

Supercomputing in Plain English

Overview:

What the Heck is Supercomputing?

CS1313 Spring 2009





What is Supercomputing?

Supercomputing is the biggest, fastest computing right this minute.

Likewise, a supercomputer is one of the biggest, fastest computers right this minute.

So, the definition of supercomputing is constantly changing.

Rule of Thumb: A supercomputer is typically at least 100 times as powerful as a PC.

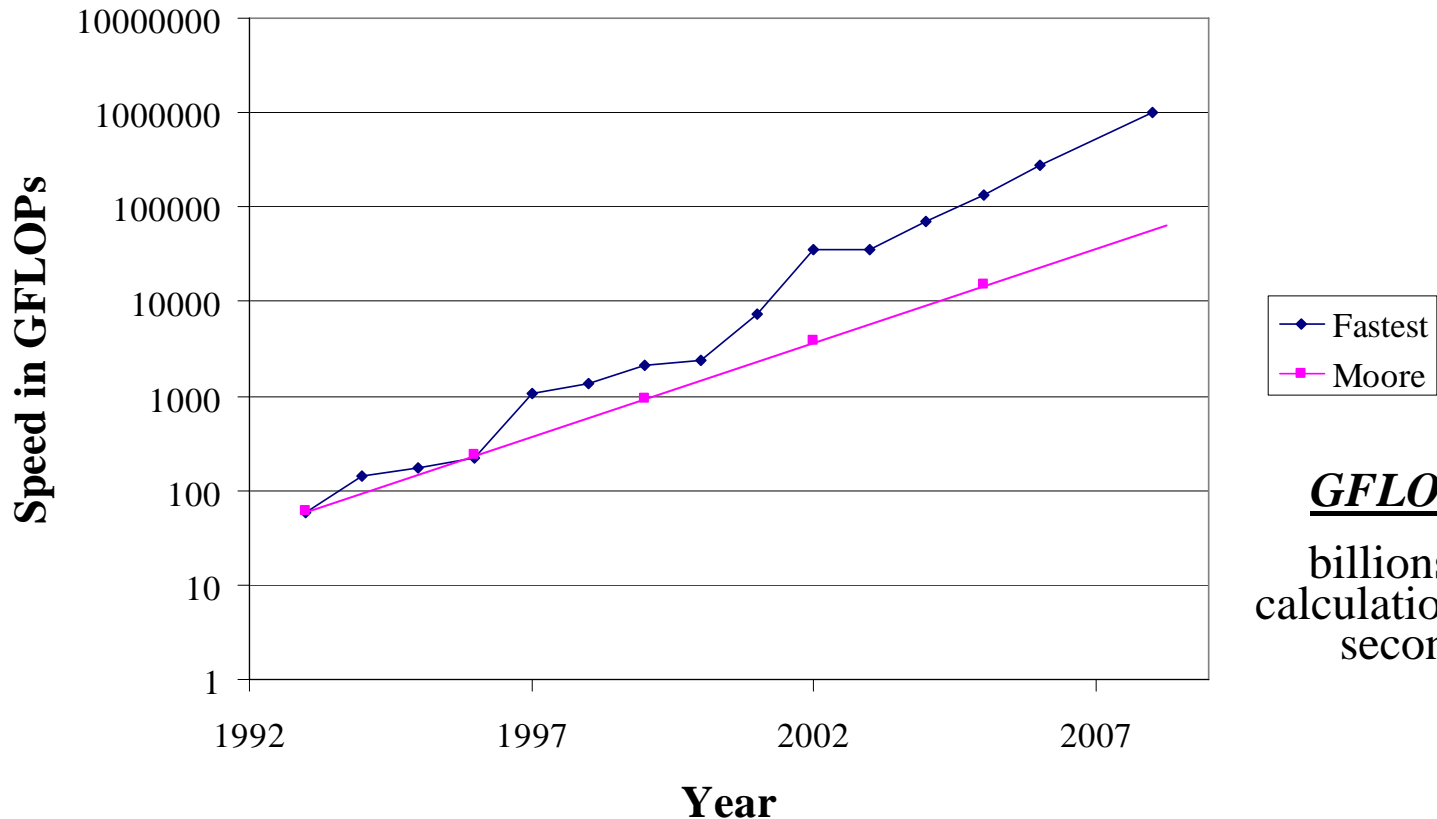
Jargon: Supercomputing is also known as High Performance Computing (HPC) or High End Computing (HEC) or Cyberinfrastructure (CI).





Fastest Supercomputer vs. Moore

Fastest Supercomputer in the World



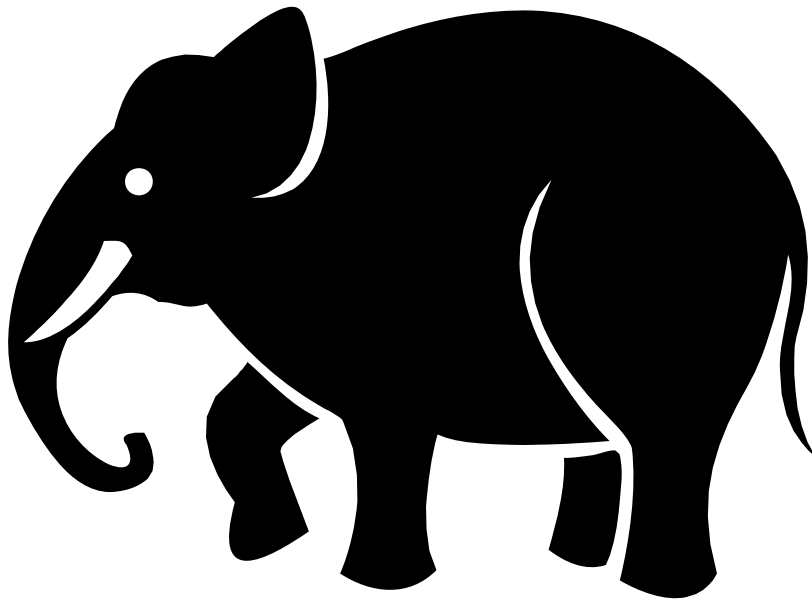
GFLOPs:
billions of
calculations per
second



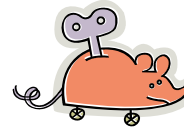
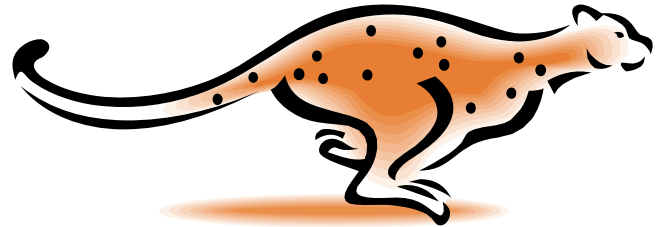


What is Supercomputing About?

Size



Speed



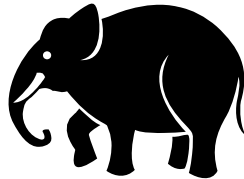
Laptop





What is Supercomputing About?

- **Size:** Many problems that are interesting to scientists and engineers **can't fit on a PC** – usually because they need more than a few GB of RAM, or more than a few 100 GB of disk.



- **Speed:** Many problems that are interesting to scientists and engineers would take a very very long time to run on a PC: months or even years. But a problem that would take **a month on a PC** might take only **a few hours on a supercomputer.**

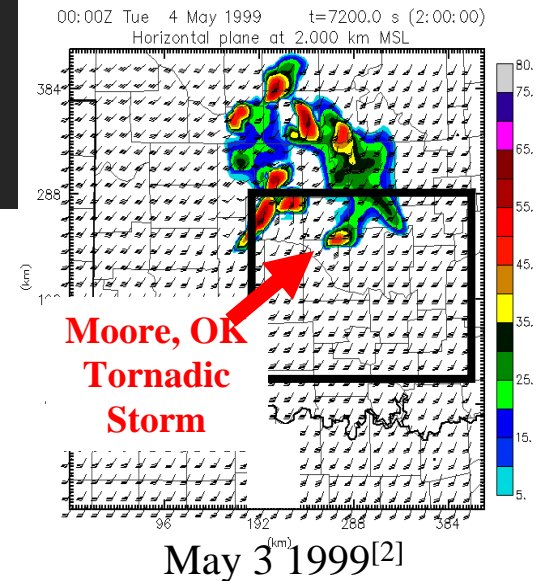
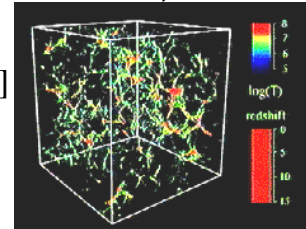




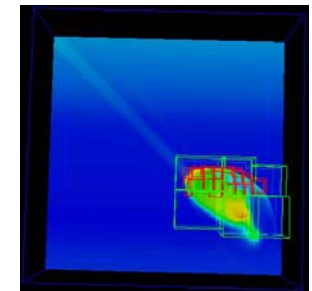
What Is HPC Used For?

- Simulation of physical phenomena, such as
 - Weather forecasting
 - Galaxy formation
 - Oil reservoir management
- Data mining: finding needles of information in a haystack of data, such as
 - Gene sequencing
 - Signal processing
 - Detecting storms that might produce tornados
- Visualization: turning a vast sea of data into pictures that a scientist can understand

[1]



[3]





Supercomputing Issues

- The tyranny of the *storage hierarchy*
- *Parallelism*: doing multiple things at the same time





OSCER





What is OSCER?

- Multidisciplinary center
- Division of OU Information Technology
- Provides:
 - Supercomputing education
 - Supercomputing expertise
 - Supercomputing resources: hardware, storage, software
- For:
 - Undergrad students
 - Grad students
 - Staff
 - Faculty
 - Their collaborators (including off campus)





Who is OSCER? Academic Depts

- Aerospace & Mechanical Engr
- Anthropology
- Biochemistry & Molecular Biology
- Biological Survey
- Botany & Microbiology
- Chemical, Biological & Materials Engr
- Chemistry & Biochemistry
- Civil Engr & Environmental Science
- Computer Science
- Economics
- Electrical & Computer Engr
- Finance
- Health & Sport Sciences
- History of Science
- Industrial Engr
- Geography
- Geology & Geophysics
- Library & Information Studies
- Mathematics
- Meteorology
- Petroleum & Geological Engr
- Physics & Astronomy
- Psychology
- Radiological Sciences
- Surgery
- Zoology

E m E W

More than 150 faculty & staff in 26 depts in Colleges of Arts & Sciences, Atmospheric & Geographic Sciences, Business, Earth & Energy, Engineering, and Medicine – with more to come!





