CS 594 Spring 2003
Lecture 1:
Overview of High-Performance Computing

Jack Dongarra
Computer Science Department
University of Tennessee

Most Important Slide

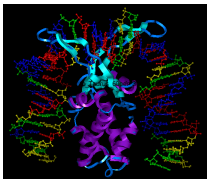
- Netlib - software repository
  - Go to http://www.netlib.org/
- Register for the na-digest
  - Go to http://www.netlib.org/na-net/
  - Register to receive the na-digest
    - » http://www.netlib.org/na-net/join_mail_forw.html

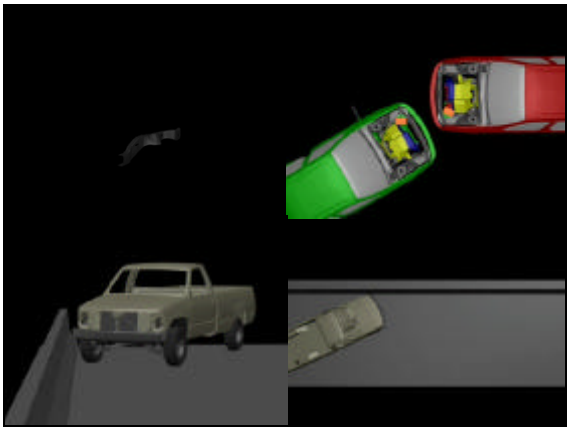
Computational Science

- HPC offers a new way to do science:
  - Experiment - Theory - Computation
- Computation used to approximate physical systems - Advantages include:
  - Playing with simulation parameters to study emergent trends
  - Possible replay of a particular simulation event
  - Study systems where no exact theories exist

Why Turn to Simulation?

- When the problem is too . . .
  - Complex
  - Large / small
  - Expensive
  - Dangerous
- to do any other way.

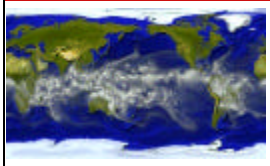




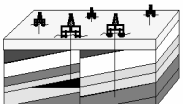
Automotive Industry

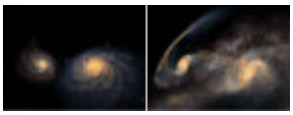
- Huge users of HPC technology;
  - Ford is 25th largest user of HPC in the world
- Main uses of simulation:
  - Aerodynamics (similar to aerospace)
  - Crash simulation
  - Metal sheet formation
  - Noise/vibration optimization
  - Traffic simulation
- Main benefits:
  - Reduced time to market of new cars
  - Increased quality
  - Reduced need to build prototypes
  - more efficient & integrated manufacturing processes

Why Turn to Simulation?



- Climate / Weather Modeling
- Data intensive problems (data-mining, oil reservoir simulation)
- Problems with large length and time scales (cosmology)





Units of High Performance Computing

1 Mflop/s	1 Megaflop/s	10^6 Flop/sec
1 Gflop/s	1 Gigaflop/s	10^9 Flop/sec
1 Tflop/s	1 Teraflop/s	10^{12} Flop/sec
1 Pflop/s	1 Petaflop/s	10^{15} Flop/sec
1 MB	1 Megabyte	10^6 Bytes
1 GB	1 Gigabyte	10^9 Bytes
1 TB	1 Terabyte	10^{12} Bytes
1 PB	1 Petabyte	10^{15} Bytes

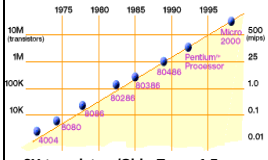
8

High-Performance Computing Today

- In the past decade, the world has experienced one of the most exciting periods in computer development.
- Microprocessors have become smaller, denser, and more powerful.
- The result is that microprocessor-based supercomputing is rapidly becoming the technology of preference in attacking some of the most important problems of science and engineering.

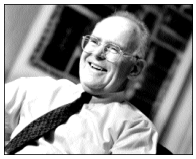
9

Technology Trends: Microprocessor Capacity



2X transistors/Chip Every 1.5 years
Called "Moore's Law"

Microprocessors have become smaller, denser, and more powerful.
Not just processors, bandwidth, storage, etc

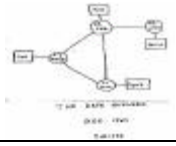


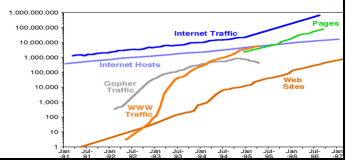
Gordon Moore (co-founder of Intel) predicted in 1965 that the transistor density of semiconductor chips would double roughly every 18 months.

10

Internet – 4th Revolution in Telecommunications

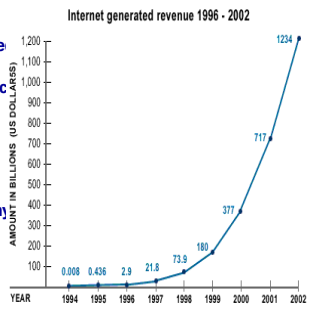
- Telephone, Radio, Television
- Growth in Internet outstrips the others
- Exponential growth since 1985
- Traffic doubles every 100 days





The Web Phenomenon

- 90 - 93 Web invented
- U of Illinois Mosaic release March 94, ~ 0.1% traffic
- September 93 ~ 1% traffic w/200 sites
- June 94 ~ 10% of traffic w/2,000 sites
- Today 60% of traffic w/2,000,000 sites
- Every organization, company, school



