Supercomputing in Plain English

Overview:
What the Heck is Supercomputing?

Henry Neeman, Director

Director, OU Supercomputing Center for Education & Research (OSCER)
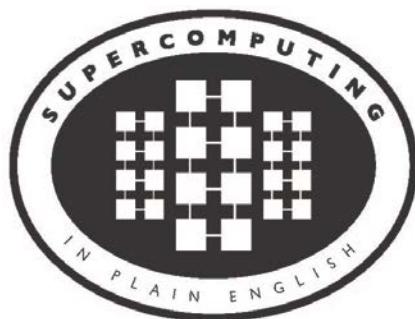**Assistant Vice President, Information Technology – Research Strategy Advisor**
Associate Professor, College of Engineering
Adjunct Associate Professor, School of Computer Science
University of Oklahoma
Tuesday January 20 2015

This is an experiment!

It's the nature of these kinds of videoconferences that
FAILURES ARE GUARANTEED TO HAPPEN!
NO PROMISES!

So, please bear with us. Hopefully everything will work out well enough.

If you lose your connection, you can retry the same kind of connection, or try connecting another way.

Remember, if all else fails, you always have the toll free phone bridge to fall back on.

PLEASE MUTE YOURSELF

No matter how you connect, **PLEASE MUTE YOURSELF**, so that we cannot hear you.

At OU, we will turn off the sound on all conferencing technologies.

That way, we won't have problems with echo cancellation.

Of course, that means we cannot hear questions.

So for questions, you'll need to send e-mail.

PLEASE MUTE YOURSELF.

PLEASE MUTE YOURSELF.

Download the Slides Beforehand

Before the start of the session, please download the slides from the Supercomputing in Plain English website:

`http://www.oscer.ou.edu/education/`

That way, if anything goes wrong, you can still follow along with just audio.

PLEASE MUTE YOURSELF.

H.323 (Polycom etc) #1

If you want to use H.323 videoconferencing – for example, Polycom – then:

- If you AREN'T registered with the OneNet gatekeeper (which is probably the case), then:
  - Dial `164.58.250.47`
  - Bring up the virtual keypad.
    On some H.323 devices, you can bring up the virtual keypad by typing:
    `#`
    (You may want to try without first, then with; some devices won't work with the #, but give cryptic error messages about it.)
  - When asked for the conference ID, or if there's no response, enter:
    `0409`
  - On most but not all H.323 devices, you indicate the end of the ID with:
    `#`

H.323 (Polycom etc) #2

If you want to use H.323 videoconferencing – for example, Polycom – then:

- If you ARE already registered with the OneNet gatekeeper (most institutions aren't), dial:

    `2500409`

Many thanks to James Deaton, Skyler Donahue, Jeremy Wright and Steven Haldeman of OneNet for providing this.

PLEASE MUTE YOURSELF.

Wowza #1

You can watch from a Windows, MacOS or Linux laptop using Wowza from the following URL:

<div align="center">

`http://jwplayer.onenet.net/stream6/sipe.html`

</div>

Wowza behaves a lot like YouTube, except live.

Many thanks to James Deaton, Skyler Donahue, Jeremy Wright and Steven Haldeman of OneNet for providing this.

PLEASE MUTE YOURSELF.

Wowza #2

Wowza has been tested on multiple browsers on each of:

- Windows (7 and 8): IE, Firefox, Chrome, Opera, Safari
- MacOS X: Safari, Firefox
- Linux: Firefox, Opera

PLEASE MUTE YOURSELF.

Toll Free Phone Bridge

IF ALL ELSE FAILS, you can use our toll free phone bridge:

800-832-0736

* 623 2874 #

Please mute yourself and use the phone to listen.

Don't worry, we'll call out slide numbers as we go.

Please use the phone bridge **ONLY** if you cannot connect any other way: the phone bridge can handle only 100 simultaneous connections, and we have over 500 participants.

Many thanks to OU CIO Loretta Early for providing the toll free phone bridge.

PLEASE MUTE YOURSELF.

Please Mute Yourself

No matter how you connect, **PLEASE MUTE YOURSELF**, so that we cannot hear you.

(For Wowza, you don't need to do that, because the information only goes from us to you, not from you to us.)

At OU, we will turn off the sound on all conferencing technologies.

That way, we won't have problems with echo cancellation.

Of course, that means we cannot hear questions.

So for questions, you'll need to send e-mail.

PLEASE MUTE YOURSELF.

PLEASE MUTE YOURSELF.

Questions via E-mail Only

Ask questions by sending e-mail to:

<u>sipe2015@gmail.com</u>

All questions will be read out loud and then answered out loud.

PLEASE MUTE YOURSELF.

Onsite: Talent Release Form

If you're attending onsite, you **<u>MUST</u>** do one of the following:

- complete and sign the Talent Release Form,

OR

- sit behind the cameras (where you can't be seen) and don't talk at all.

If you aren't onsite, then **PLEASE MUTE YOURSELF.**

TENTATIVE Schedule

Tue Jan 20: Overview: What the Heck is Supercomputing?
Tue Jan 27: The Tyranny of the Storage Hierarchy
Tue Feb 3: Instruction Level Parallelism
Tue Feb 10: Stupid Compiler Tricks
Tue Feb 17: Shared Memory Multithreading
Tue Feb 24: Distributed Multiprocessing
Tue March 3: Applications and Types of Parallelism
Tue March 10: Multicore Madness
Tue March 17: **NO SESSION** (OU's Spring Break)
Tue March 24: **NO SESSION** (Henry has a huge grant proposal due)
Tue March 31: High Throughput Computing
Tue Apr 7: GPGPU: Number Crunching in Your Graphics Card
Tue Apr 14: Grab Bag: Scientific Libraries, I/O Libraries, Visualization

Thanks for helping!

- OU IT
    - OSCER operations staff (Brandon George, Dave Akin, Brett Zimmerman, Josh Alexander, Patrick Calhoun)
    - Horst Severini, OSCER Associate Director for Remote & Heterogeneous Computing
    - Debi Gentis, OSCER Coordinator
    - Jim Summers
    - The OU IT network team
- James Deaton, Skyler Donahue, Jeremy Wright and Steven Haldeman, OneNet
- Kay Avila, U Iowa
- Stephen Harrell, Purdue U

This is an experiment!

It's the nature of these kinds of videoconferences that
**FAILURES ARE GUARANTEED TO HAPPEN!
NO PROMISES!**

So, please bear with us. Hopefully everything will work out well enough.

If you lose your connection, you can retry the same kind of connection, or try connecting another way.

Remember, if all else fails, you always have the toll free phone bridge to fall back on.

PLEASE MUTE YOURSELF.

Coming in 2015!

Red Hat Tech Day, Thu Jan 22 2015 @ OU

    http://goo.gl/forms/jORZCz9xh7

Linux Clusters Institute workshop May 18-22 2015 @ OU

    http://www.linuxclustersinstitute.org/workshops/

Great Plains Network Annual Meeting, May 27-29, Kansas City

Advanced Cyberinfrastructure Research & Education Facilitators (ACI-REF) Virtual Residency May 31 - June 6 2015

XSEDE2015, July 26-30, St. Louis MO

    https://conferences.xsede.org/xsede15

IEEE Cluster 2015, Sep 23-27, Chicago IL

    http://www.mcs.anl.gov/ieeecluster2015/

OKLAHOMA SUPERCOMPUTING SYMPOSIUM 2015, **Sep 22-23 2015** @ OU

SC13, Nov 15-20 2015, Austin TX

    http://sc15.supercomputing.org/

PLEASE MUTE YOURSELF.

Supercomputing in Plain English

People

Things

Thanks for your attention!