Who is OSCER? Groups

- Advanced Center for Genome Technology
- Center for Analysis & Prediction of Storms
- Center for Aircraft & Systems/Support Infrastructure
- Cooperative Institute for Mesoscale Meteorological Studies
- Center for Engineering Optimization
- Fears Structural Engineering Laboratory
- Human Technology Interaction Center
- Institute of Exploration & Development Geosciences
- Instructional Development Program
- Interaction, Discovery, Exploration, Adaptation Laboratory
- Microarray Core Facility
- OU Information Technology
- OU Office of the VP for Research
- Oklahoma Center for High Energy Physics
- Robotics, Evolution, Adaptation, and Learning Laboratory
- Sasaki Applied Meteorology Research Institute
- Symbiotic Computing Laboratory





Who? External Collaborators

1. California State Polytechnic University Pomona (**masters**)
2. Colorado State University
3. Contra Costa College (CA, **2-year**)
4. Delaware State University (**EPSCoR**, **masters**)
5. Earlham College (IN, **bachelors**)
6. East Central University (OK, **EPSCoR**, **masters**)
7. Emporia State University (KS, **EPSCoR**, **masters**)
8. Great Plains Network
9. Harvard University (MA) E M E W
10. Kansas State University (**EPSCoR**)
11. Langston University (OK, **EPSCoR**, **masters**)
12. Longwood University (VA, **masters**)
13. Marshall University (WV, **EPSCoR**, **masters**)
14. Navajo Technical College (NM, **EPSCoR**, **2-year**)
15. NOAA National Severe Storms Laboratory (**EPSCoR**)
16. NOAA Storm Prediction Center (**EPSCoR**)
17. Oklahoma Baptist University (**EPSCoR**, **bachelors**)
18. Oklahoma City University (**EPSCoR**, **masters**)
19. Oklahoma Climatological Survey (**EPSCoR**)
20. Oklahoma Medical Research Foundation (**EPSCoR**)
21. Oklahoma School of Science & Mathematics (**EPSCoR**, **high school**)
22. Oklahoma State University (**EPSCoR**)
23. Purdue University (IN)
24. Riverside Community College (CA, **2-year**)
25. St. Cloud State University (MN, **masters**)
26. St. Gregory's University (OK, **EPSCoR**, **bachelors**)
27. Southwestern Oklahoma State University (**EPSCoR**, **masters**)
28. Syracuse University (NY)
29. Texas A&M University-Corpus Christi (**masters**)
30. University of Arkansas (**EPSCoR**)
31. University of Arkansas Little Rock (**EPSCoR**)
32. University of Central Oklahoma (**EPSCoR**)
33. University of Illinois at Urbana-Champaign
34. University of Kansas (**EPSCoR**)
35. University of Nebraska-Lincoln (**EPSCoR**)
36. University of North Dakota (**EPSCoR**)
37. University of Northern Iowa (**masters**)





Who Are the Users?

Over 500 users so far, including:

- Roughly equal split between students vs faculty/staff (students are the bulk of the active users);
- many off campus users (roughly 20%);
- ... more being added every month.

Comparison: TeraGrid, consisting of 11 resource provide sites across the US, has ~4500 unique users.





Biggest Consumers

- Center for Analysis & Prediction of Storms: daily real time weather forecasting
- Oklahoma Center for High Energy Physics: simulation and data analysis of banging tiny particles together at unbelievably high speeds





Why OSCER?

- Computational Science & Engineering has become **sophisticated enough** to take its place alongside experimentation and theory.
- **Most students** – and most faculty and staff – **don't learn much CSE**, because it's seen as needing too much computing background, and needs HPC, which is seen as very hard to learn.
- **HPC can be hard to learn**: few materials for novices; most documents written for experts as reference guides.
- **We need a new approach**: HPC and CSE for computing novices – **OSCER's mandate!**





Why Bother Teaching Novices?

- Application scientists & engineers typically know their applications very well, much better than a collaborating computer scientist ever would.
- Commercial software lags far behind the research community.
- Many potential CSE users don't need full time CSE and HPC staff, just some help.
- One HPC expert can help dozens of research groups.
- Today's novices are tomorrow's top researchers, especially because today's top researchers will eventually retire.





What Does OSCER Do? Teaching



Science and engineering faculty from all over America learn supercomputing at OU by playing with a jigsaw puzzle (NCSI @ OU 2004).



Supercomputing in Plain English: Overview
CS1313, Spring 2009



What Does OSCER Do? Rounds



OU undergrads, grad students, staff and faculty learn how to use supercomputing in their specific research.



Supercomputing in Plain English: Overview
CS1313, Spring 2009



OK Supercomputing Symposium 2009



2003 Keynote:
Peter Freeman
NSF
Computer &
Information
Science &
Engineering
Assistant Director



2004 Keynote:
Sangtae Kim
NSF Shared
Cyberinfrastructure
Division Director



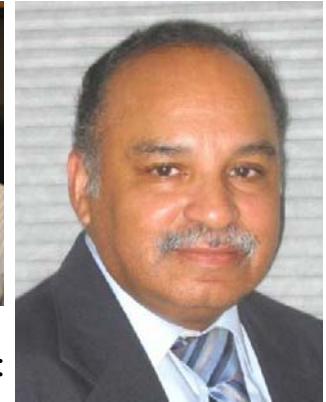
2005 Keynote:
Walt Brooks
NASA Advanced
Supercomputing
Division Director



2006 Keynote:
Dan Atkins
Head of NSF's
Office of
Cyber-
infrastructure



2007 Keynote:
Jay Boisseau
Director
Texas Advanced
Computing Center
U. Texas Austin



2008 Keynote:
José Muñoz
Deputy Office
Director/ Senior
Scientific Advisor
Office of Cyber-
infrastructure
National Science
Foundation



2009 Keynote:
Ed Seidel
Director
NSF Office of
Cyber-
infrastructure

FREE! Wed Oct 7 2009 @ OU

<http://symposium2009.oscer.ou.edu/>

Parallel Programming Workshop

FREE! Tue Oct 6 2009 @ OU

**Sponsored by SC09 Education Program
FREE! Symposium Wed Oct 7 2009 @ OU**

Supercomputing in Plain English: Overview
CS1313, Spring 2009





SC09 Summer Workshops

This coming summer, the SC09 Education Program, part of the SC09 (Supercomputing 2009) conference, is planning to hold two weeklong supercomputing-related workshops in Oklahoma, for **FREE** (except you pay your own transport):

- **At OSU Sun May 17 – the May 23:**
FREE Computational Chemistry for Chemistry Educators (2010 TENTATIVE: Computational Biology)
- **At OU Sun Aug 9 – Sat Aug 15:**
FREE Parallel Programming & Cluster Computing

We'll alert everyone when the details have been ironed out and the registration webpage opens.

Please note that you must **apply** for a seat, and acceptance **CANNOT** be guaranteed.





SC09 Summer Workshops

1. May 17-23: Oklahoma State U: Computational Chemistry
2. May 25-30: Calvin Coll (MI): Intro to Computational Thinking
3. June 7-13: U Cal Merced: Computational Biology
4. June 7-13: Kean U (NJ): Parallel Progrmg & Cluster Comp
5. June 14-20: Widener U (PA): Computational Physics
6. July 5-11: Atlanta U Ctr: Intro to Computational Thinking
7. July 5-11: Louisiana State U: Parallel Progrmg & Cluster Comp
8. July 12-18: U Florida: Computational Thinking Grades 6-12
9. July 12-18: Ohio Supercomp Ctr: Computational Engineering
10. Aug 2- 8: U Arkansas: Intro to Computational Thinking
11. Aug 9-15: U Oklahoma: Parallel Progrmg & Cluster Comp





OSCER Resources



Dell Intel Xeon Linux Cluster

1,072 Intel Xeon CPU chips/4288 cores

- 526 dual socket/quad core Harpertown 2.0 GHz, 16 GB each
- 3 dual socket/quad core Harpertown 2.66 GHz, 16 GB each
- 3 dual socket/quad core Clovertown 2.33 GHz, 16 GB each
- 2 x quad socket/quad core Tigerton, 2.4 GHz, 128 GB each

8,768 GB RAM

~105 TB globally accessible disk

QLogic Infiniband

Force10 Networks Gigabit Ethernet

Red Hat Enterprise Linux 5

Peak speed: 34.45 TFLOPs*

*TFLOPs: trillion calculations per second



sooner.oscer.ou.edu





Dell Intel Xeon Linux Cluster

DEBUTED NOVEMBER 2008 AT:

- #90 worldwide
- #47 in the US
- #14 among US academic
- #10 among US academic excluding TeraGrid
- #2 in the Big 12
- #1 in the Big 12 excluding TeraGrid



`sooner.oscer.ou.edu`





Dell Intel Xeon Linux Cluster

Purchased mid-July 2008

First friendly user Aug 15 2008

Full production Oct 3 2008

Christmas Day 2008: >~75% of nodes and ~66% of cores were in use.



`sooner.oscer.ou.edu`





What is a Cluster?

“... [W]hat a ship is ... It's not just a keel and hull and a deck and sails. That's what a ship needs. But what a ship is ... is freedom.”

– Captain Jack Sparrow

“Pirates of the Caribbean”





What a Cluster is

A cluster **needs** of a collection of small computers, called **nodes**, hooked together by an **interconnection network** (or **interconnect** for short).

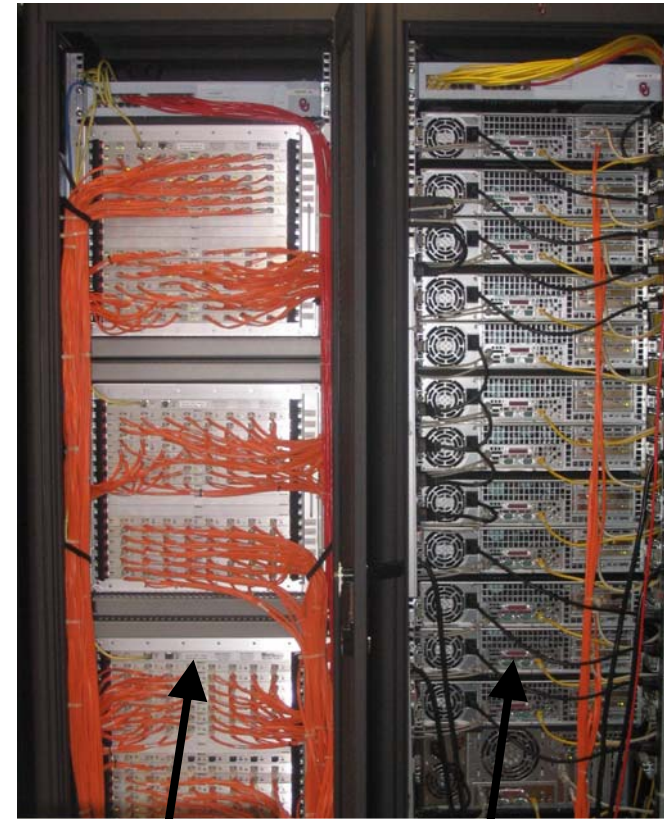
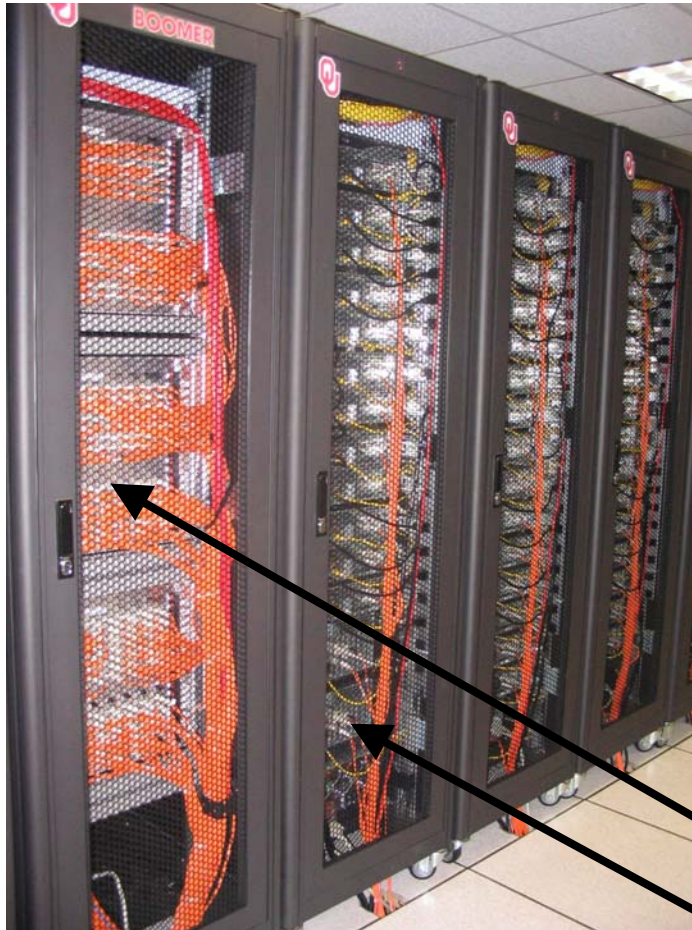
It also **needs** software that allows the nodes to communicate over the interconnect.