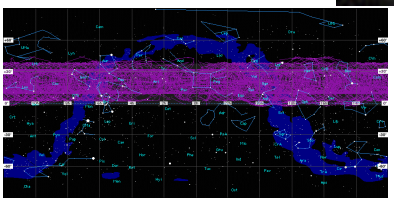
Peer to Peer Computing

- Peer-to-peer is a style of networking in which a group of computers communicate directly with each other.
- Wireless communication
- Home computer in the utility room, next to the water heater and furnace.
- Web tablets
- Imbedded computers in things all tied together.
  - Books, furniture, milk cartons, etc
- Smart Appliances
  - Refrigerator, scale, etc

Internet On Everything

SETI@home: Global Distributed Computing

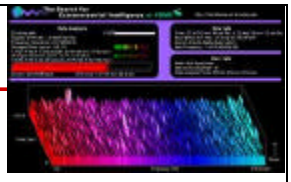
- Running on 500,000 PCs, ~1000 CPU Years per Day
  - 485,821 CPU Years so far
- Sophisticated Data & Signal Processing Analysis
- Distributes Datasets from Arecibo Radio Telescope



15

SETI@home

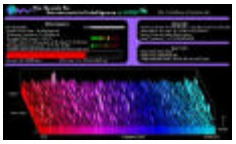
- Use thousands of Internet-connected PCs to help in the search for extraterrestrial intelligence.
- When their computer is idle or being wasted this software will download a 300 kilobyte chunk of data for analysis. Performs about 3 Tflops for each client in 15 hours.
- The results of this analysis are sent back to the SETI team, combined with thousands of other participants.



- Largest distributed computation project in existence
  - Averaging 40 Tflop/s
- Today a number of companies trying this for profit.

16

Grid Computing - from ET to Anthrax



17

Google™

- Google query attributes
  - 150M queries/day (2000/second)
  - 3B documents in the index
- Data centers
  - 15,000 Linux systems in 6 data centers
    - 15 TFlop/s and 1000 TB total capability
    - 40-80 1U/2U servers/cabinet
    - 100 MB Ethernet switches/cabinet with gigabit Ethernet uplink
  - growth from 4,000 systems (June 2000)
    - 18M queries then
- Performance and operation
  - simple reissue of failed commands to new servers
  - no performance debugging
    - problems are not reproducible

Source: Monika Henzinger, Google

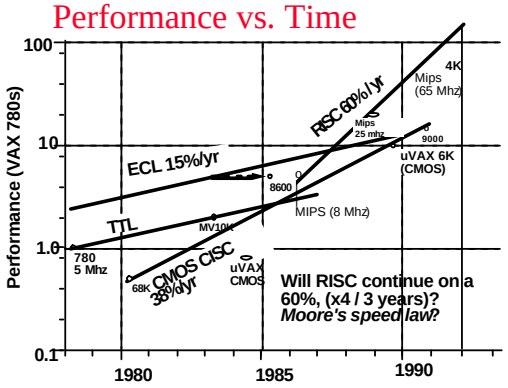
18

Next Generation Web

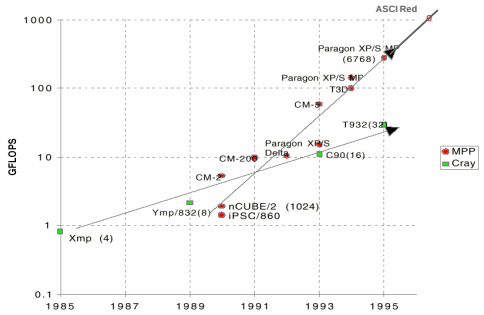
- To treat CPU cycles and software like commodities.
- Enable the coordinated use of geographically distributed resources – in the absence of central control and existing trust relationships.
- Computing power is produced much like utilities such as power and water are produced for consumers.
- Users will have access to "power" on demand
- This is one of our efforts at UT.

19

Performance vs. Time

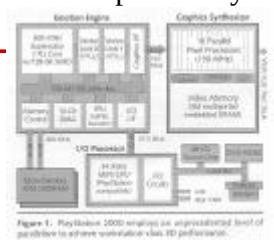


Trends in Computer Performance



21

Other Examples: Sony PlayStation2



- Emotion Engine: 6.2 Gflop/s, 75 million polygons per second (Microprocessor Report, 13:5)
  - Superscalar MIPS core + vector coprocessor + graphics/DRAM
  - Claim: "*Toy Story*" realism brought to games
  - About $250

22

Sony PlayStation2 Export Limits?

svtech/news/breaking/merc/docs/060772.htm

Japan limits Playstation2 export, fears military use

TOKYO, April 16 (Reuters) - Japan has slapped export controls on Sony Corp's new, hugely popular Playstation2 video game because the machine is so sophisticated it could be used for military purposes, media said on Sunday.

The hit home game machine, which includes a digital video disc (DVD) player and will eventually offer Internet access, is Sony's most profitable product. The company said it had shipped 1.4 million in the month after the game's March 4 launch.

The console and its eight-megabyte memory card have been designated as ``general-purpose products related to conventional weapons'' because they contain components that could be used for military devices such as missile guidance systems, Kyodo news agency

23

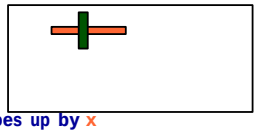
Where Has This Performance Improvement Come From?

- Technology?
- Organization?
- Instruction Set Architecture?
- Software?
- Some combination of all of the above?