Questions?
www.oscer.ou.edu

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
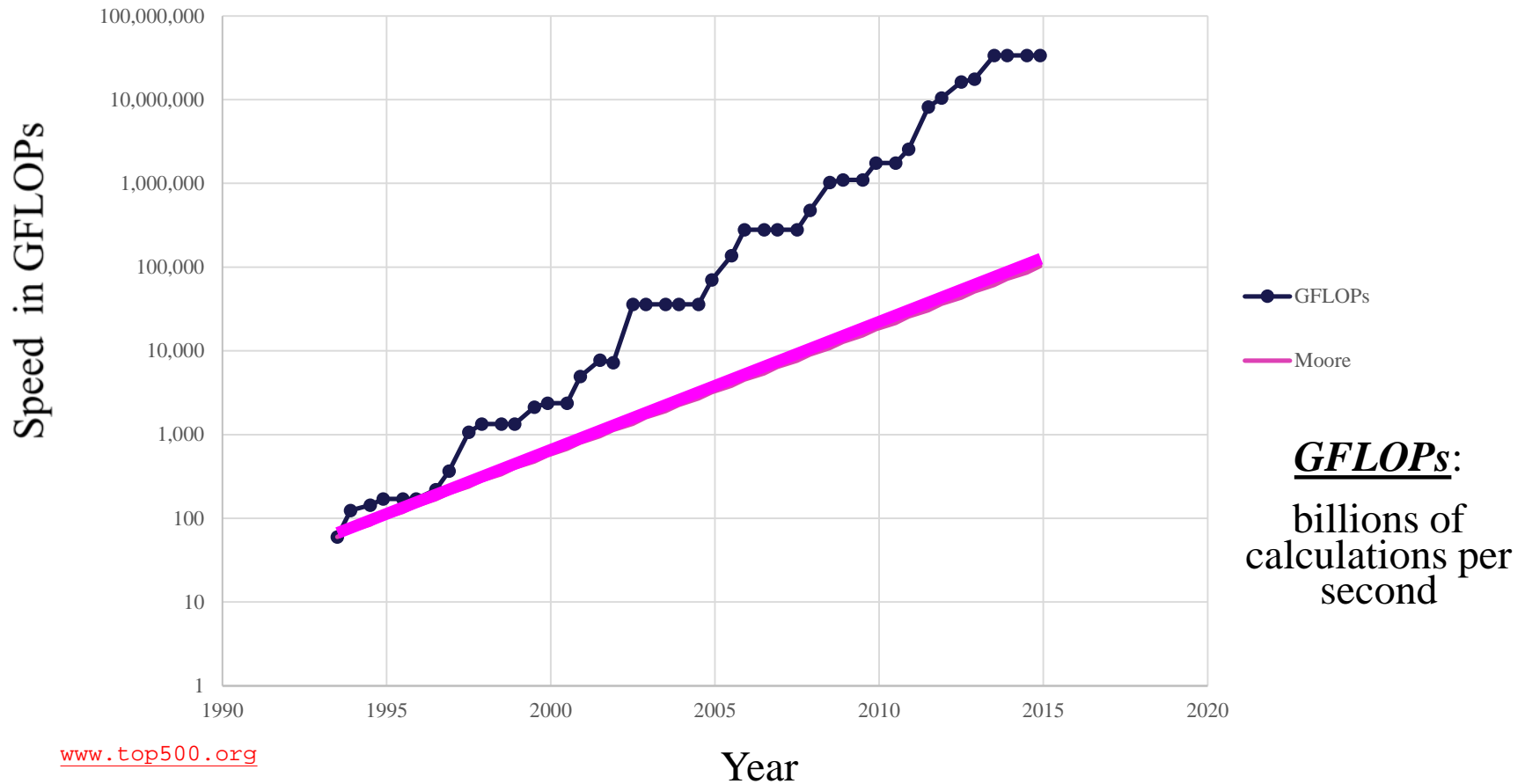
Fastest Supercomputer in the World vs Moore

GFLOPs:

billions of calculations per second

Supercomputing in Plain English: Overview
Tue Jan 20 2015

What is Supercomputing About?
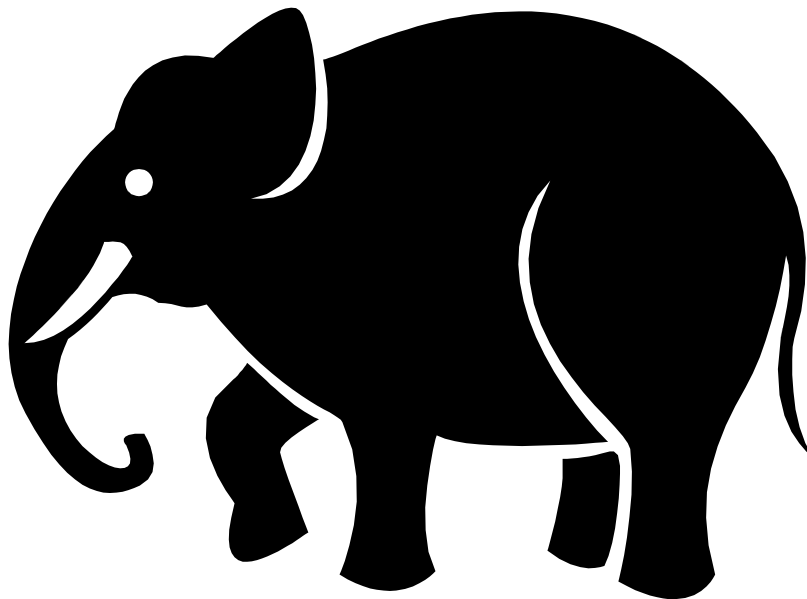
Size

Speed
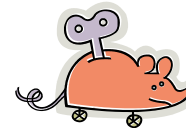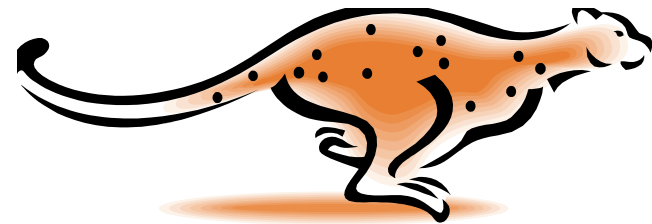
Laptop

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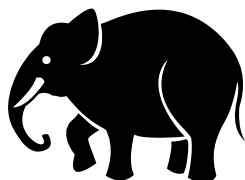
- **Speed**: Many problems that are interesting to scientists and engineers would take a very very long time to run on a PC: months or even years. But a problem that would take **a month on a PC** might take only **an hour on a supercomputer**.

What Is HPC Used For?

- *Simulation* of physical phenomena, such as
  - Weather forecasting
  - Galaxy formation
  - Oil reservoir management

  [1]

- *Data mining*: finding **needles** of information in a **haystack** of data, such as
  - Gene sequencing
  - Signal processing
  - Detecting storms that might produce tornados

Moore, OK Tornadic Storm

May 3 1999[2]

- *Visualization*: turning a vast sea of **data** into **pictures** that a scientist can understand
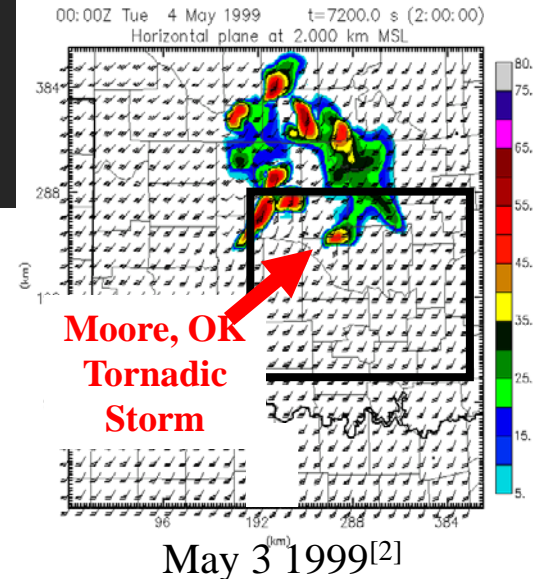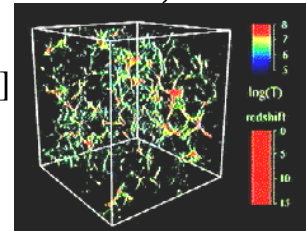
[3]
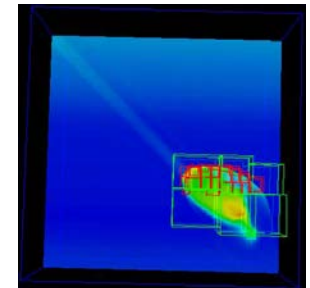
Supercomputing Issues

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

What is a Cluster Supercomputer?