But what a cluster **is** ... is all of these components working together as if they're one big computer ... a **super** computer.





An Actual Cluster



Interconnect

Nodes





Condor Pool

Condor is a software technology that allows idle desktop PCs to be used for number crunching.

OU IT has deployed a large Condor pool (773 desktop PCs in IT student labs all over campus).

It provides a huge amount of additional computing power – more than was available in all of OSCER in 2005.

13+ TFLOPs peak compute speed.

And, the cost is very very low – almost literally free.

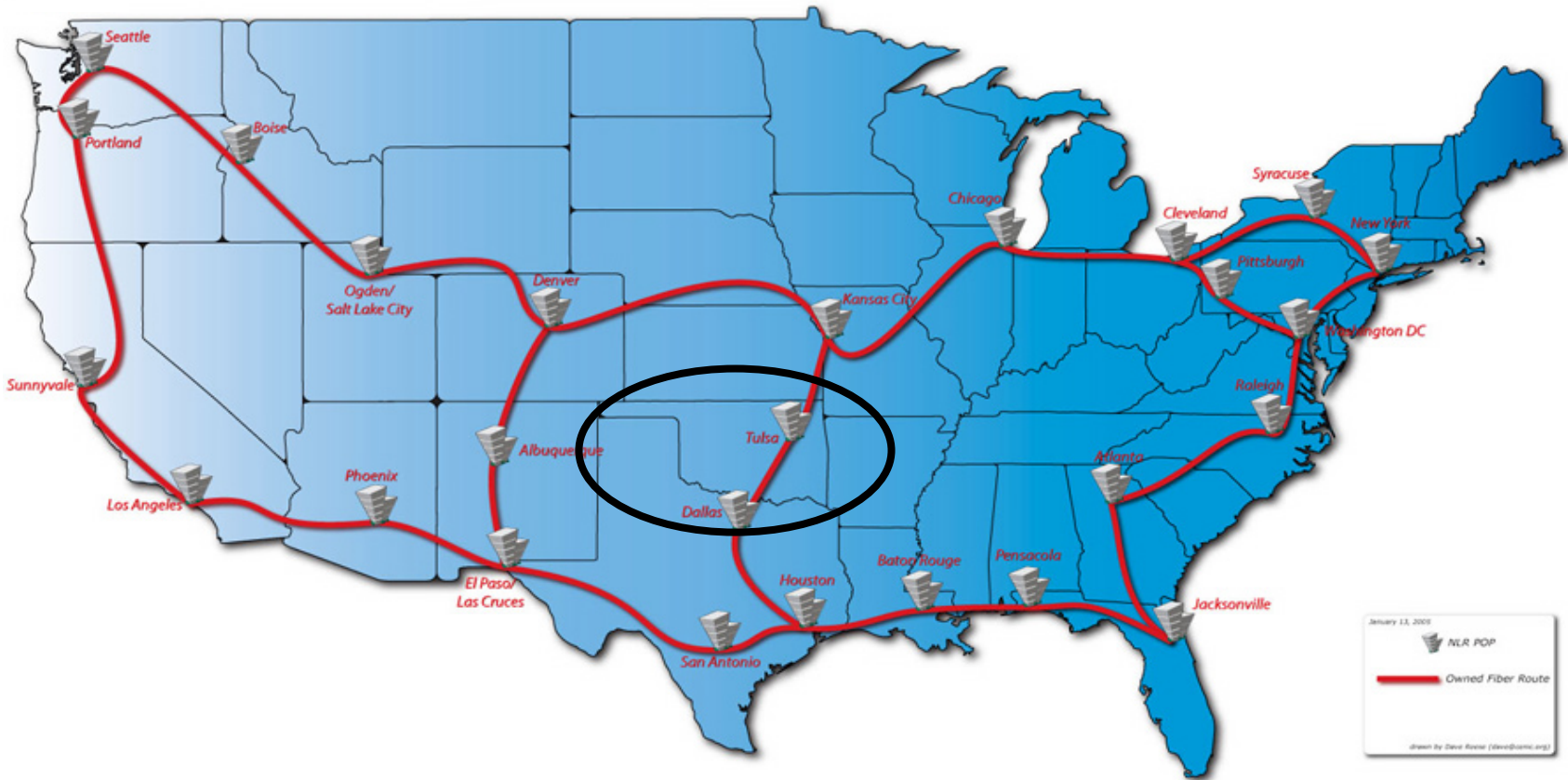
Also, we've been seeing empirically that Condor gets about 80% of each PC's time.

OU and OSU have worked together to get a comparable Condor pool set up at OSU.





National Lambda Rail



January 13, 2005
NLR POP
Owned Fiber Route
Shaped by Dave Rouse (dave@bama.org)



Internet2

Internet2 Network



- CONNECTORS**
- 3ROX
 - CENIC
 - CIC OmniPoP
 - Drexel University
 - GPN
 - Indiana GigaPoP
 - KyRON
 - LEARN
 - LONI
 - MAGPI
 - MAX
 - MCNC
 - Merit Network
 - MREN
 - NOX
 - NYSERNet
 - Oregon Gigapop
 - Pacific Northwest GigaPoP
 - SoX
 - University of Memphis
 - University of New Mexico
 - University of South Florida
 - University of Utah/UEN

www.internet2.edu



Supercomputing in Plain English: Overview
CS1313, Spring 2009

A Quick Primer on Hardware





Henry's Laptop

Dell Latitude D620^[4]



- Pentium 4 Core Duo T2400
1.83 GHz w/2 MB L2 Cache
("Yonah")
- 2 GB (2048 MB)
667 MHz DDR2 SDRAM
- 100 GB 7200 RPM SATA Hard Drive
- DVD_±RW/CD-RW Drive (8x)
- 1 Gbps Ethernet Adapter
- 56 Kbps Phone Modem



Typical Computer Hardware

- Central Processing Unit
- Primary storage
- Secondary storage
- Input devices
- Output devices





Central Processing Unit

Also called *CPU* or *processor*: the “brain”

Components

- *Control Unit*: figures out what to do next – for example, whether to load data from memory, or to add two values together, or to store data into memory, or to decide which of two possible actions to perform (*branching*)
- *Arithmetic/Logic Unit*: performs calculations – for example, adding, multiplying, checking whether two values are equal
- *Registers*: where data reside that are being used right now





Primary Storage

- Main Memory

- Also called RAM (“Random Access Memory”)
- Where data reside when they’re being used by a program that’s currently running

- Cache

- Small area of much faster memory
- Where data reside when they’re about to be used and/or have been used recently

- Primary storage is volatile: values in primary storage disappear when the power is turned off.





Secondary Storage

- Where data and programs reside that are going to be used in the future
- Secondary storage is non-volatile: values don't disappear when power is turned off.
- Examples: hard disk, CD, DVD, Blu-ray, magnetic tape, floppy disk
- Many are portable: can pop out the CD/DVD/tape/floppy and take it with you





Input/Output

- Input devices – for example, keyboard, mouse, touchpad, joystick, scanner
- Output devices – for example, monitor, printer, speakers



The Tyranny of the Storage Hierarchy

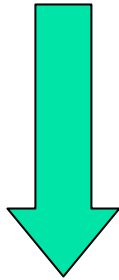




The Storage Hierarchy



Fast, expensive, few



Slow, cheap, a lot



[5]

- Registers
- Cache memory
- Main memory (RAM)
- Hard disk
- Removable media (CD, DVD etc)
- Internet



RAM is Slow

The speed of data transfer between Main Memory and the CPU is much slower than the speed of calculating, so the CPU spends most of its time waiting for data to come in or go out.

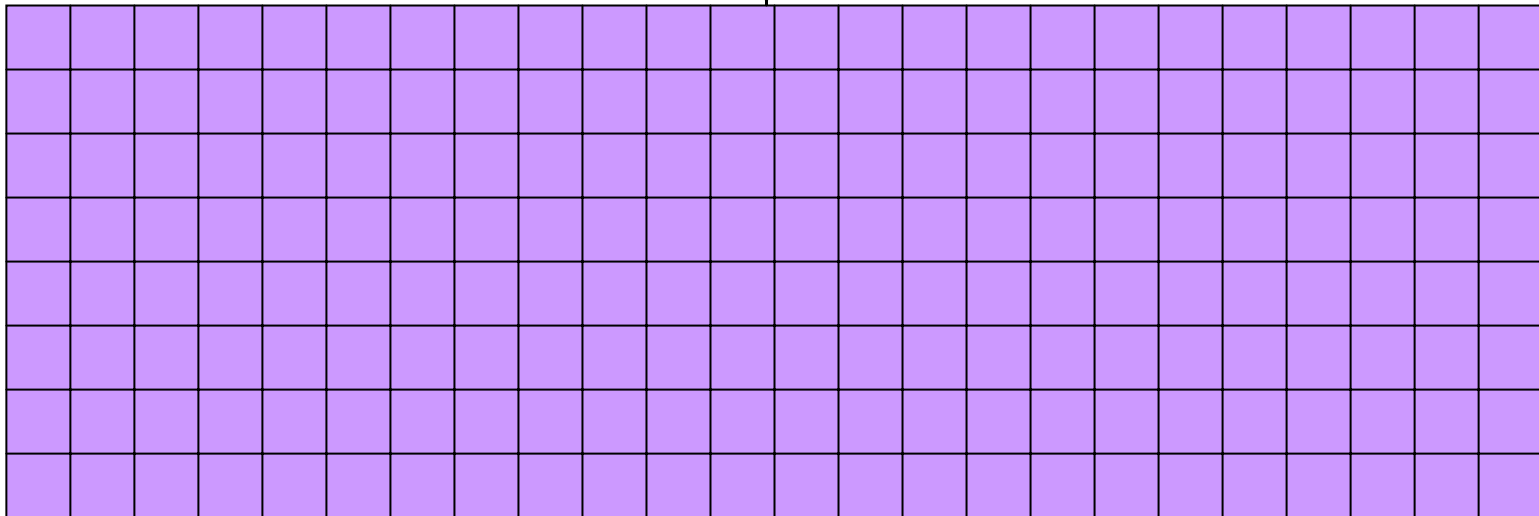
CPU

351 GB/sec^[6]



Bottleneck

3.4 GB/sec^[7] (1%)

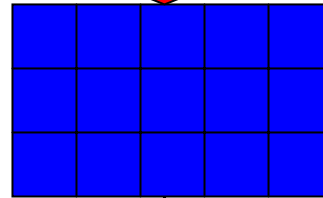




Why Have Cache?

Cache is much closer to the speed of the CPU, so the CPU doesn't have to wait nearly as long for stuff that's already in cache: it can do more operations per second!