24

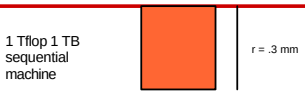
1st Principles

- What happens when the feature size shrinks by a factor of x ?



- Clock rate goes up by x
  - actually less than x, because of power consumption
- Transistors per unit area goes up by x^2
- Die size also tends to increase
  - typically another factor of ~x
- Raw computing power of the chip goes up by ~ x^4 !
  - of which x^3 is devoted either to parallelism or locality

How fast can a serial computer be?



- Consider the 1 Tflop sequential machine
  - data must travel some distance, r, to get from memory to CPU
  - to get 1 data element per cycle, this means 10^{12} times per second at the speed of light, $c = 3x10^8$ m/s
  - so $r < c/10^{12} = .3$ mm
- Now put 1 TB of storage in a .3 mm^2 area
  - each word occupies about 3 $Angstroms^2$, the size of a small atom

26

Processor-Memory Problem

- Processors issue instructions roughly every nanosecond.
- DRAM can be accessed roughly every 100 nanoseconds (!).
- DRAM cannot keep processors busy! And the gap is *growing*:
  - processors getting faster by 60% per year
  - DRAM getting faster by 7% per year (SDRAM and EDO RAM might help, but not enough)

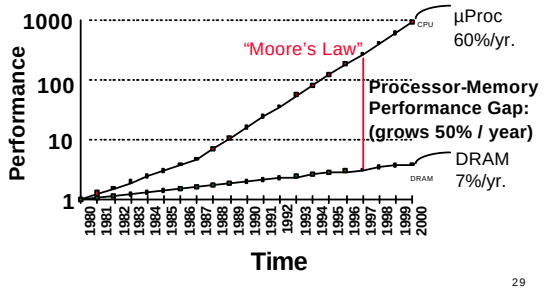
27

RAM

- SDRAM incorporates new features that allow it to keep pace with bus speeds as high as 100 MHz. It does this primarily by allowing two sets of memory addresses to be opened simultaneously.
  - Data can then be retrieved alternately from each set, eliminating the delays that normally occur when one bank of addresses must be shut down and another prepared for reading during each request.
- EDO (extended data output) RAM is a type of random access memory (RAM) chip that improves the time to read from memory on faster microprocessors such as the Intel Pentium.
  - This form of dynamic RAM speeds access to memory locations by working on a simple assumption: the next time memory is accessed, it will be at a contiguous address in a contiguous chunk of hardware. This assumption speeds up memory access times by up to 10 percent over standard DRAM.

28

Processor-DRAM Gap (latency)



29

•Why Parallel Computing

- Desire to solve bigger, more realistic applications problems.
- Fundamental limits are being approached.
- More cost effective solution

30

Principles of Parallel Computing

- Parallelism and Amdahl's Law
- Granularity
- Locality
- Load balance
- Coordination and synchronization
- Performance modeling

➡ All of these things makes parallel programming even harder than sequential programming.

31

"Automatic" Parallelism in Modern Machines

- Bit level parallelism
  - within floating point operations, etc.
- Instruction level parallelism (ILP)
  - multiple instructions execute per clock cycle
- Memory system parallelism
  - overlap of memory operations with computation
- OS parallelism
  - multiple jobs run in parallel on commodity SMPs

Limits to all of these -- for very high performance, need user to identify, schedule and coordinate parallel tasks

32

Finding Enough Parallelism

- Suppose only part of an application seems parallel
- Amdahl's law
  - let f_s be the fraction of work done sequentially, $(1-f_s)$ is fraction parallelizable
  - N = number of processors
- Even if the parallel part speeds up perfectly may be limited by the sequential part

33

Amdahl's Law

Amdahl's Law places a strict limit on the speedup that can be realized by using multiple processors. Two equivalent expressions for Amdahl's Law are given below:

$t_N = (f_p/N + f_s)t_1$ Effect of multiple processors on run time

$S = 1/(f_s + f_p/N)$ Effect of multiple processors on speedup

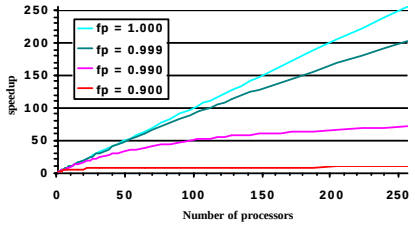
Where:

f_s = serial fraction of code
f_p = parallel fraction of code = $1 - f_s$
N = number of processors

34

Illustration of Amdahl's Law

It takes only a small fraction of serial content in a code to degrade the parallel performance. It is essential to determine the scaling behavior of your code before doing production runs using large numbers of processors



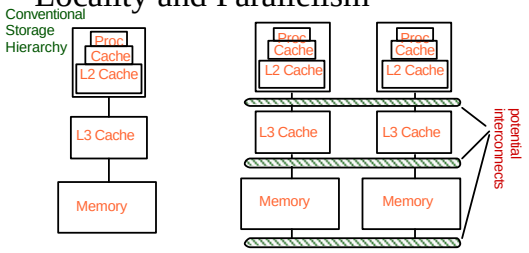
35

Overhead of Parallelism

- Given enough parallel work, this is the biggest barrier to getting desired speedup
- Parallelism overheads include:
  - cost of starting a thread or process
  - cost of communicating shared data
  - cost of synchronizing
  - extra (redundant) computation
- Each of these can be in the range of milliseconds (=millions of flops) on some systems
- Tradeoff: Algorithm needs sufficiently large units of work to run fast in parallel (I.e. large granularity), but not so large that there is not enough parallel work