"… [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"

http://lh3.ggpht.com/_6hgSmco4R9M/SfpFA3057zI/AAAAAAAACSg/G-AGCgLrQOk/s1600-h/pirates%5B5%5D.jpg

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
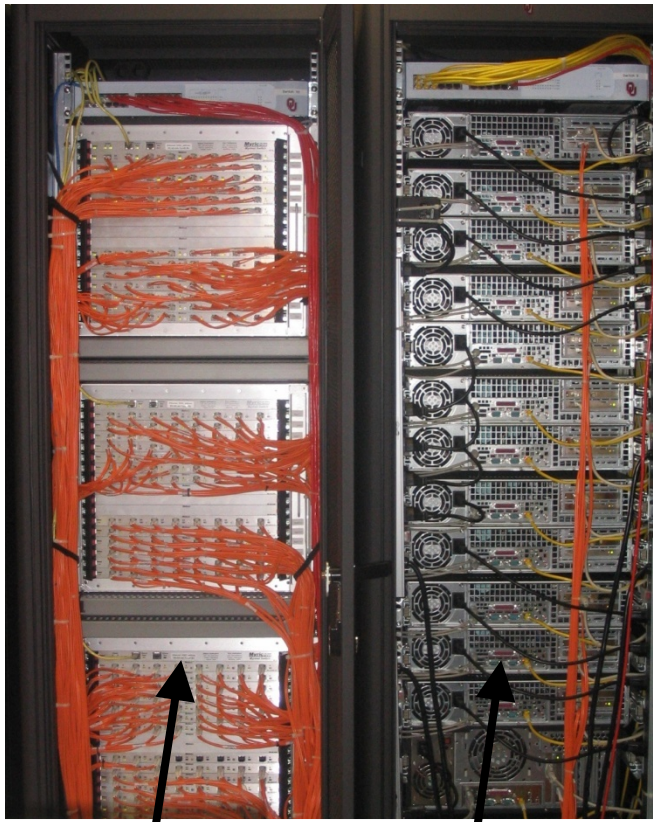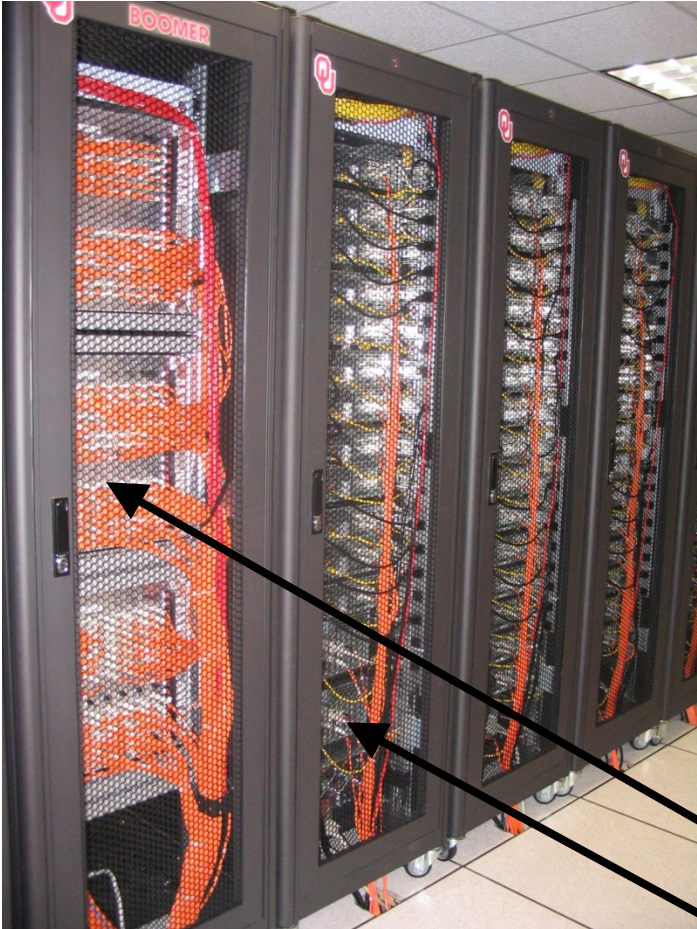
Interconnect

Nodes

Also named Boomer, in service 2002-5.

A Quick Primer on Hardware

Henry's Laptop

Dell Latitude E5540[4]

- Intel Core i3-4010U
dual core, 1.7 GHz, 3 MB L3 Cache
- 12 GB 1600 MHz DDR3L SDRAM
- 340 GB SATA 5400 RPM Hard Drive
- DVD+RW/CD-RW Drive
- 1 Gbps Ethernet Adapter

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
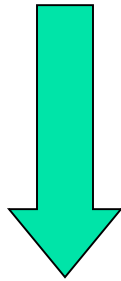
Slow, cheap, a lot

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

[5]

RAM is Slow

CPU 653 GB/sec

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.

Bottleneck

15 GB/sec (2.3%)
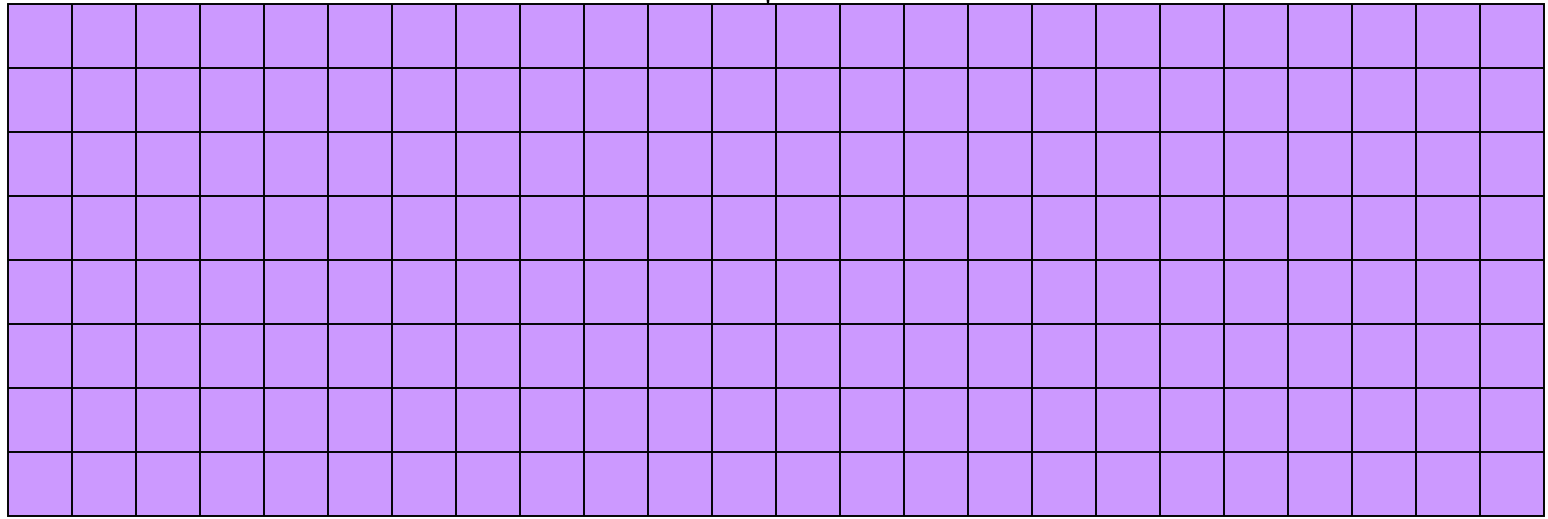
Why Have Cache?

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
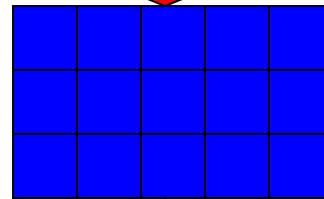
46 GB/sec (7%)

15 GB/sec (2.3%)
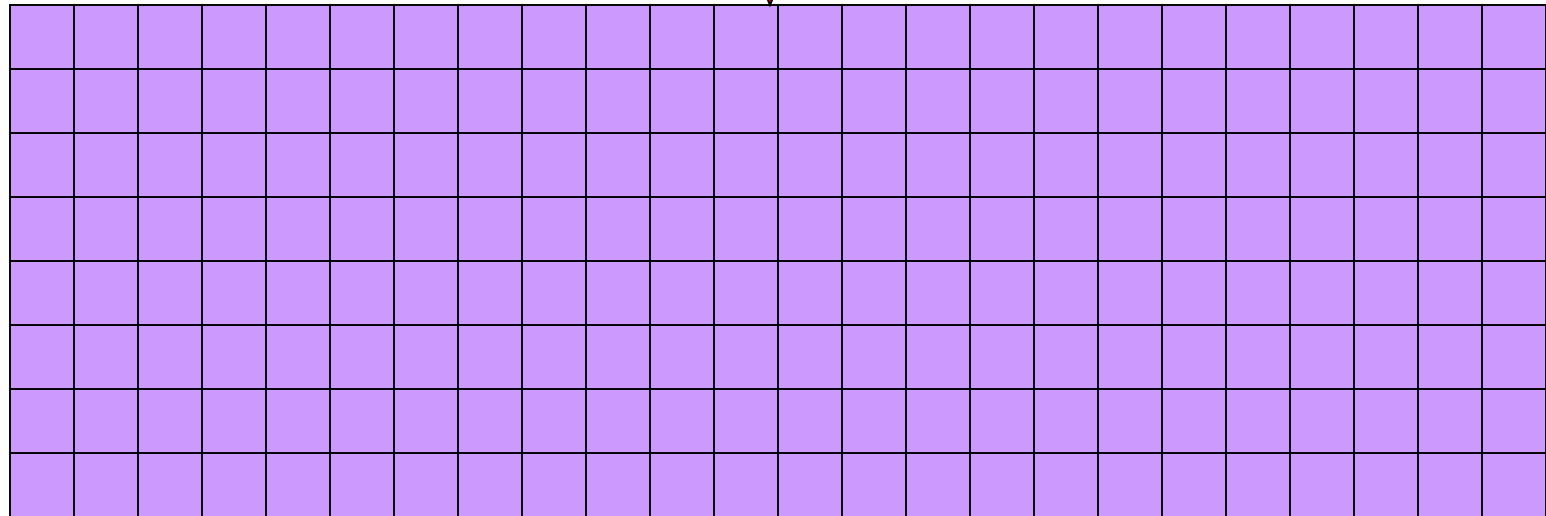
Henry's Laptop

Dell Latitude E5540[4]