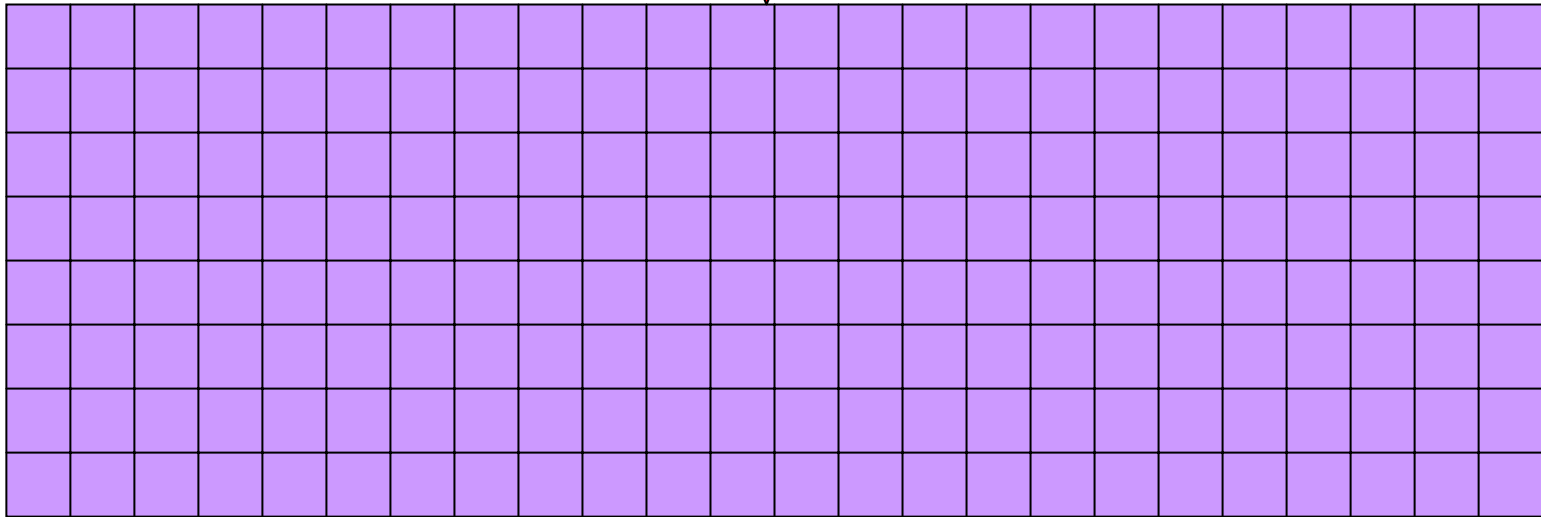
CPU



14.2 GB/sec (4x RAM)^[7]



3.4 GB/sec^[7]





Henry's Laptop

Dell Latitude D620^[4]



- Pentium 4 Core Duo T2400
1.83 GHz w/2 MB L2 Cache
("Yonah")
- 2 GB (2048 MB)
667 MHz DDR2 SDRAM
- 100 GB 7200 RPM SATA Hard Drive
- DVD±RW/CD-RW Drive (8x)
- 1 Gbps Ethernet Adapter
- 56 Kbps Phone Modem



Storage Speed, Size, Cost

Henry's Laptop	Registers (Pentium 4 Core Duo 1.83 GHz)	Cache Memory (L2)	Main Memory (667 MHz DDR2 SDRAM)	Hard Drive (SATA 7200 RPM)	Ethernet (1000 Mbps)	DVD±RW (8x)	Phone Modem (56 Kbps)
Speed (MB/sec) [peak]	359,792 ^[6] (14,640 MFLOP/s*)	14,500 ^[7]	3400 ^[7]	100 ^[9]	125	10.8 ^[10]	0.007
Size (MB)	304 bytes** ^[11]	2	2048	100,000	unlimited	unlimited	unlimited
Cost (\$/MB)	–	\$5 ^[12]	\$0.03 ^[12]	\$0.0001 ^[12]	charged per month (typically)	\$0.00003 ^[12]	charged per month (typically)

* MFLOP/s: millions of floating point operations per second

** 8 32-bit integer registers, 8 80-bit floating point registers, 8 64-bit MMX integer registers, 8 128-bit floating point XMM registers





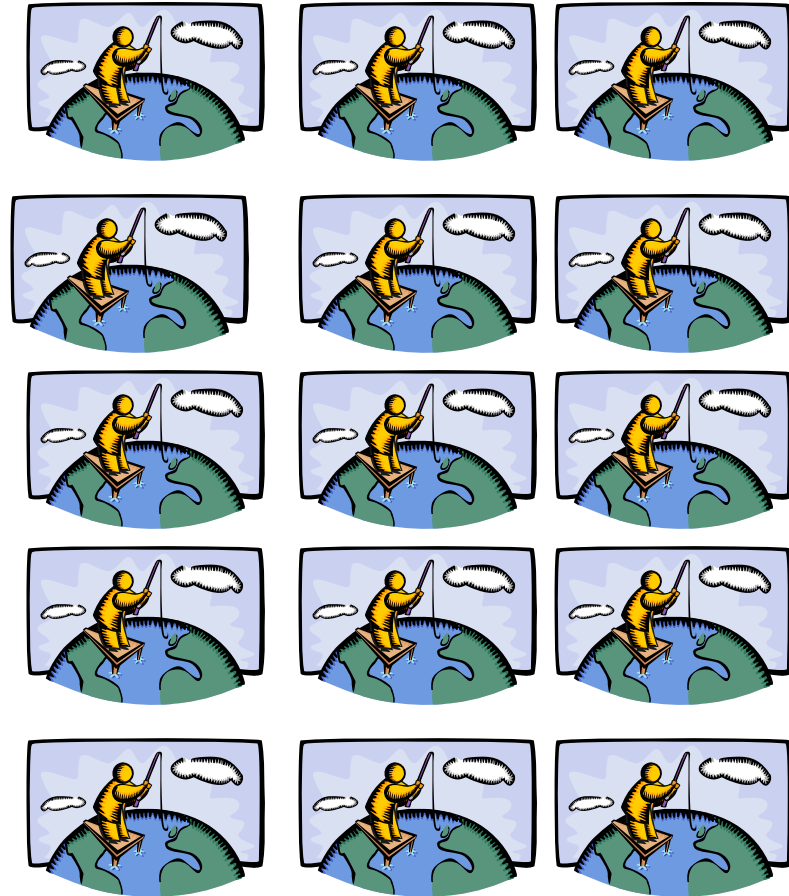
Parallelism



Parallelism

Parallelism means doing multiple things at the same time: you can get more work done in the same time.

Less fish ...

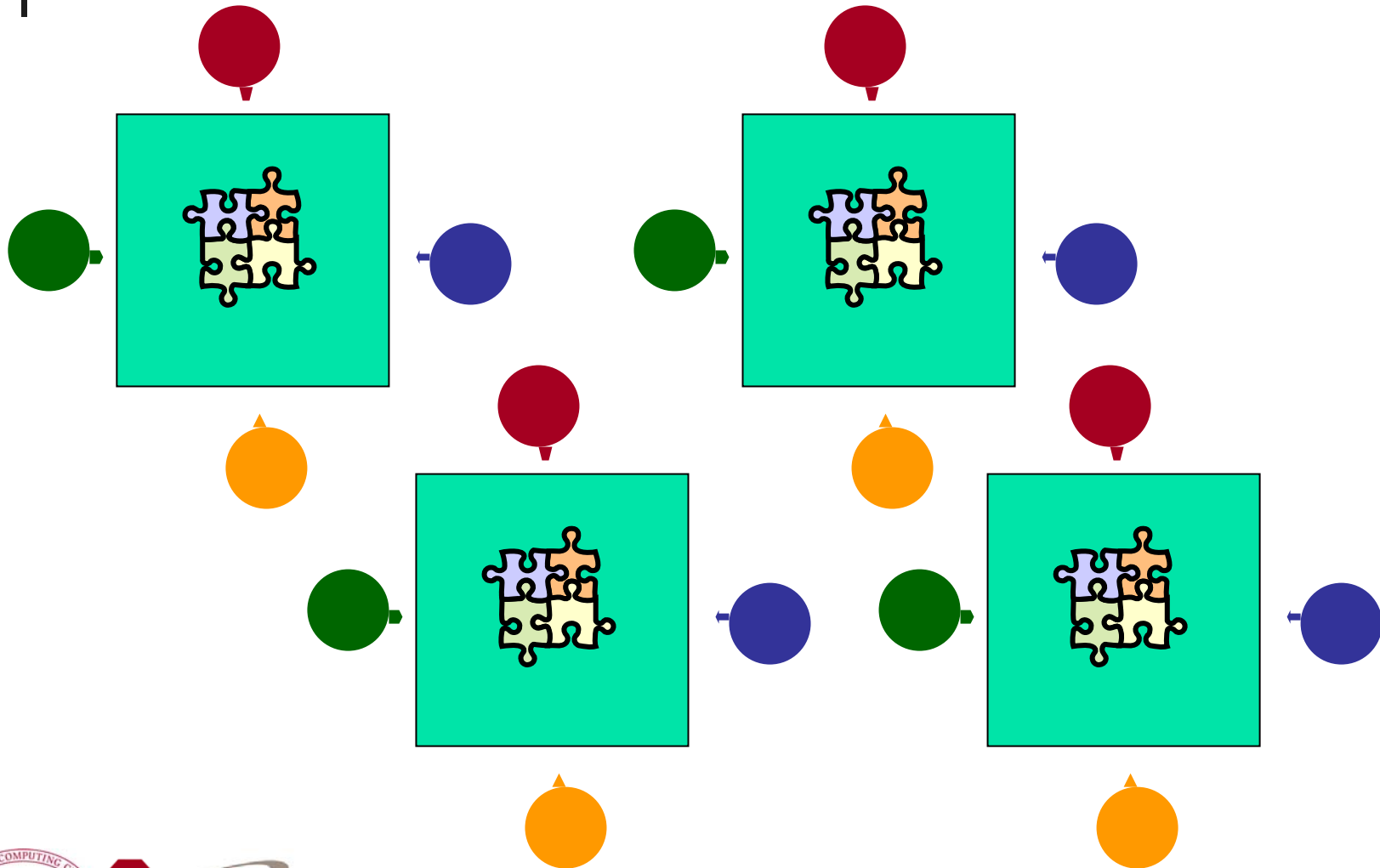


More fish!





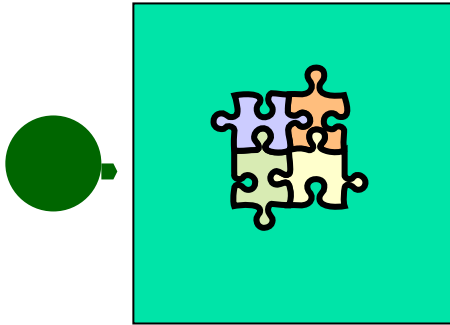
The Jigsaw Puzzle Analogy





Serial Computing

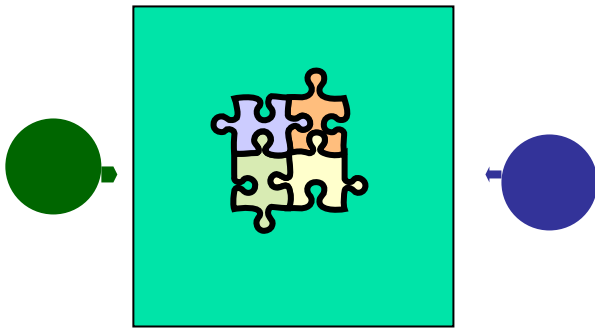
Suppose you want to do a jigsaw puzzle that has, say, a thousand pieces.



We can imagine that it'll take you a certain amount of time. Let's say that you can put the puzzle together in an hour.



