centers operate at a dis-economy of scale
- Based on unique, fast computers
  - Center can only afford
  - Divvy cycles among all communities
  - Cycles centric; but politically difficult to maintain highest power vs more centers
  - Data is shipped to centers requiring, expensive, fast networking
- Output = diffuse among gp centers; Can centers support on-demand, real time web services?

Community Centric Computing... Versus Computer Centers

- Goal: Enable technical communities to create and take responsibility for their own computing environments of personal, data, and program collaboration and distribution.
- Design based on technology and cost, e.g. networking, apps programs maintenance, databases, and providing 24x7 web and other services
- Many alternative styles and locations are possible
  - Service from existing centers, including many state centers
  - Software vendors could be encouraged to supply apps web services
  - NCAR style center based on shared data and apps
  - Instrument- and model-based databases. Both central & distributed when multiple viewpoints create the whole.
  - Wholly distributed services supplied by many individual groups

Centers Centric: “batch processing”

- Center is responsible
  – Computing is “free” to users
  – Provides a vast service array for all
  – Runs & supports all apps
  – Computing grant disconnected fm work
- Counter to technologies directions
  – More costly. Large centers operate at a dis-economy of scale
- Based on unique, large expensive computers that
  – Center can only afford
  – Divvied up among all communities
  – Cycles centric; but politically difficult to maintain highest power against pressure on funders for more centers
  – Data is shipped to centers requiring, expensive, fast networking
- Output = diffuse among general purpose centers;
  Can centers support on-demand, real time web services?

Re-Centering to Community Centers

- There is little rational support for general purpose centers
  - Scalability changes the architecture of the entire Cyberinfrastructure
  - No need to have a computer bigger than the largest parallel app.
  - They aren't super.
  - World is substantially data driven, not cycles driven.
  - Demand is de-coupled from supply planning, payment or services
- Scientific / Engineering computing has to be the responsibility of each of its communities
  - Communities form around instruments, programs, databases, etc.
  - Output is web service for the entire community

The End