- Intel Core i3-4010U
  dual core, 1.7 GHz, 3 MB L3 Cache
- 12 GB 1600 MHz DDR3L SDRAM
- 340 GB SATA 5400 RPM Hard Drive
- DVD+RW/CD-RW Drive
- 1 Gbps Ethernet Adapter

Supercomputing in Plain English: Overview
Tue Jan 20 2015

Storage Speed, Size, Cost

Henry's Laptop	Registers (Intel Core2 Duo 1.6 GHz)	Cache Memory (L3)	Main Memory (1600MHz DDR3L SDRAM)	Hard Drive	Ethernet (1000 Mbps)	DVD+R (16x)	Phone Modem (56 Kbps)
Speed (MB/sec) [peak]	668,672[6] (16 GFLOP/s*)	46,000	15,000 [7]	100[9]	125	32 [10]	0.007
Size (MB)	464 bytes** [11]	3	12,288 4096 times as much as cache	340,000	unlimited	unlimited	unlimited
Cost ($/MB)	–	$38 [12]	$0.0084 [12] ~1/4500 as much as cache	$0.00003 [12]	charged per month (typically)	$0.000045 [12]	charged per month (typically)

* GFLOP/s: billions of floating point operations per second
** 16 64-bit general purpose registers, 8 80-bit floating point registers,
16 128-bit floating point vector registers

Why the Storage Hierarchy?

Why does the Storage Hierarchy always work? Why are faster forms of storage more expensive and slower forms cheaper?

Proof by contradiction:

Suppose there were a storage technology that was **slow** and **expensive**.

How much of it would you buy?

Comparison
- Zip: 100 MB Cartridge $6.50 ($0.065 per MB), speed 2.4 MB/sec
- Blu-Ray: 25 GB Disk ~$1 ($0.00004 per MB), speed 72 MB/sec
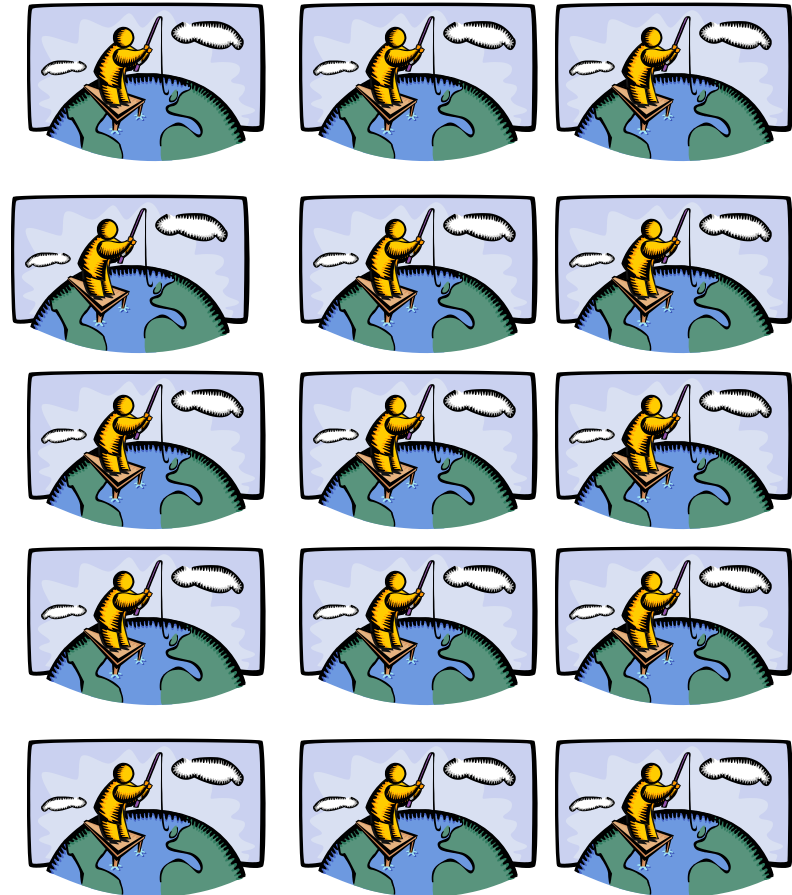
Not surprisingly, no one buys Zip drives any more.

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!

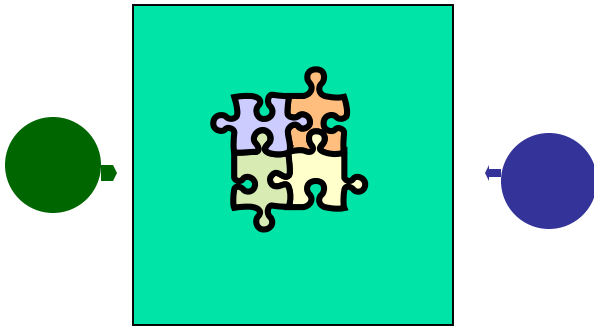# The Jigsaw Puzzle Analogy
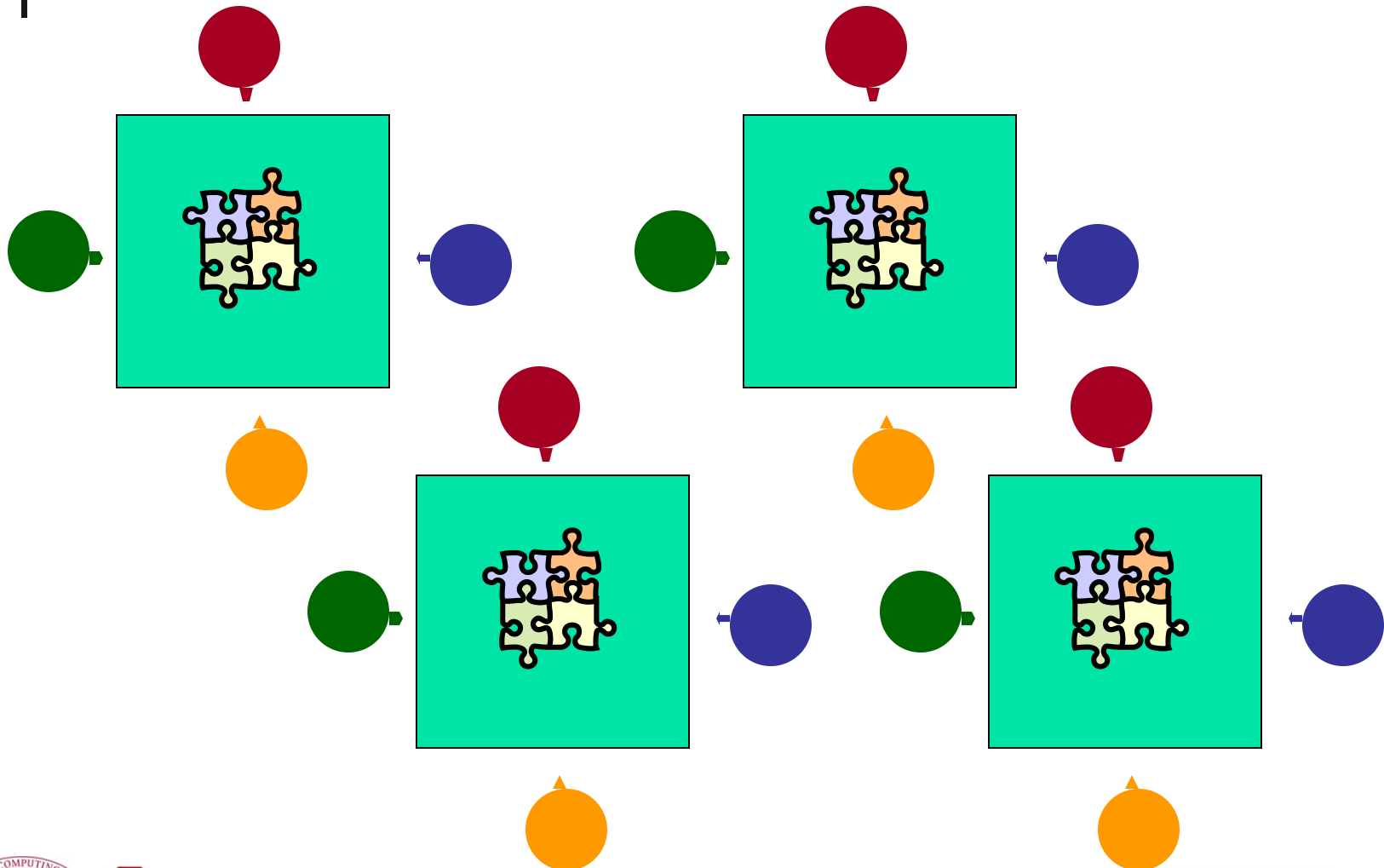
Serial Computing
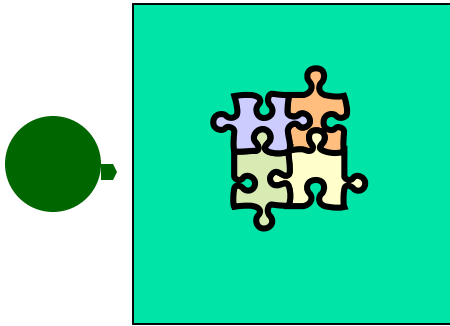
Suppose you want to do a jigsaw puzzle that has, say, a thousand pieces.