Shared Memory Parallelism

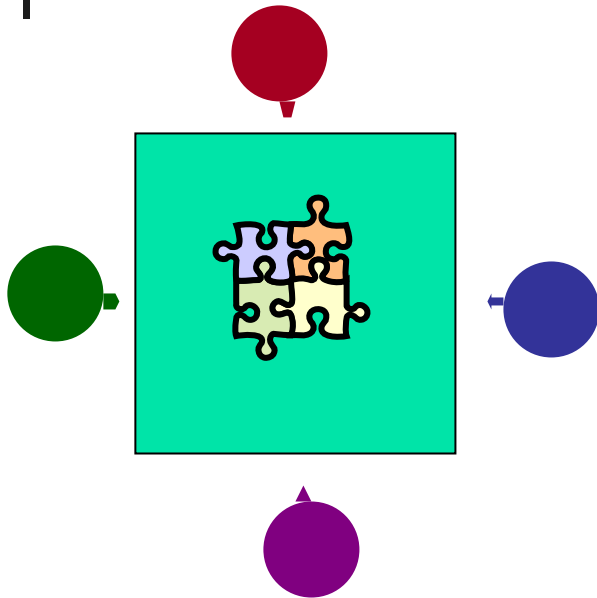


If Scott sits across the table from you, then he can work on his half of the puzzle and you can work on yours. Once in a while, you'll both reach into the pile of pieces at the same time (you'll contend for the same resource), which will cause a little bit of slowdown. And from time to time you'll have to work together (communicate) at the interface between his half and yours. The speedup will be nearly 2-to-1: y'all might take 35 minutes instead of 30.





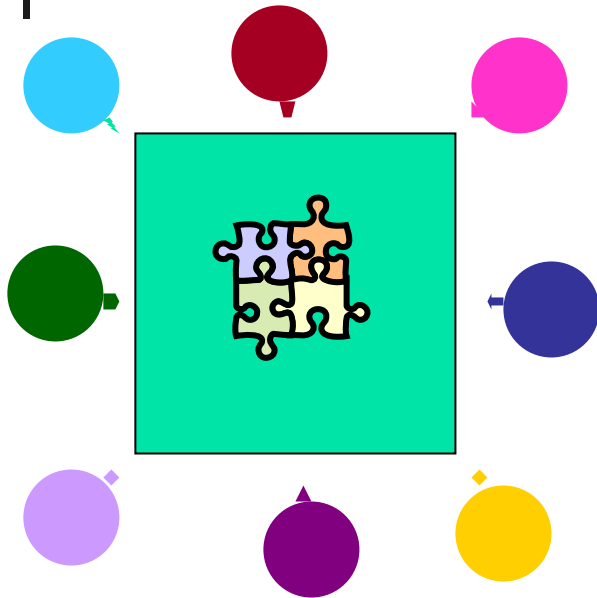
The More the Merrier?



Now let's put Paul and Charlie on the other two sides of the table. Each of you can work on a part of the puzzle, but there'll be a lot more contention for the shared resource (the pile of puzzle pieces) and a lot more communication at the interfaces. So y'all will get noticeably less than a 4-to-1 speedup, but you'll still have an improvement, maybe something like 3-to-1: the four of you can get it done in 20 minutes instead of an hour.



Diminishing Returns



If we now put Dave and Tom and Horst and Brandon on the corners of the table, there's going to be a whole lot of contention for the shared resource, and a lot of communication at the many interfaces. So the speedup y'all get will be much less than we'd like; you'll be lucky to get 5-to-1.

So we can see that adding more and more workers onto a shared resource is eventually going to have a diminishing return.



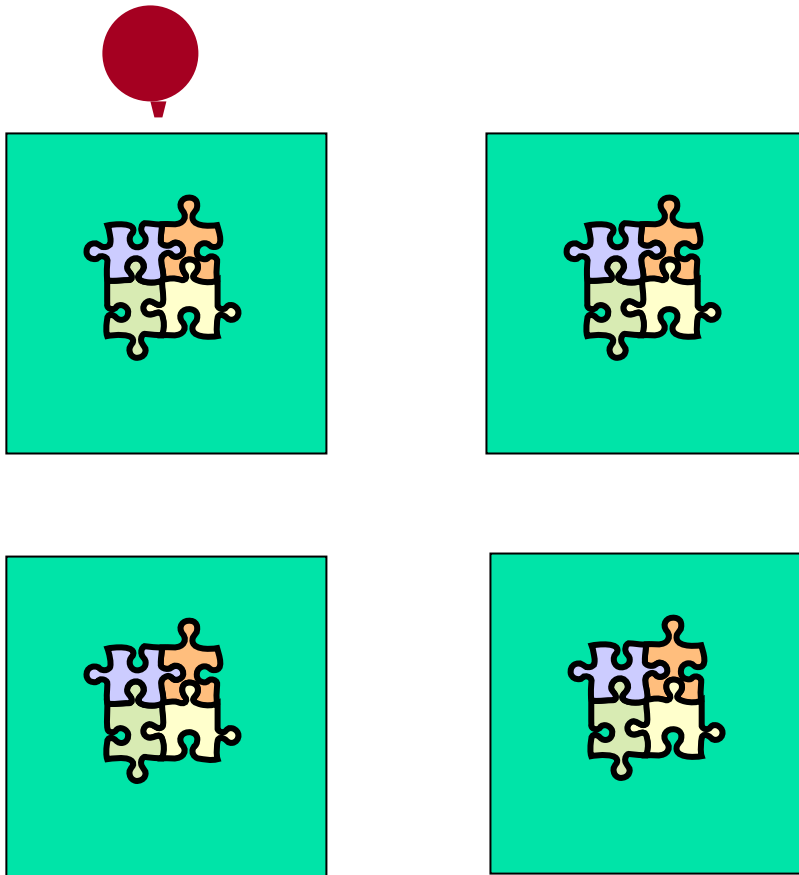
Distributed Parallelism



Now let's try something a little different. Let's set up two tables, and let's put you at one of them and Scott at the other. Let's put half of the puzzle pieces on your table and the other half of the pieces on Scott's. Now y'all can work completely independently, without any contention for a shared resource. **BUT**, the cost per communication is **MUCH** higher (you have to scootch your tables together), and you need the ability to split up (*decompose*) the puzzle pieces reasonably evenly, which may be tricky to do for some puzzles.



More Distributed Processors

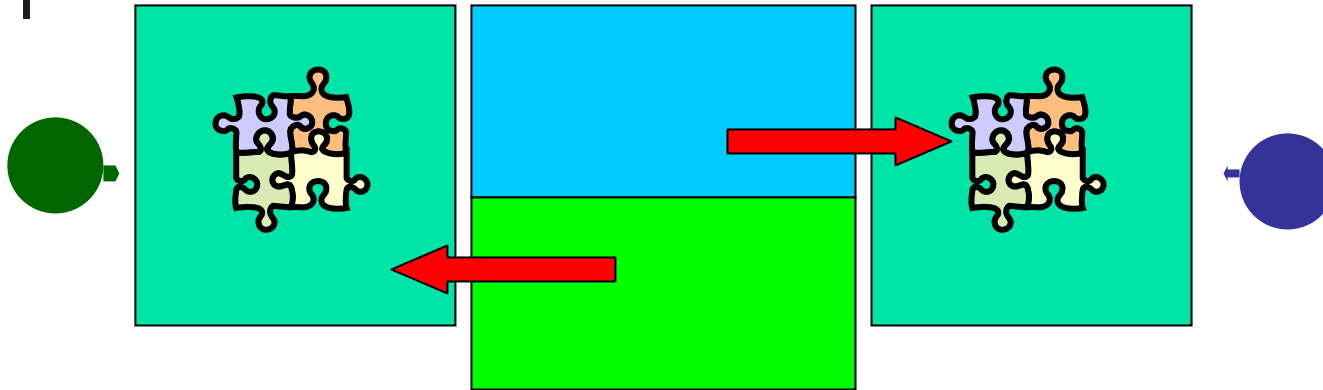


It's a lot easier to add more processors in distributed parallelism. But, you always have to be aware of the need to decompose the problem and to communicate among the processors. Also, as you add more processors, it may be harder to load balance the amount of work that each processor gets.





Load Balancing

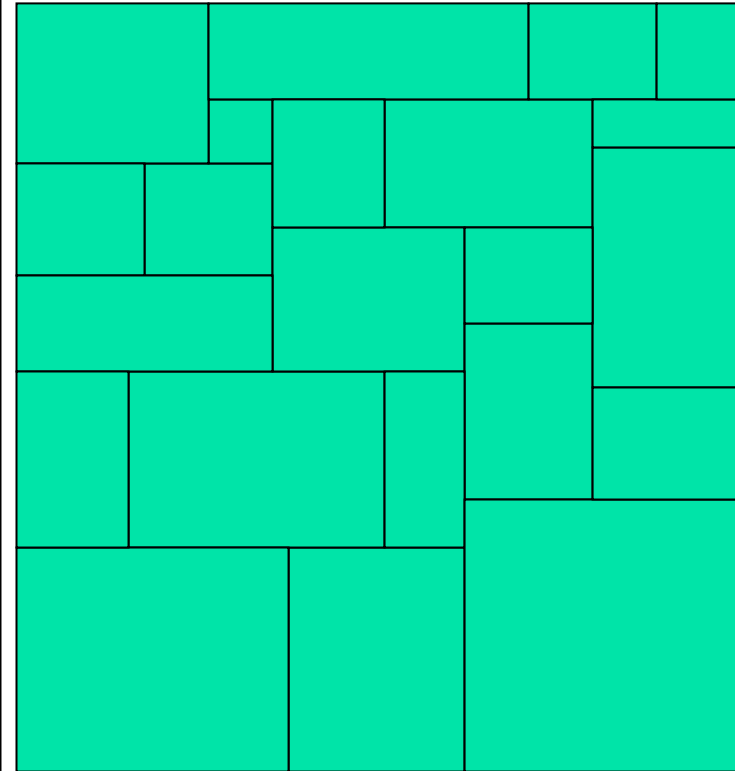
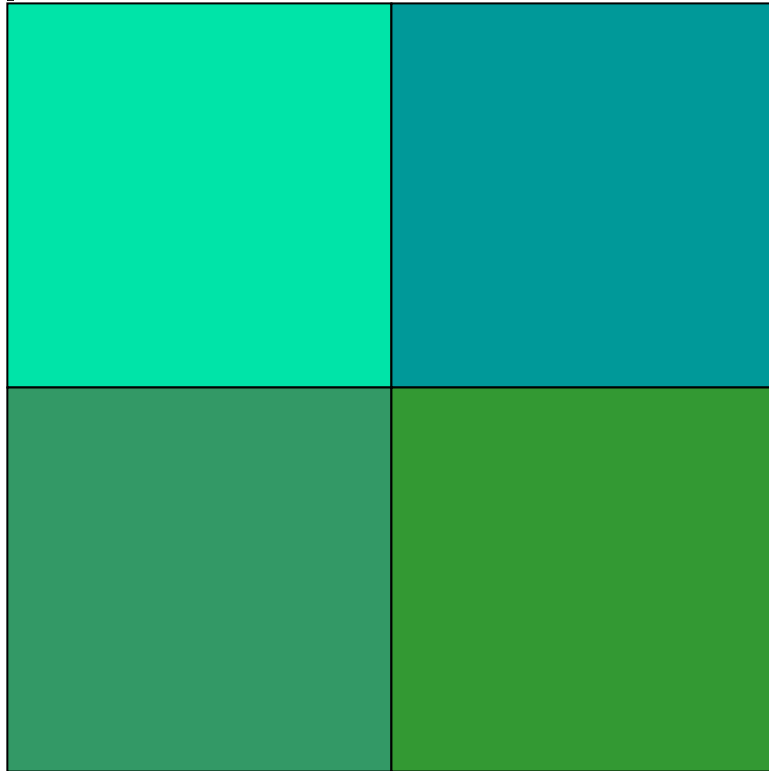


Load balancing means ensuring that everyone completes their workload at roughly the same time.