36

Locality and Parallelism



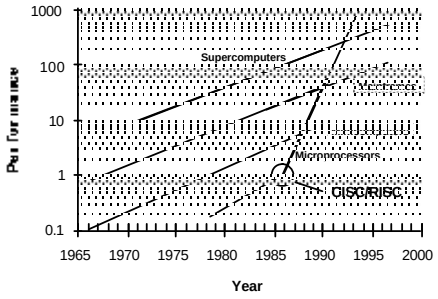
- Large memories are slow, fast memories are small
- Storage hierarchies are large and fast on average
- Parallel processors, collectively, have large, fast $
  - the slow accesses to "remote" data we call "communication"
- Algorithm should do most work on local data

Load Imbalance

- **Load imbalance is the time that some processors in the system are idle due to**
  - **insufficient parallelism (during that phase)**
  - **unequal size tasks**
- **Examples of the latter**
  - **adapting to "interesting parts of a domain"**
  - **tree-structured computations**
  - **fundamentally unstructured problems**
- **Algorithm needs to balance load**

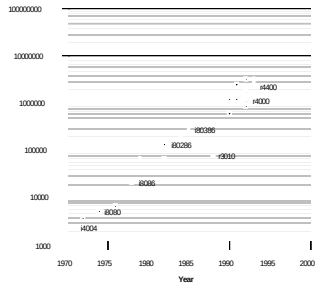
38

Performance Trends Revisited (Architectural Innovation)



39

Performance Trends Revisited (Microprocessor Organization)

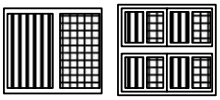


- Bit Level Parallelism
- Pipelining
- Caches
- Instruction Level Parallelism
- Out-of-order Xeq
- Speculation
- . . .

40

What is Ahead?

- **Greater instruction level parallelism?**
- **Bigger caches?**
- **Multiple processors per chip?**
- **Complete systems on a chip? (Portable Systems)**



- **High performance LAN, Interface, and Interconnect**

41

Directions

- **Move toward shared memory**
  - **SMPs and Distributed Shared Memory**
  - **Shared address space w/deep memory hierarchy**
- **Clustering of shared memory machines for scalability**
- **Efficiency of message passing and data parallel programming**
  - **Helped by standards efforts such as MPI and HPF**

42

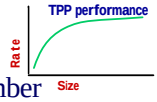
High Performance Computers

- ~ 20 years ago
  - $1x10^6$ Floating Point Ops/sec (Mflop/s)
    - » Scalar based
- ~ 10 years ago
  - $1x10^9$ Floating Point Ops/sec (Gflop/s)
    - » Vector & Shared memory computing, bandwidth aware
    - » Block partitioned, latency tolerant
- ~ Today
  - $1x10^{12}$ Floating Point Ops/sec (Tflop/s)
    - » Highly parallel, distributed processing, message passing, network based
    - » data decomposition, communication/computation
- ~ 10 years away
  - $1x10^{15}$ Floating Point Ops/sec (Pflop/s)
    - » Many more levels MH, combination/grids&HPC
    - » More adaptive, LT and bandwidth aware, fault tolerant, extended precision, attention to SMP nodes

Top 500 Computers

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

$Ax=b$, *dense problem*

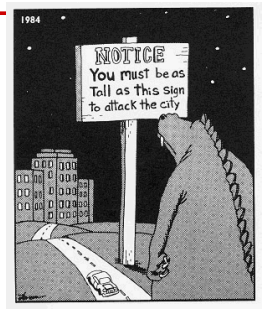


Updated twice a year
SC'xy in the States in November
Meeting in Mannheim, Germany in June

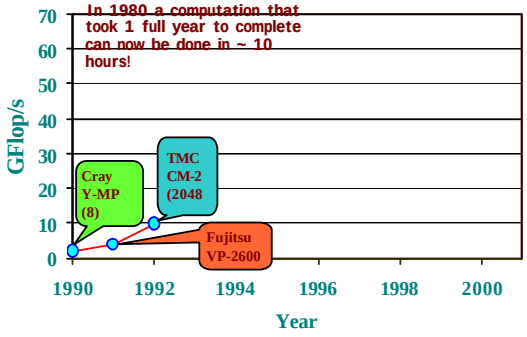
10 Year for Top500 and 25 Year for Linpack Benchmark

Big Means What?

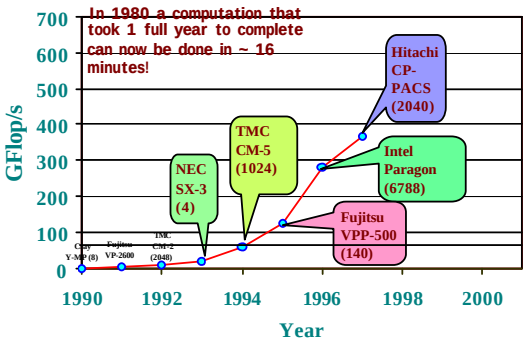
- **Over the last 10 years the range for the Top500 has increased greater than Moore's Law**
- **1993:**
  - #1 = 59.7 GFlop/s
  - #500 = 422 MFlop/s
- **2002:**
  - #1 = 35.8 TFlop/s
  - #500 = 196 GFlop/s