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?
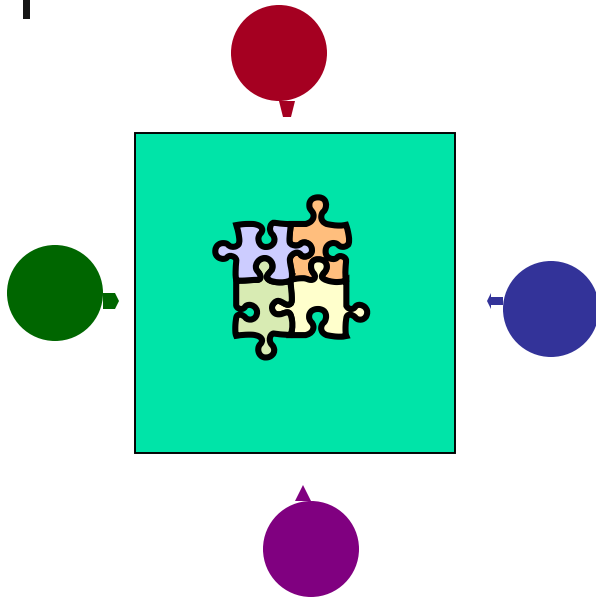
Now let's put Paul and Charlie on the other two sides of the table. Each of you can work on a part of the puzzle, but there'll be a lot more contention for the shared resource (the pile of puzzle pieces) and a lot more communication at the interfaces. So y'all will get noticeably less than a 4-to-1 speedup, 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
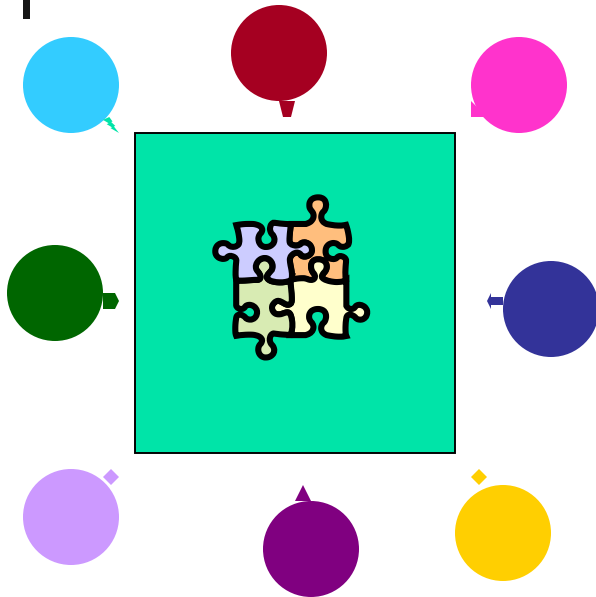
If we now put Dave and Tom and Horst and Brandon on the corners of the table, there's going to be a whole lot of contention for the shared resource, and a lot of communication at the many interfaces. So the speedup y'all get will be much less than we'd like; you'll be lucky to get 5-to-1.

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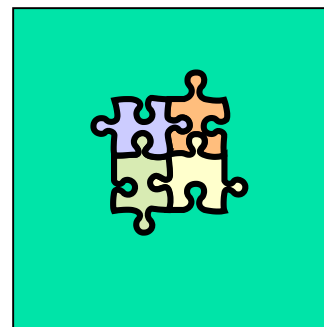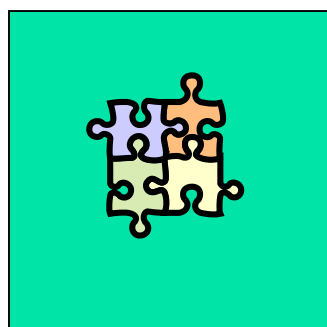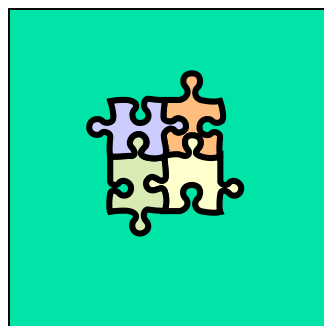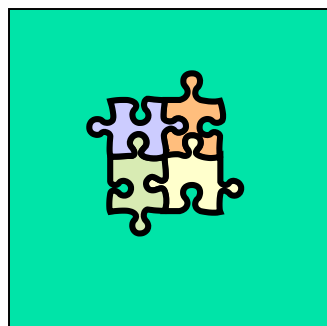
Distributed Parallelism

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.
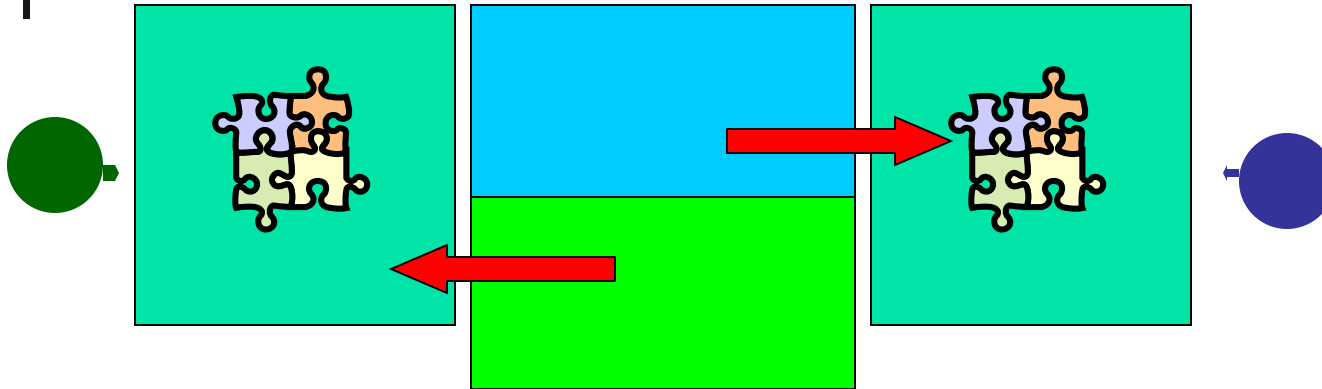
More Distributed Processors