For example, if the jigsaw puzzle is half grass and half sky, then you can do the grass and Scott can do the sky, and then y'all only have to communicate at the horizon – and the amount of work that each of you does on your own is roughly equal. So you'll get pretty good speedup.



Load Balancing

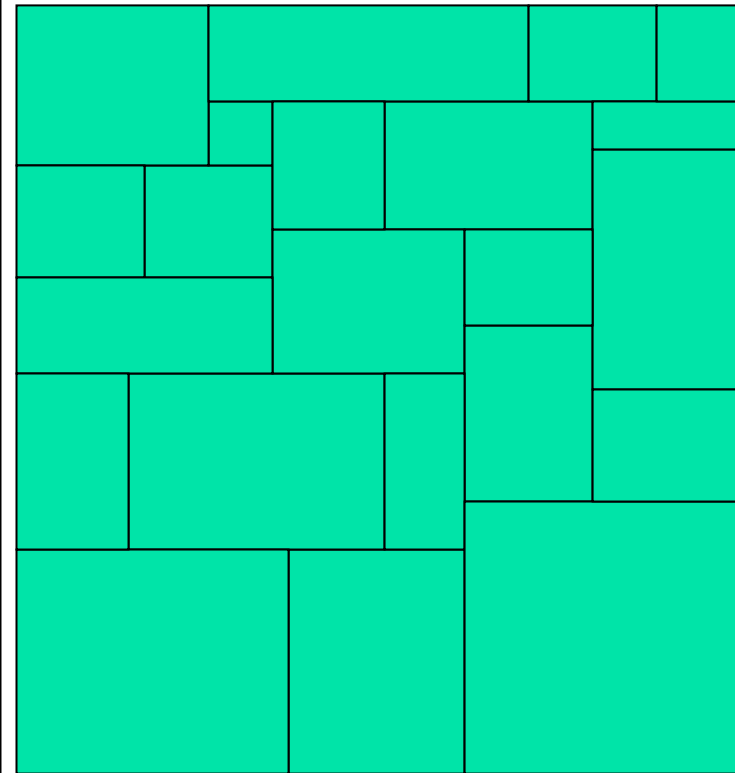
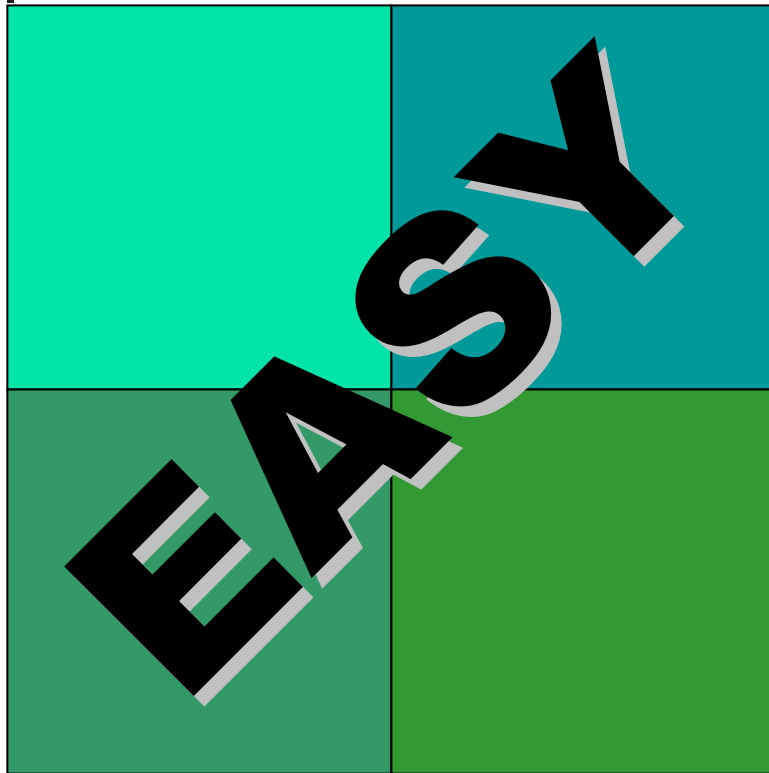


Load balancing can be easy, if the problem splits up into chunks of roughly equal size, with one chunk per processor. Or load balancing can be very hard.





Load Balancing

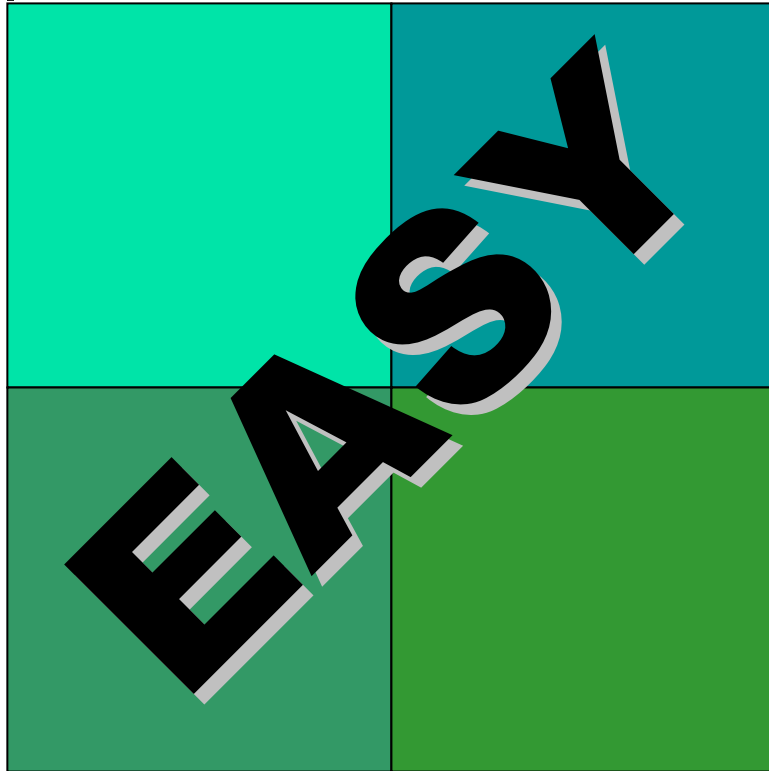


Load balancing can be easy, if the problem splits up into chunks of roughly equal size, with one chunk per processor. Or load balancing can be very hard.





Load Balancing



Load balancing can be easy, if the problem splits up into chunks of roughly equal size, with one chunk per processor. Or load balancing can be very hard.





Moore's Law



Moore's Law

In 1965, Gordon Moore was an engineer at Fairchild Semiconductor.

He noticed that the number of transistors that could be squeezed onto a chip was doubling about every 18 months.

It turns out that computer speed is roughly proportional to the number of transistors per unit area.

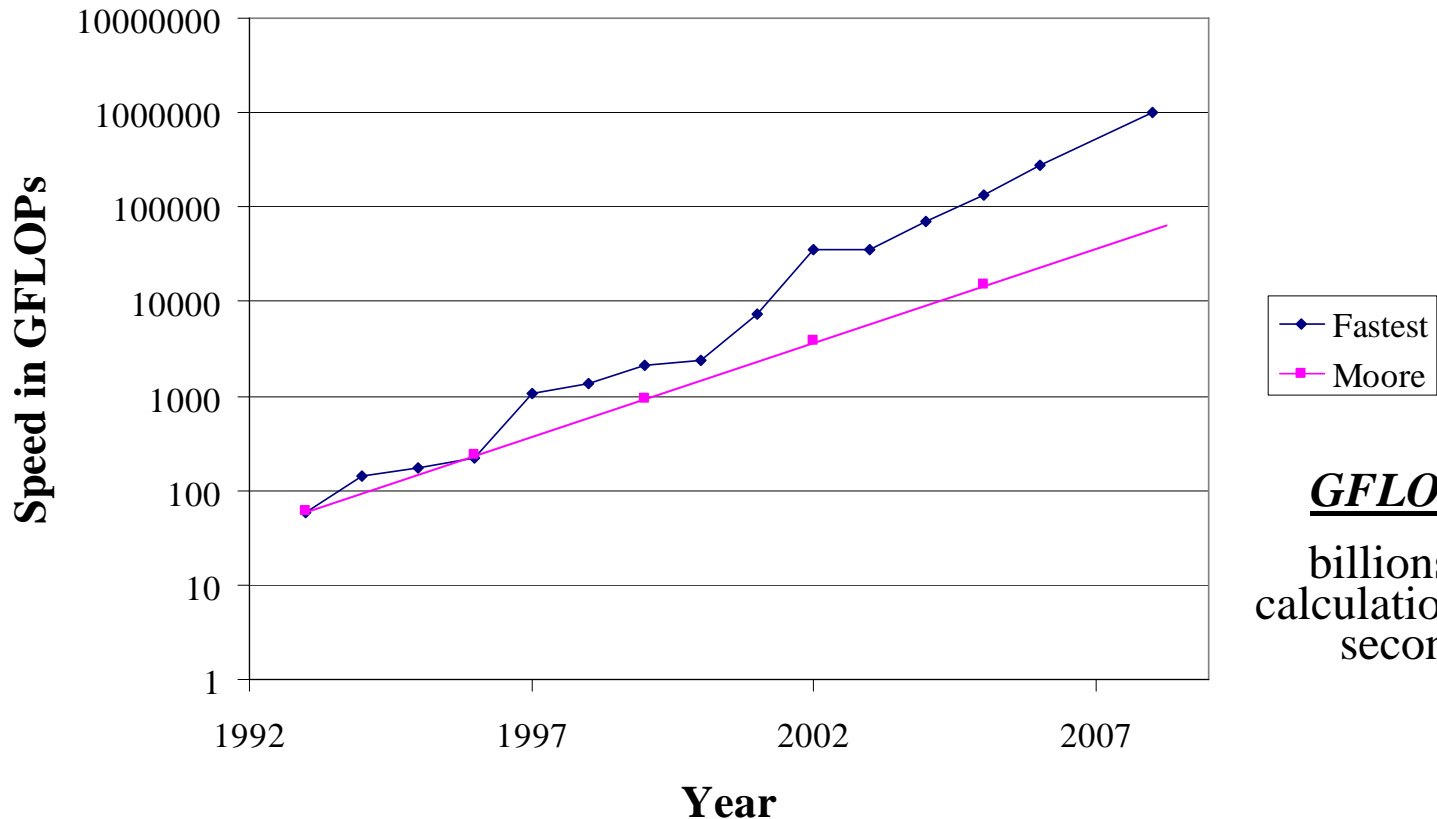
Moore wrote a paper about this concept, which became known as “*Moore's Law.*”