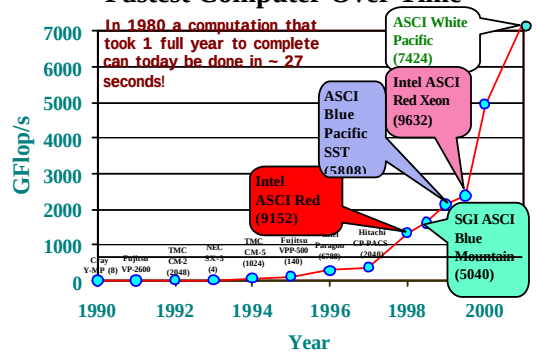
Fastest Computer Over Time

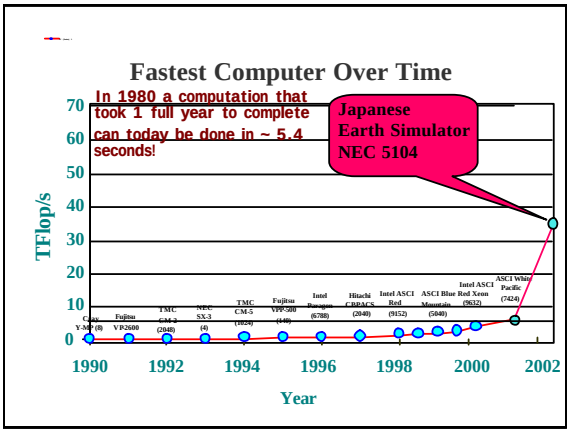


Fastest Computer Over Time



Fastest Computer Over Time





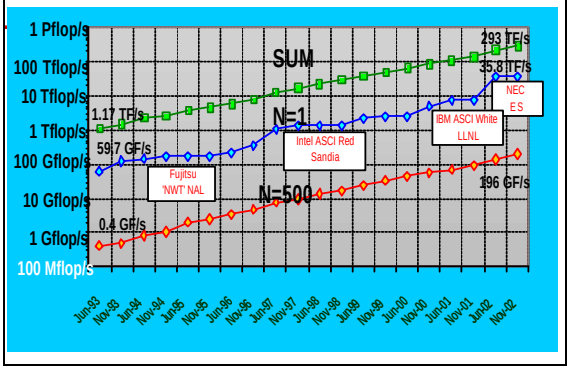
20th List: The TOP10

Rank	Manufacturer	Computer	R_{max} [TF/s]	Installation Site	Country	Year	Area of Installation	# Proc
1	NEC	Earth-Simulator	35.86	Earth Simulator Center	Japan	2002	Research	5120
2	HP	ASCI Q, AlphaServer SC	7.73	Los Alamos National Laboratory	USA	2002	Research	4096
2	HP	ASCI Q, AlphaServer SC	7.73	Los Alamos National Laboratory	USA	2002	Research	4096
4	IBM	ASCI White SP Power3	7.23	Lawrence Livermore National Laboratory	USA	2000	Research	8192
5	Linux NetworX	MCR Cluster	5.69	Lawrence Livermore National Laboratory	USA	2002	Research	8192
6	HP	AlphaServer SC ES45 1 GHz	4.46	Pittsburgh Supercomputing Center	USA	2001	Academic	3016
7	HP	AlphaServer SC ES45 1 GHz	3.98	Commissariat a l'Energie Atomique (CEA)	France	2001	Research	2560
8	HPTi	Xeon Cluster - Myrinet2000	3.34	Forecast Systems Laboratory - NOAA	USA	2002	Research	1536
9	IBM	pSeries 690 Turbo	3.16	HPCx	UK	2002	Academic	1280
10	IBM	pSeries 690 Turbo	3.16	NCAR (National Center for Atmospheric Research)	USA	2002	Research	1216