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.
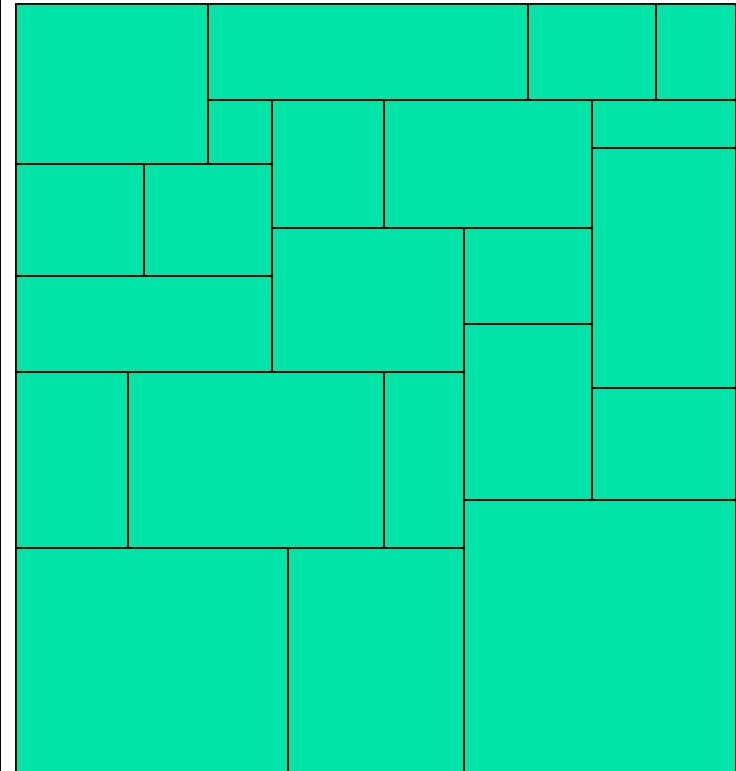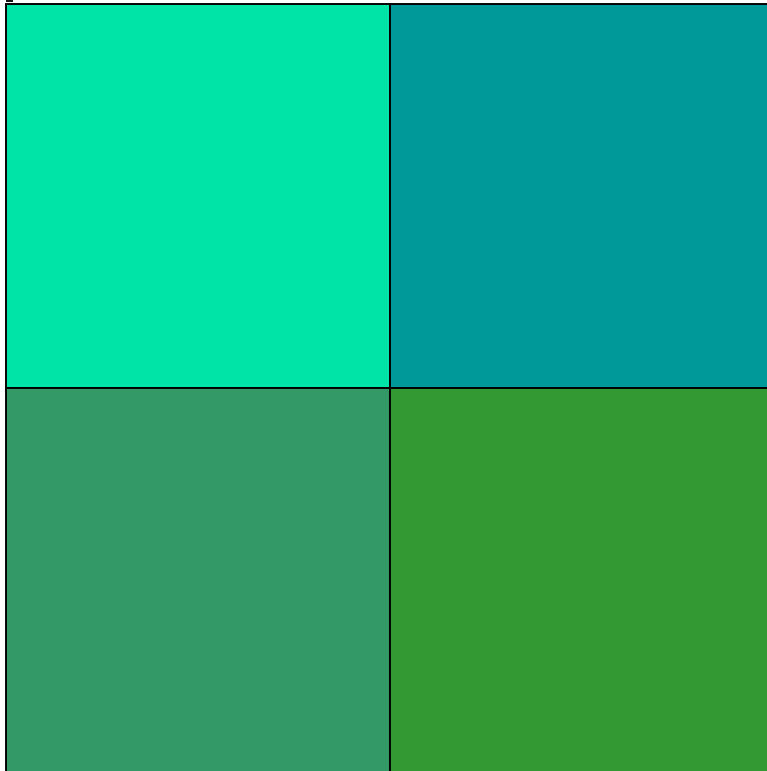
Load Balancing

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.
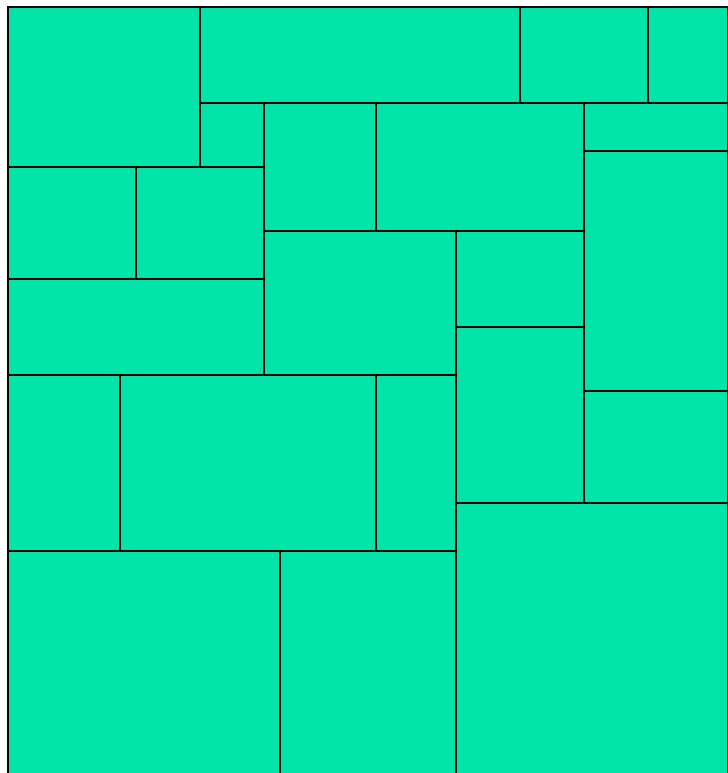
Load Balancing

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.
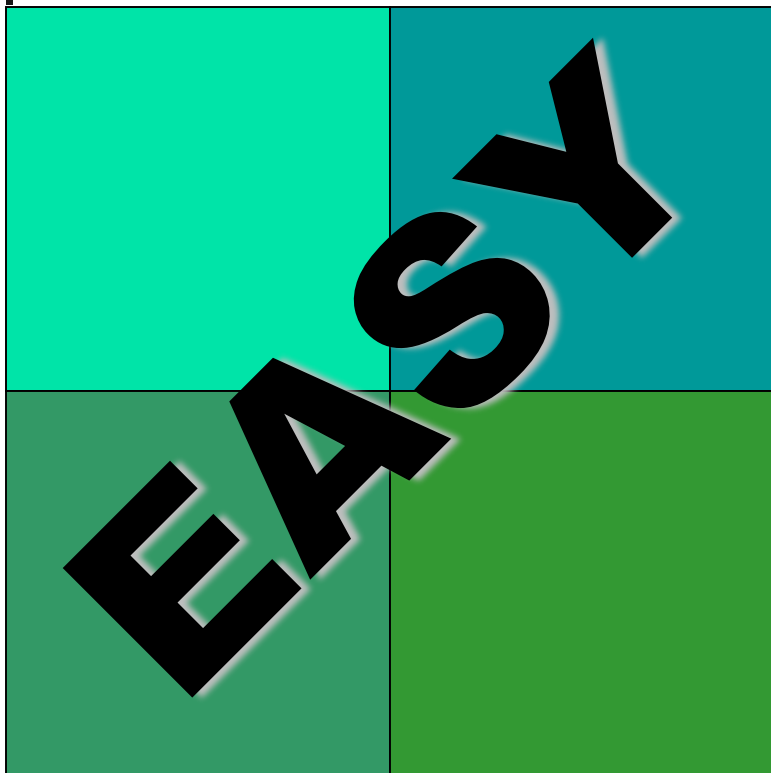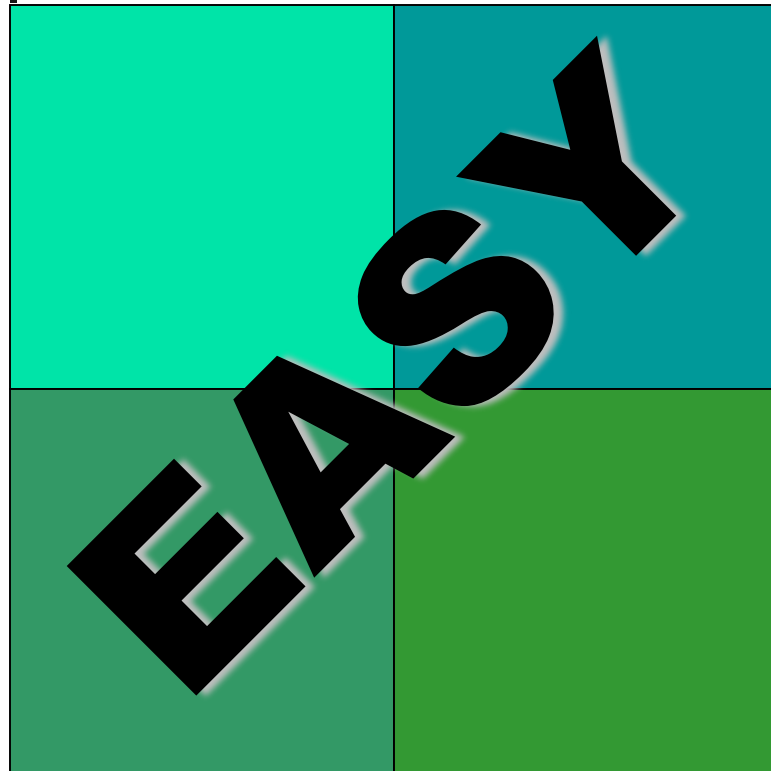
Load Balancing

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 2 years.