Fastest Supercomputer vs. Moore

Fastest Supercomputer in the World

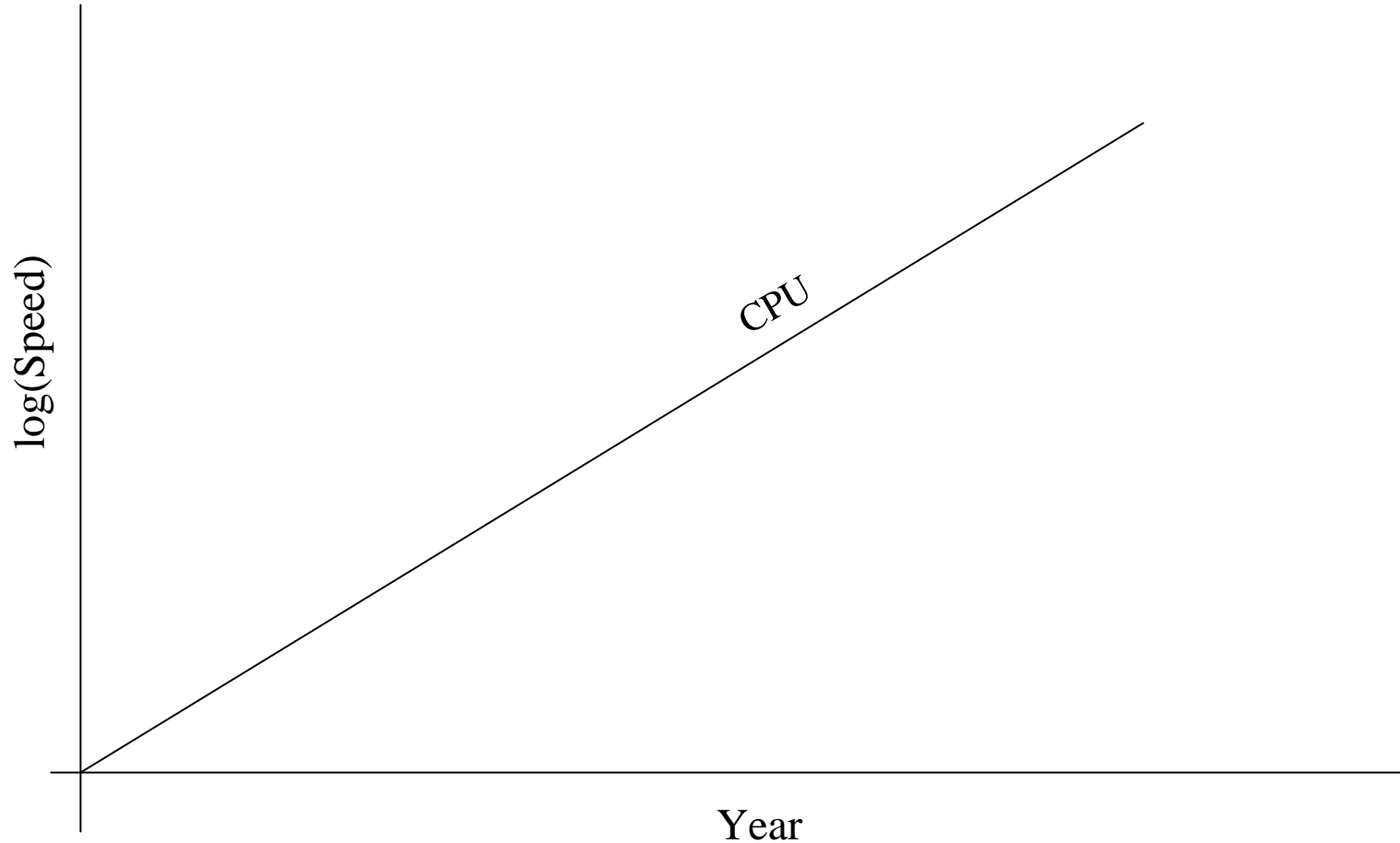


GFLOPs:
billions of
calculations per
second



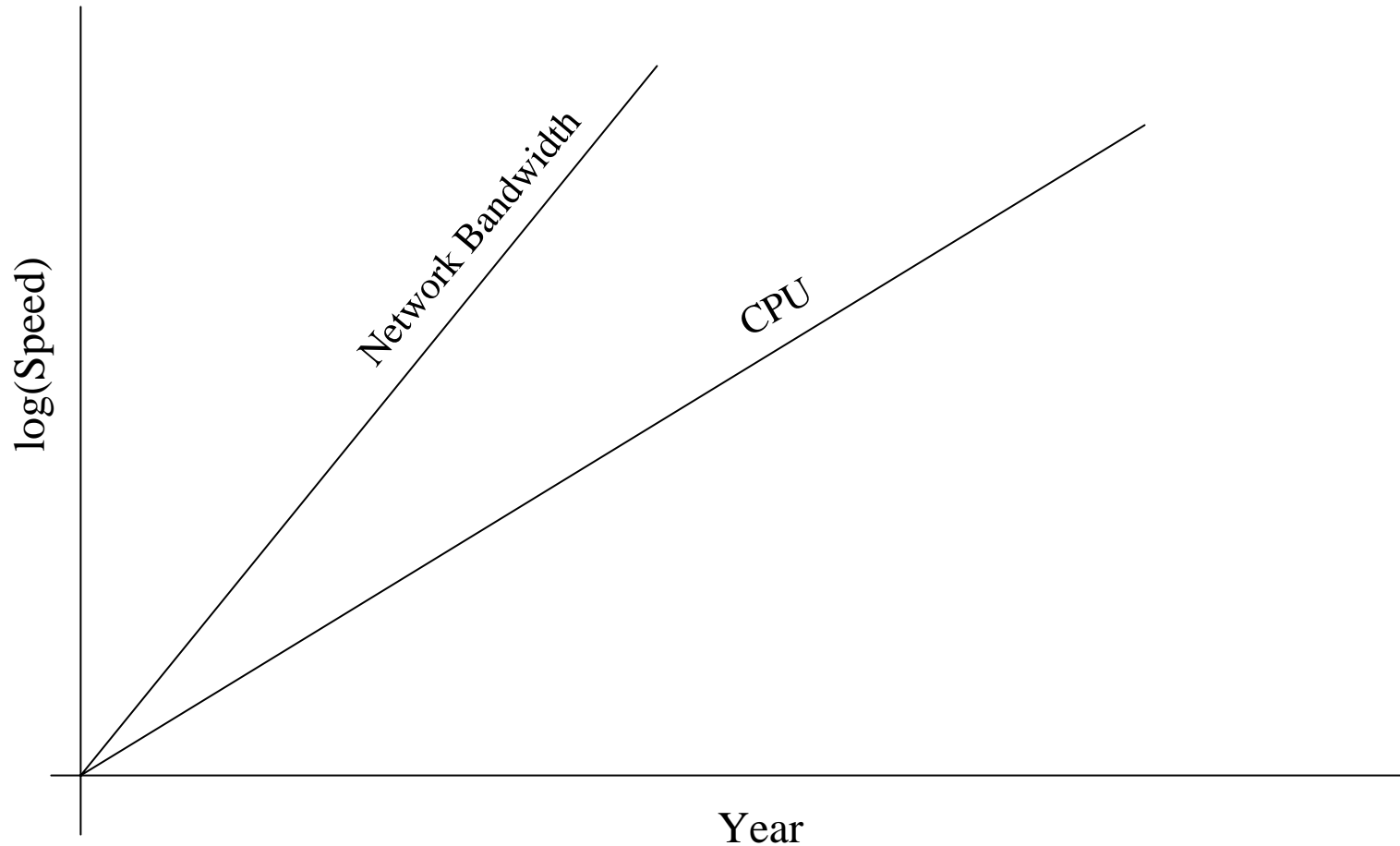


Moore's Law in Practice



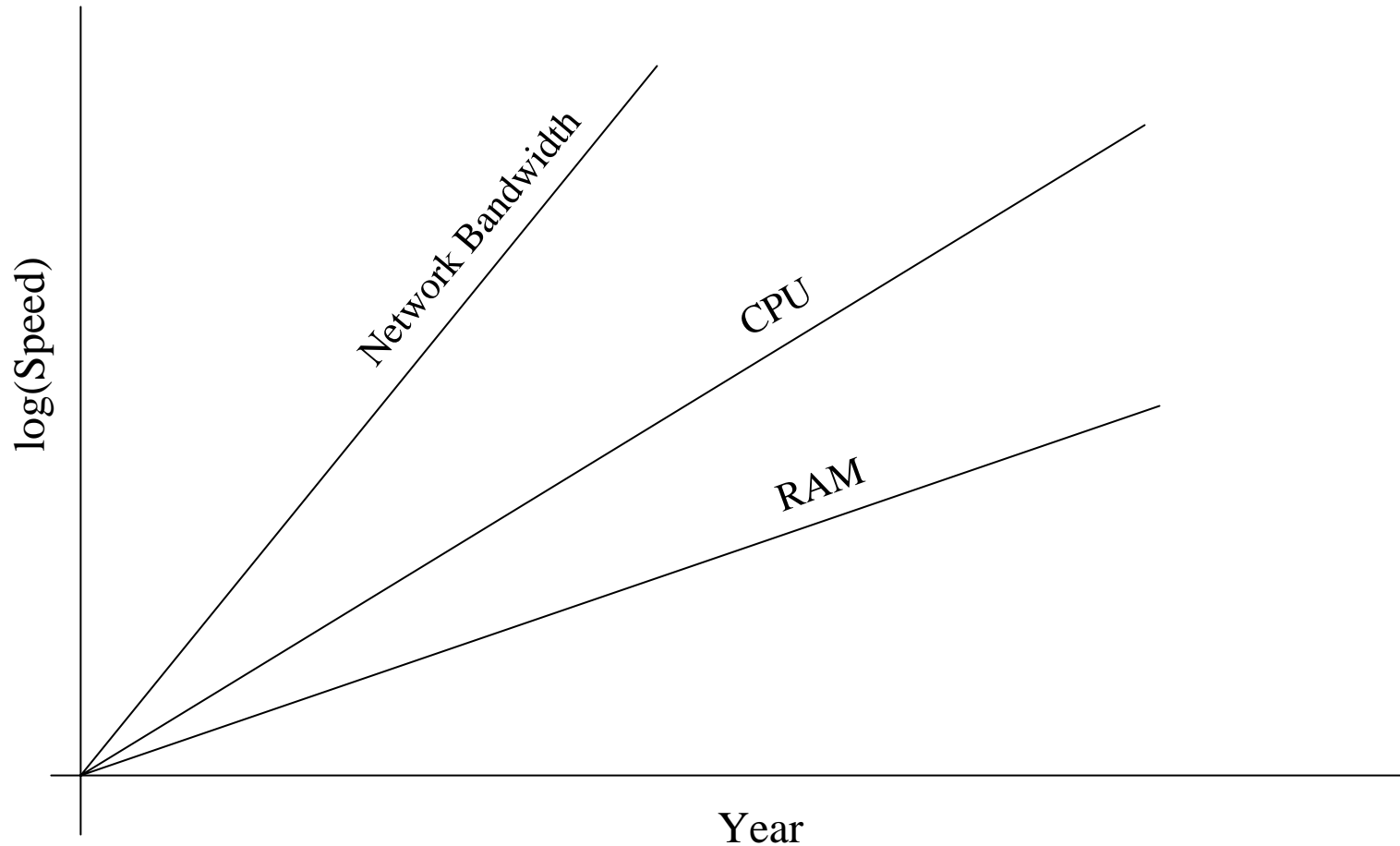


Moore's Law in Practice



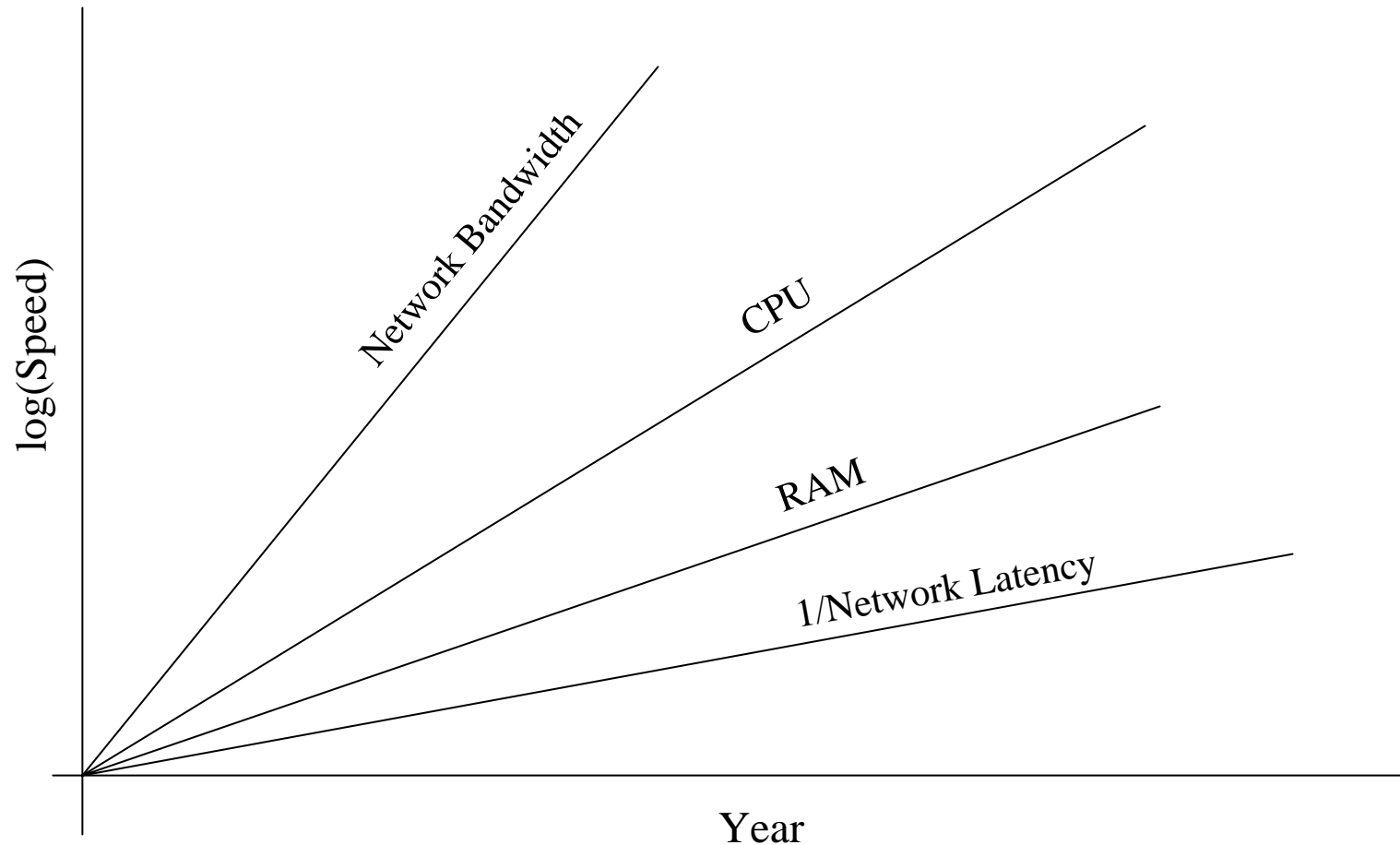


Moore's Law in Practice



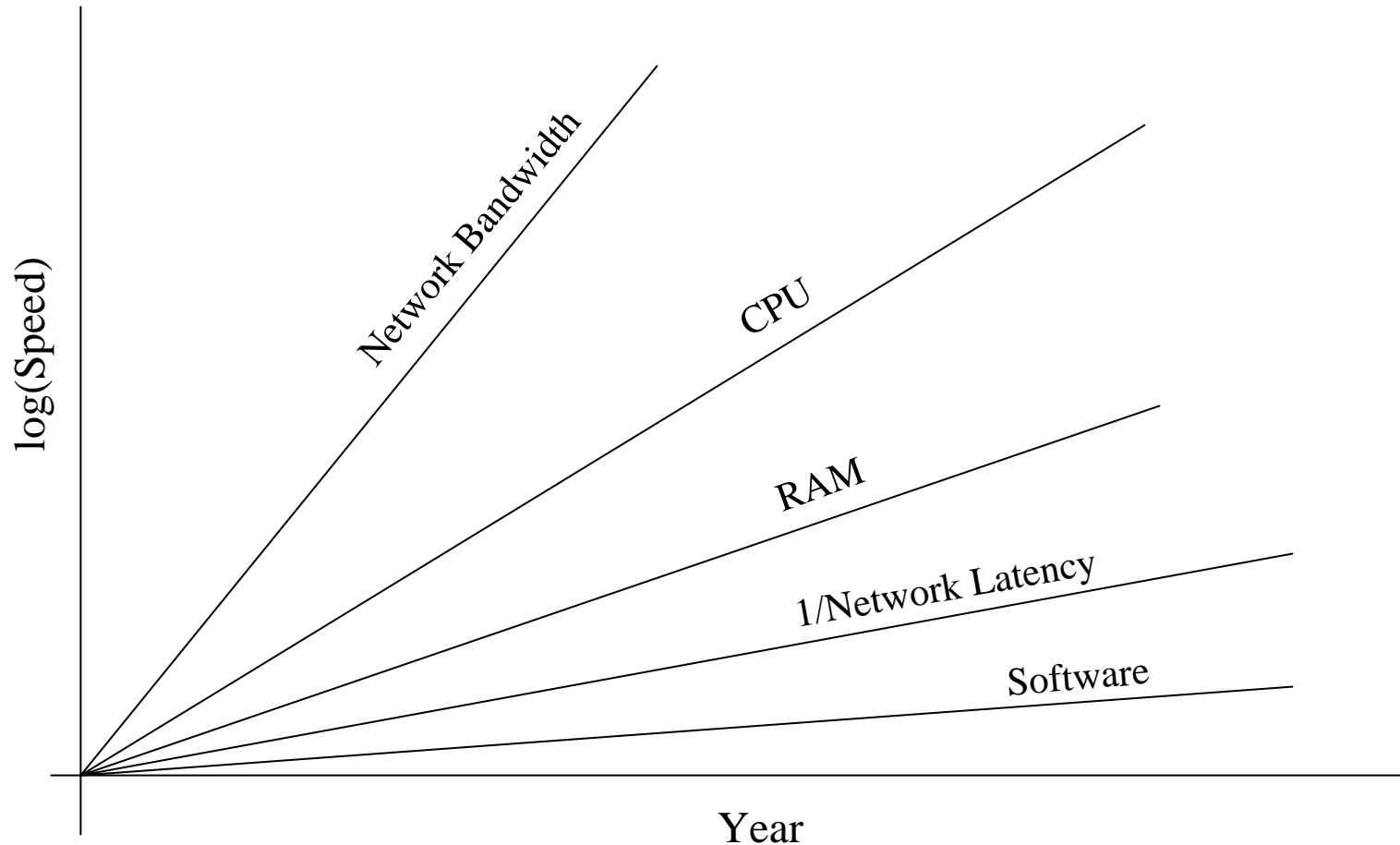


Moore's Law in Practice





Moore's Law in Practice





Why Bother?



Why Bother with HPC at All?

It's clear that making effective use of HPC takes quite a bit of effort, both learning how and developing software.

That seems like a lot of trouble to go to just to get your code to run faster.

It's nice to have a code that used to take a day, now run in an hour. But if you can afford to wait a day, what's the point of HPC?

Why go to all that trouble just to get your code to run faster?





Why HPC is Worth the Bother

- What HPC gives you that you won't get elsewhere is the ability to do bigger, better, more exciting science. If your code can run faster, that means that you can tackle much bigger problems in the same amount of time that you used to need for smaller problems.
- HPC is important not only for its own sake, but also because what happens in HPC today will be on your desktop in about 10 to 15 years: it puts you ahead of the curve.





The Future is Now

Historically, this has always been true:

Whatever happens in supercomputing today will be on your desktop in 10 – 15 years.

So, if you have experience with supercomputing, you'll be ahead of the curve when things get to the desktop.





OK Supercomputing Symposium 2009



2003 Keynote:
Peter Freeman
NSF
Computer &
Information
Science &
Engineering
Assistant Director



2004 Keynote:
Sangtae Kim
NSF Shared
Cyberinfrastructure
Division Director



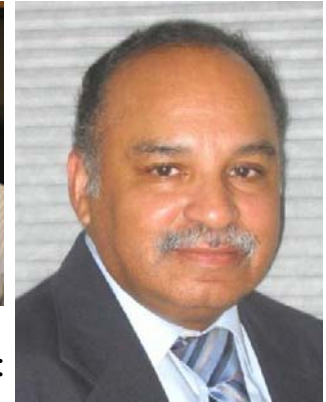
2005 Keynote:
Walt Brooks
NASA Advanced
Supercomputing
Division Director



2006 Keynote:
Dan Atkins
Head of NSF's
Office of
Cyber-
infrastructure



2007 Keynote:
Jay Boisseau
Director
Texas Advanced
Computing Center
U. Texas Austin



2008 Keynote:
José Muñoz
Deputy Office
Director/ Senior
Scientific Advisor
Office of Cyber-
infrastructure
National Science
Foundation



2009 Keynote:
Ed Seidel
Director
NSF Office of
Cyber-
infrastructure

FREE! Wed Oct 7 2009 @ OU
<http://symposium2009.oscer.ou.edu/>

Parallel Programming Workshop

FREE! Tue Oct 6 2009 @ OU