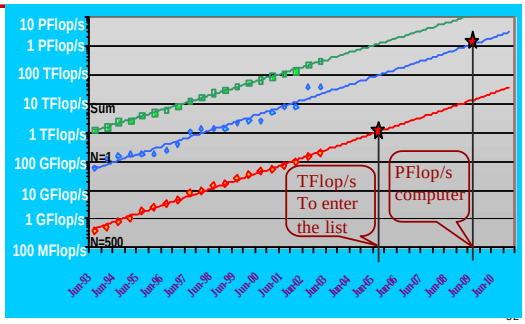
50

182 fell off; 500 was 318 in June

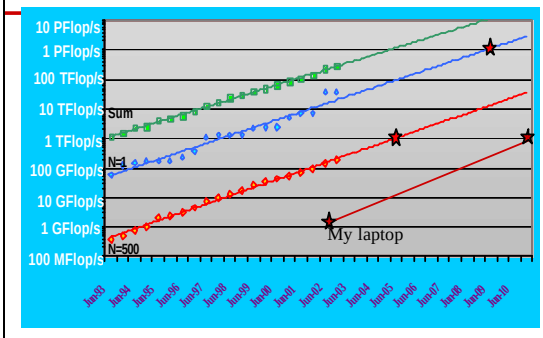
TOP500 - Performance



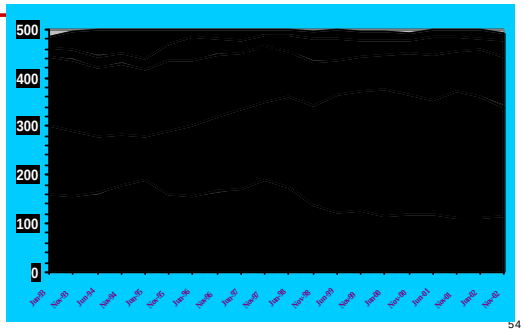
Performance Extrapolation



Performance Extrapolation



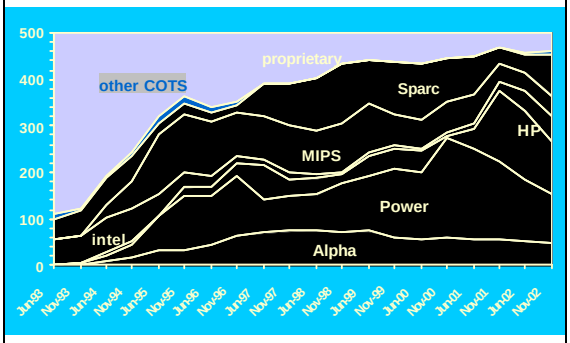
Customer Type



54

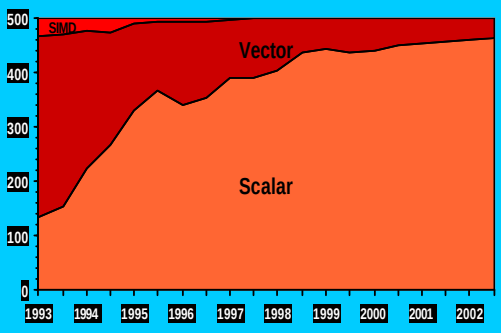
226 Industry, 115 Research, 102 Academic, 32 Classified

Chip Technology

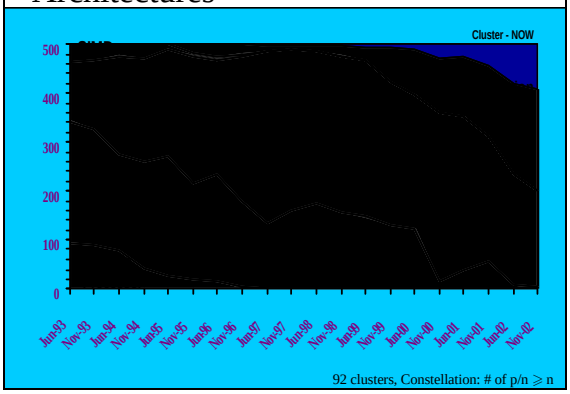


55

Processor Type

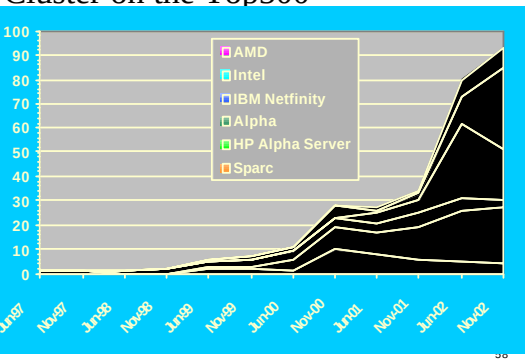


Architectures



92 clusters, Constellation: # of p/n ≥ n

Cluster on the Top500



58

Top500 Conclusions

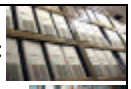
- **Microprocessor based supercomputers have brought a major change in accessibility and affordability.**
- **MPPs continue to account of more than half of all installed high-performance computers worldwide.**

59

Performance Numbers on RISC Processors

Processor	Cycle Time	Linpack n=100	Linpack n=1000	Peak
Intel P4	2540	1190 (23%)	2355 (46%)	5080
Intel/HP Itanium 2	1000	1102 (27%)	3534 (88%)	4000
Compaq Alpha	1000	824 (41%)	1542 (77%)	2000
AMD Athlon	1200	558 (23%)	998 (42%)	2400
HP PA	550	468 (21%)	1583 (71%)	2200
IBM Power 3	375	424 (28%)	1208 (80%)	1500
Intel P3	933	234 (25%)	514 (55%)	933
PowerPC G4	533	231 (22%)	478 (45%)	1066
SUN Ultra 80	450	208 (23%)	607 (67%)	900
SGI Origin 2K	300	173 (29%)	553 (92%)	600
Cray T90	454	705 (39%)	1603 (89%)	1800
Cray C90	238	387 (41%)	902 (95%)	952
Cray Y-MP	166	161 (48%)	324 (97%)	333
Cray X-MP	118	121 (51%)	218 (93%)	235
Cray J-90	100	106 (53%)	190 (95%)	200
Cray 1	80	27 (17%)	110 (69%)	160

High-Performance Computing Directions: Beowulf-class PC Clusters



Definition:

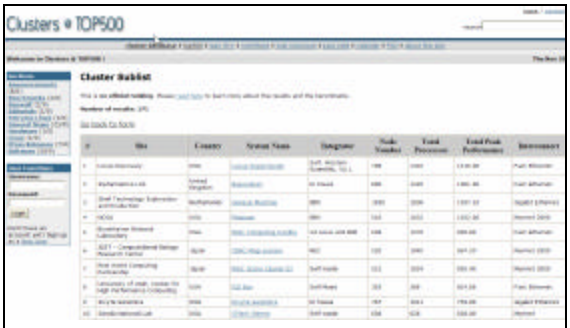
- **COTS PC Nodes**
  - Pentium, Alpha, PowerPC, SMP
- **COTS LAN/SAN Interconnect**
  - Ethernet, Myrinet, Giganet, ATM
- **Open Source Unix**
  - Linux, BSD
- **Message Passing Computing**
  - MPI, PVM
  - HPF

Advantages:

- **Best price-perfo**
- **Low entry-level cost**
- **Just-in-place configuration**
- **Vendor invulnerable**
- **Scalable**
- **Rapid technology tracking**

Enabled by PC hardware, networks and operating system achieving capabilities of scientific workstations at a fraction of the cost and availability of industry standard message passing libraries. However, much more of a contact sport.