It turns out that computer speed, and storage capacity, 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."_**

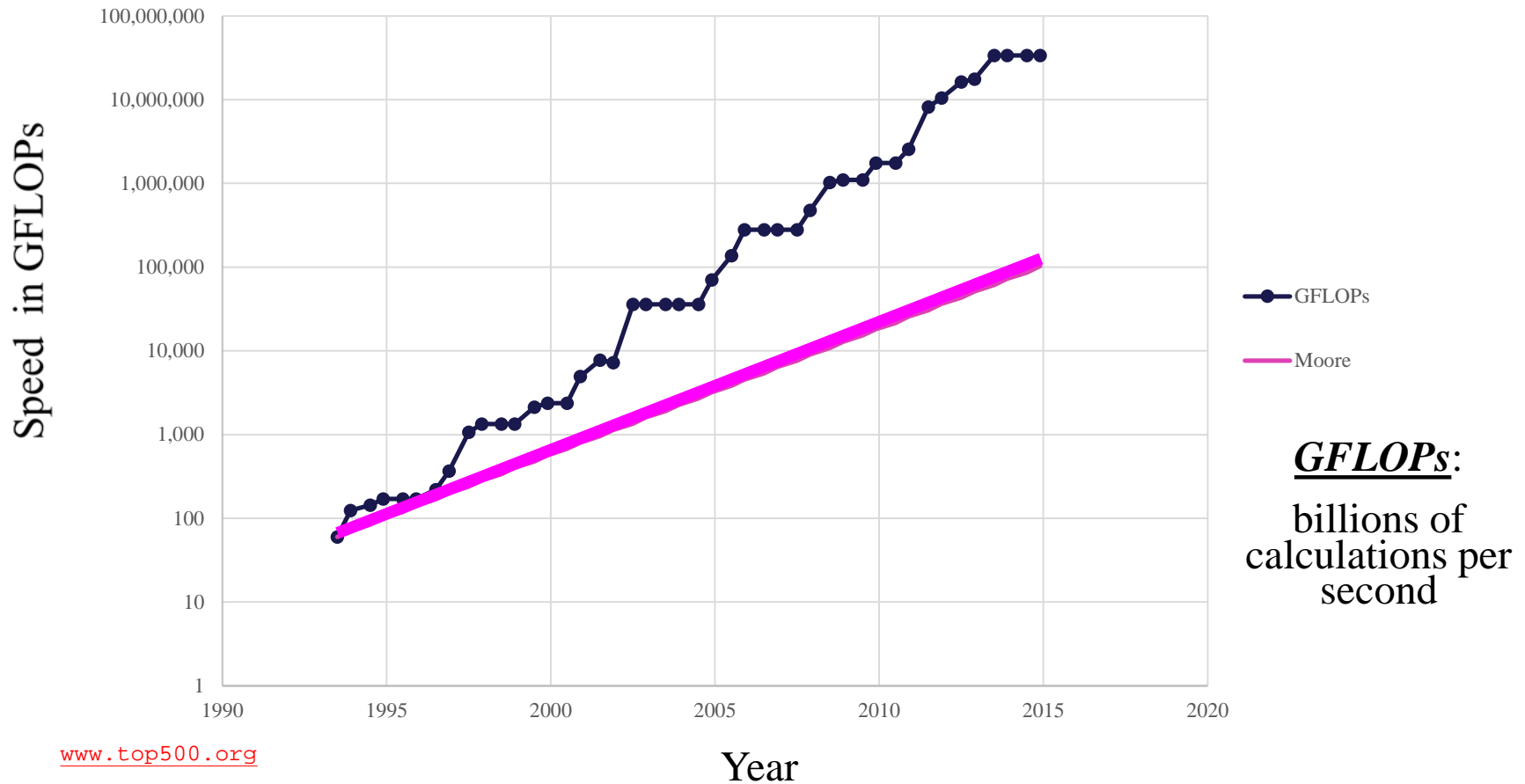(Originally, he predicted a doubling every year, but not long after, he revised that to every other year.)

G. Moore, 1965: "Cramming more components onto integrated circuits." *Electronics*, 38 (8), 114-117.

Fastest Supercomputer vs. Moore

Fastest Supercomputer in the World vs Moore
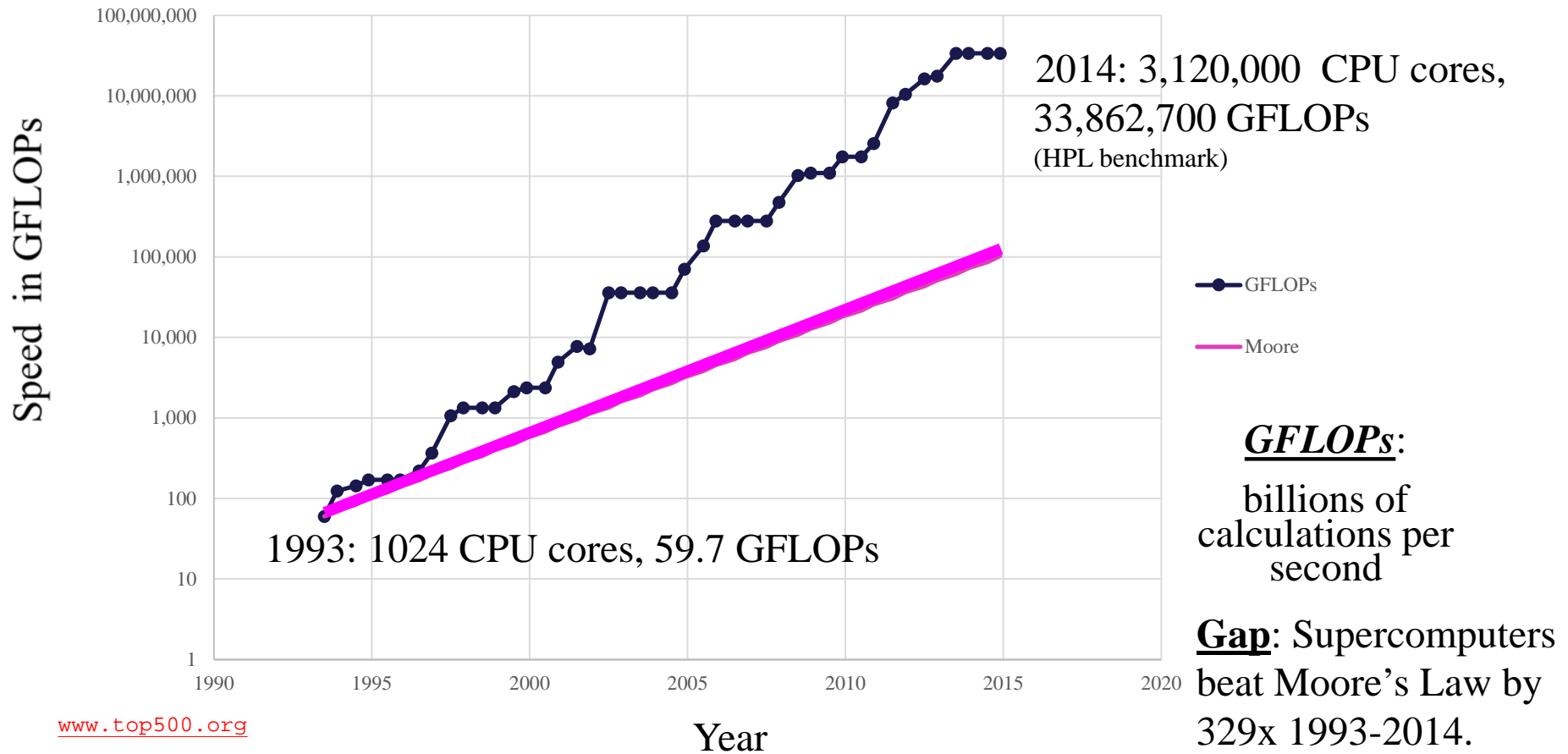

GFLOPs:

billions of calculations per second

Supercomputing in Plain English: Overview
Tue Jan 20 2015

Fastest Supercomputer vs. Moore

Fastest Supercomputer in the World vs Moore

2014: 3,120,000 CPU cores, 33,862,700 GFLOPs
(HPL benchmark)

1993: 1024 CPU cores, 59.7 GFLOPs

GFLOPs: billions of calculations per second

Gap: Supercomputers beat Moore's Law by 329x 1993-2014.