Sponsored by SC09 Education Program
FREE! Symposium Wed Oct 7 2009 @ OU

Supercomputing in Plain English: Overview
CS1313, Spring 2009





SC09 Summer Workshops

This coming summer, the SC09 Education Program, part of the SC09 (Supercomputing 2009) conference, is planning to hold two weeklong supercomputing-related workshops in Oklahoma, for **FREE** (except you pay your own transport):

- **At OSU Sun May 17 – the May 23:**
FREE Computational Chemistry for Chemistry Educators (2010 TENTATIVE: Computational Biology)
- **At OU Sun Aug 9 – Sat Aug 15:**
FREE Parallel Programming & Cluster Computing

We'll alert everyone when the details have been ironed out and the registration webpage opens.

Please note that you must **apply** for a seat, and acceptance **CANNOT** be guaranteed.





SC09 Summer Workshops

1. May 17-23: Oklahoma State U: Computational Chemistry
2. May 25-30: Calvin Coll (MI): Intro to Computational Thinking
3. June 7-13: U Cal Merced: Computational Biology
4. June 7-13: Kean U (NJ): Parallel Progrmg & Cluster Comp
5. June 14-20: Widener U (PA): Computational Physics
6. July 5-11: Atlanta U Ctr: Intro to Computational Thinking
7. July 5-11: Louisiana State U: Parallel Progrmg & Cluster Comp
8. July 12-18: U Florida: Computational Thinking Grades 6-12
9. July 12-18: Ohio Supercomp Ctr: Computational Engineering
10. Aug 2- 8: U Arkansas: Intro to Computational Thinking
11. Aug 9-15: U Oklahoma: Parallel Progrmg & Cluster Comp



**Thanks for your
attention!**



Questions?

www.oscer.ou.edu



References

- [1] Image by Greg Bryan, Columbia U.
- [2] “[Update on the Collaborative Radar Acquisition Field Test \(CRAFT\): Planning for the Next Steps.](#)”
Presented to NWS Headquarters August 30 2001.
- [3] See <http://hneeman.oscer.ou.edu/hamr.html> for details.
- [4] <http://www.dell.com/>
- [5] <http://www.vw.com/newbeetle/>
- [6] Richard Gerber, The Software Optimization Cookbook: High-performance Recipes for the Intel Architecture. Intel Press, 2002, pp. 161-168.
- [7] RightMark Memory Analyzer. <http://cpu.rightmark.org/>
- [8] <ftp://download.intel.com/design/Pentium4/papers/24943801.pdf>
- [9] <http://www.seagate.com/cda/products/discsales/personal/family/0,1085,621,00.html>
- [10] http://www.samsung.com/Products/OpticalDiscDrive/SlimDrive/OpticalDiscDrive_SlimDrive_SN_S082D.asp?page=Specifications
- [11] <ftp://download.intel.com/design/Pentium4/manuals/24896606.pdf>
- [12] <http://www.pricewatch.com/>