61



- **Peak performance**
- **Interconnection**
- **http://clusters.top500.org**
- **Benchmark results to follow in the coming months**

62

Distributed and Parallel Systems

Distributed systems hetero-geneous

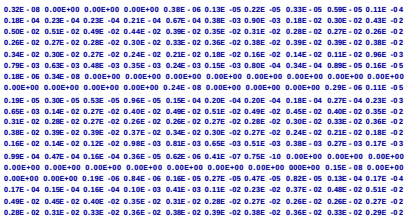
SETI@home, Entropia, GridComputing, Beowulf, Berkley NOW, SNL Cplant, Parallel Dist. mem, ASCI Tflops

Massively parallel systems homo-geneous

- **Gather (unused) resources**
- **Steal cycles**
- **System SW manages resources**
- **System SW adds value**
- **10% - 20% overhead is OK**
- **Resources drive applications**
- **Time to completion is not critical**
- **Time-shared**

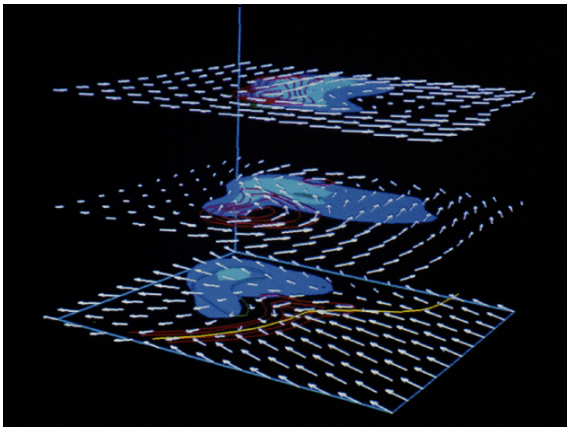
- **Bounded set of resources**
- **Apps grow to consume all cycles**
- **Application manages resources**
- **System SW gets in the way**
- **5% overhead is maximum**
- **Apps drive purchase of equipment**
- **Real-time constraints**
- **Space-shared**

Virtual Environments

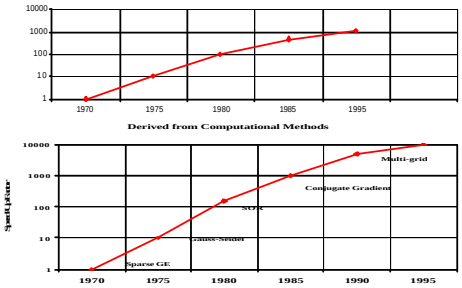


Do they make any sense?

64



Performance Improvements for Scientific Computing Problems



66

Different Architectures

- **Parallel computing: single systems with many processors working on same problem**
- **Distributed computing: many systems loosely coupled by a scheduler to work on related problems**
- **Grid Computing: many systems tightly coupled by software, perhaps geographically distributed, to work together on single problems or on related problems**

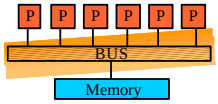
67

Types of Parallel Computers

- **The simplest and most useful way to classify modern parallel computers is by their memory model:**
  - **shared memory**
  - **distributed memory**

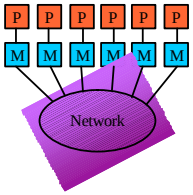
68

Shared vs. Distributed Memory



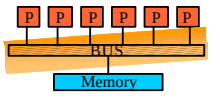
Shared memory - single address space. All processors have access to a pool of shared memory. (Ex: SGI Origin, Sun E10000)

Distributed memory - each processor has it's own local memory. Must do message passing to exchange data between processors. (Ex: CRAY T3E, IBM SP, clusters)



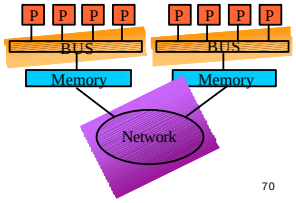
69

Shared Memory: UMA vs. NUMA



Uniform memory access (UMA): Each processor has uniform access to memory. Also known as symmetric multiprocessors (Sun E10000)

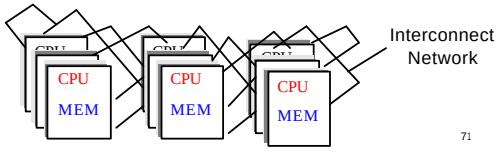
Non-uniform memory access (NUMA): Time for memory access depends on location of data. Local access is faster than non-local access. Easier to scale than SMPs (SGI Origin)



70

Distributed Memory: MPPs vs. Clusters

- **Processors-memory nodes are connected by some type of interconnect network**
  - **Massively Parallel Processor (MPP): tightly integrated, single system image.**
  - **Cluster: individual computers connected by s/w**



71

Processors, Memory, & Networks

- **Both shared and distributed memory systems have:**
  1. **processors: now generally commodity RISC processors**
  2. **memory: now generally commodity DRAM**
  3. **network/interconnect: between the processors and memory (bus, crossbar, fat tree, torus, hypercube, etc.)**
- **We will now begin to describe these pieces in detail, starting with definitions of terms.**

72

Processor-Related Terms

Clock period (cp): the minimum time interval between successive actions in the processor. Fixed, depends on design of processor. Measured in nanoseconds (~1-5 for fastest processors). Inverse of frequency (MHz)

Instruction: an action executed by a processor, such as a mathematical operation or a memory operation.

Register: a small, extremely fast location for storing data or instructions in the processor

73

Processor-Related Terms

Functional Unit: a hardware element that performs an operation on an operand or pair of operations. Common FUs are ADD, MULT, INV, SQRT, etc.

Pipeline : technique enabling multiple instructions to be overlapped in execution

Superscalar: multiple instructions are possible per clock period

Flops: floating point operations per second

74

Processor-Related Terms

Cache: fast memory (SRAM) near the processor. Helps keep instructions and data close to functional units so processor can execute more instructions more rapidly.

TLB: Translation-Lookaside Buffer keeps addresses of pages (block of memory) in main memory that have recently been accessed (a cache for memory addresses)

75

Memory-Related Terms

SRAM: Static Random Access Memory (RAM). Very fast (~10 nanoseconds), made using the same kind of circuitry as the processors, so speed is comparable.

DRAM: Dynamic RAM. Longer access times (~100 nanoseconds), but hold more bits and are much less expensive (10x cheaper).

Memory hierarchy: the hierarchy of memory in a parallel system, from registers to cache to local memory to remote memory. More later

76

Interconnect-Related Terms

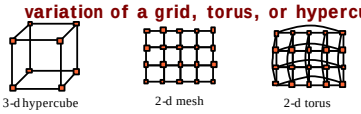
- **Latency**: How long does it take to start sending a "message"? Measured in microseconds.
  - (Also in processors: How long does it take to output results of some operations, such as floating point add, divide etc., which are pipelined?)
- **Bandwidth**: What data rate can be sustained once the message is started? Measured in Mbytes/sec.

77

Interconnect-Related Terms

Topology: the manner in which the nodes are connected.

- Best choice would be a fully connected network (every processor to every other). Unfeasible for cost and scaling reasons.
- Instead, processors are arranged in some variation of a grid, torus, or hypercube.



3-d hypercube 2-d mesh 2-d torus

78

Highly Parallel Supercomputing: Where Are We?

- Performance:
  - Sustained performance has dramatically increased during the last year.
  - On most applications, sustained performance per dollar now exceeds that of conventional supercomputers. But...
  - Conventional systems are still faster on some applications.
- Languages and compilers:
  - Standardized, portable, high-level languages such as HPF, PVM and MPI are available. But ...
  - Initial HPF releases are not very efficient.
  - Message passing programming is tedious and hard to debug.
  - Programming difficulty remains a major obstacle to usage by mainstream scientist.

79

Highly Parallel Supercomputing: Where Are We?

- Operating systems:
  - Robustness and reliability are improving.
  - New system management tools improve system utilization. But...
  - Reliability still not as good as conventional systems.
- I/O subsystems:
  - New RAID disks, HiPPI interfaces, etc. provide substantially improved I/O performance. But...
  - I/O remains a bottleneck on some systems.

80

The Importance of Standards - Software

- Writing programs for MPP is hard ...
- But ... one-off efforts if written in a standard language
- Past lack of parallel programming standards ...
  - ... has restricted uptake of technology (to "enthusiasts")
  - ... reduced portability (over a range of current architectures and between future generations)
- Now standards exist: (PVM, MPI & HPF), which ...
  - ... allows users & manufacturers to protect software investment
  - ... encourage growth of a "third party" parallel software industry & parallel versions of widely used codes

81

The Importance of Standards - Hardware

- Processors
  - commodity RISC processors
- Interconnects
  - high bandwidth, low latency communications protocol
  - no de-facto standard yet (ATM, Fibre Channel, HPPI, FDDI)
- Growing demand for total solution:
  - robust hardware + usable software
- HPC systems containing all the programming tools / environments / languages / libraries / applications packages found on desktops

82

The Future of HPC

- The expense of being different is being replaced by the economics of being the same
- HPC needs to lose its "special purpose" tag
- Still has to bring about the promise of scalable general purpose computing ...
- ... but it is dangerous to ignore this technology
- Final success when MPP technology is embedded in desktop computing
- Yesterday's HPC is today's mainframe is tomorrow's workstation

83

Achieving TeraFlops

- In 1991, 1 Gflop/s
- 1000 fold increase
  - Architecture
    - » exploiting parallelism
  - Processor, communication, memory
    - » Moore's Law
  - Algorithm improvements
    - » block-partitioned algorithms

84

Future: Petaflops (10^{15} fl pt ops/s)

Today $\approx \sqrt{10^{15}}$ flops for our workstations

- A Pflop for 1 second ≈ a typical workstation computing for 1 year.
- From an algorithmic standpoint
  - concurrency
  - data locality
  - latency & sync
  - floating point accuracy
  - dynamic redistribution of workload
  - new language and constructs
  - role of numerical libraries
  - algorithm adaptation to hardware failure

85

A Petaflops Computer System

- 1 Pflop/s sustained computing
- Between 10,000 and 1,000,000 processors
- Between 10 TB and 1PB main memory
- Commensurate I/O bandwidth, mass store, etc.
- If built today, cost $40 B and consume 1 TWatt.
- May be feasible and "affordable" by the year 2010

86