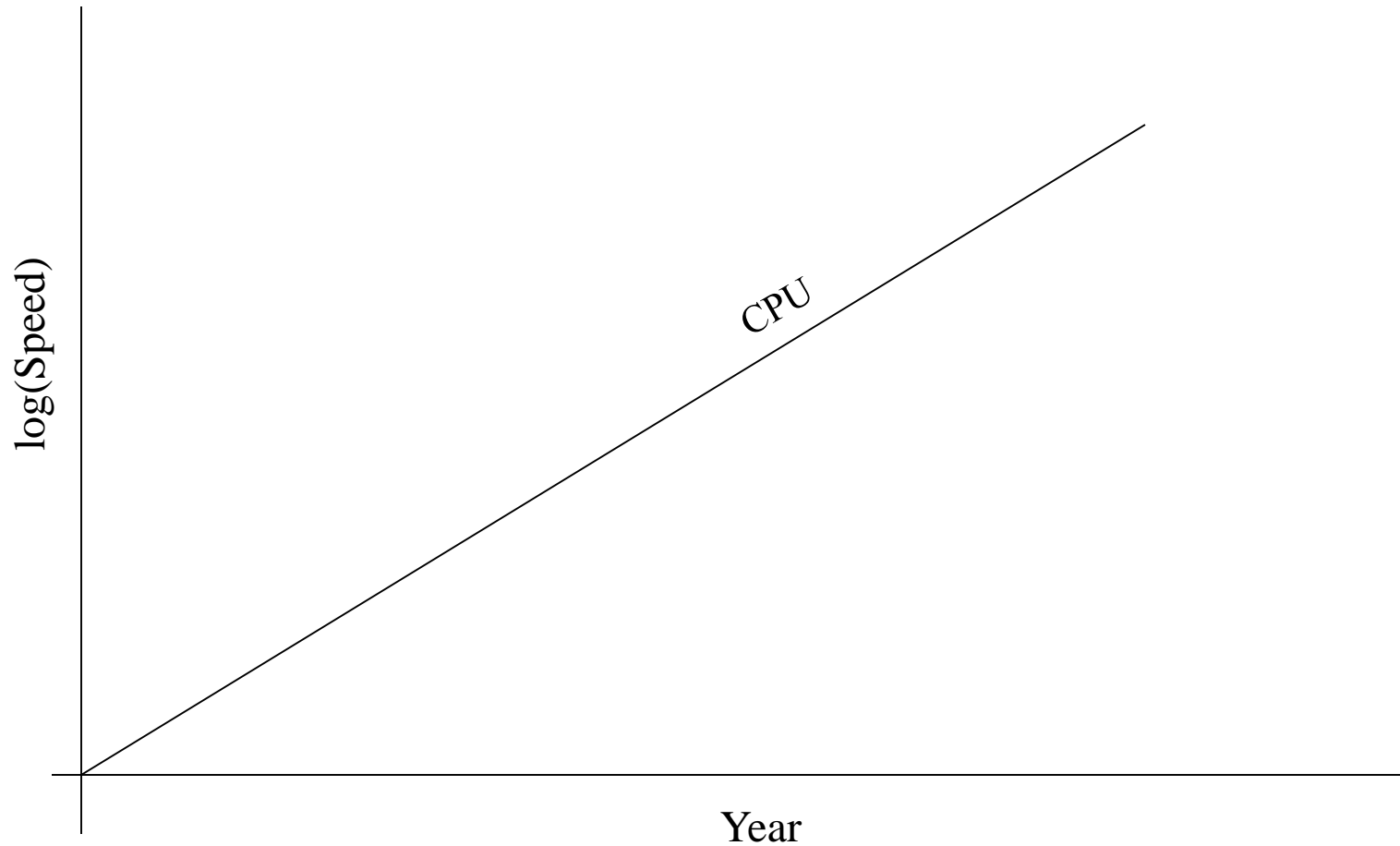Supercomputing in Plain English: Overview
Tue Jan 20 2015

Moore: Uncanny!

- Nov 1971: Intel 4004 – 2300 transistors
- March 2010: Intel Nehalem Beckton – 2.3 billion transistors
- Factor of 1,000,000 improvement in 38 1/3 years
- $2^{(38.33 \text{ years} / 1.9232455)} = 1,000,000$

So, transistor density has doubled every 23 months:
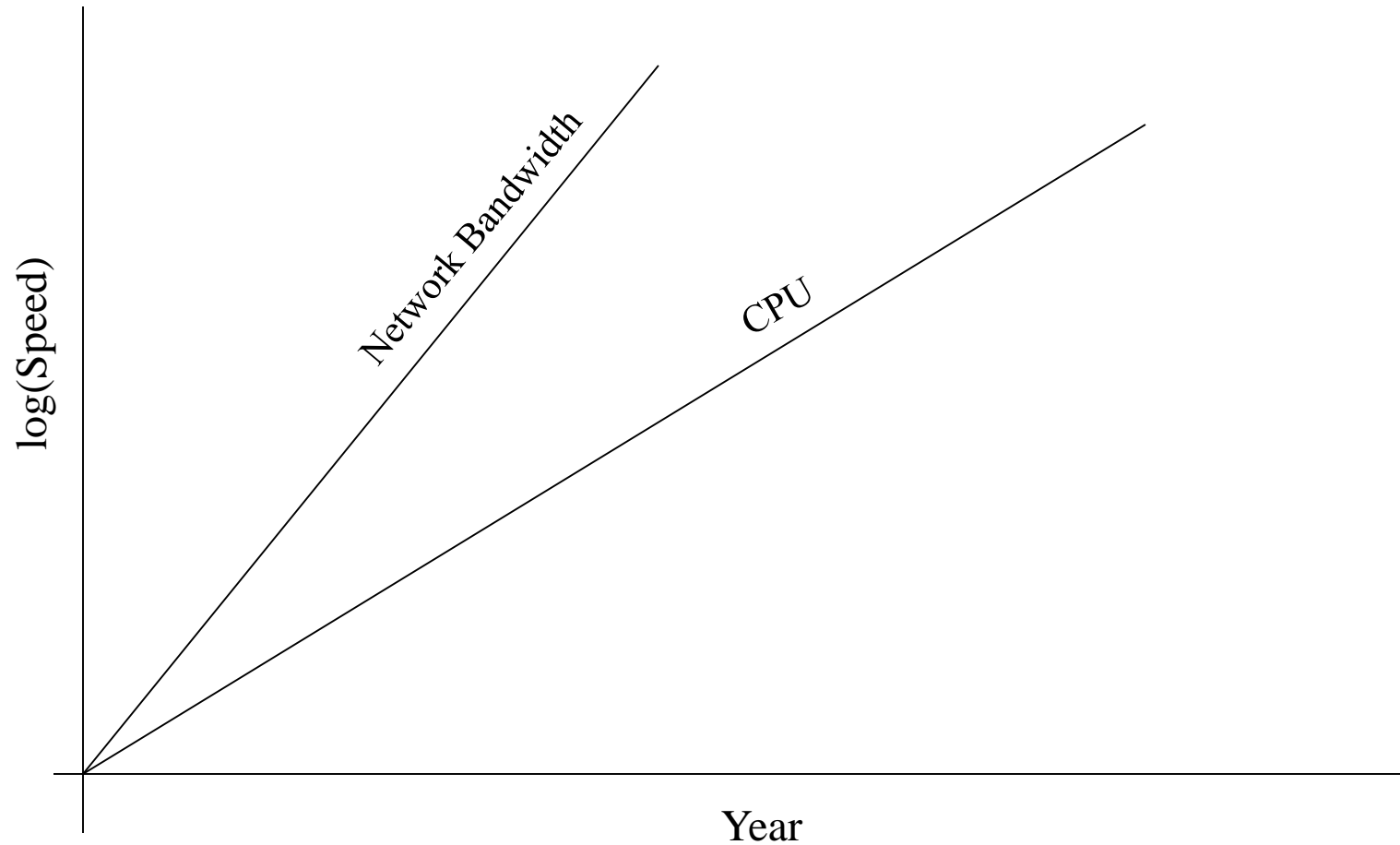
UNCANNILY ACCURATE PREDICTION!

Moore's Law in Practice

Moore's Law in Practice

A log-scale graph with "log(Speed)" on the vertical axis and "Year" on the horizontal axis. Two lines originate from the origin: a steeper line labeled "Network Bandwidth" and a less steep line labeled "CPU".
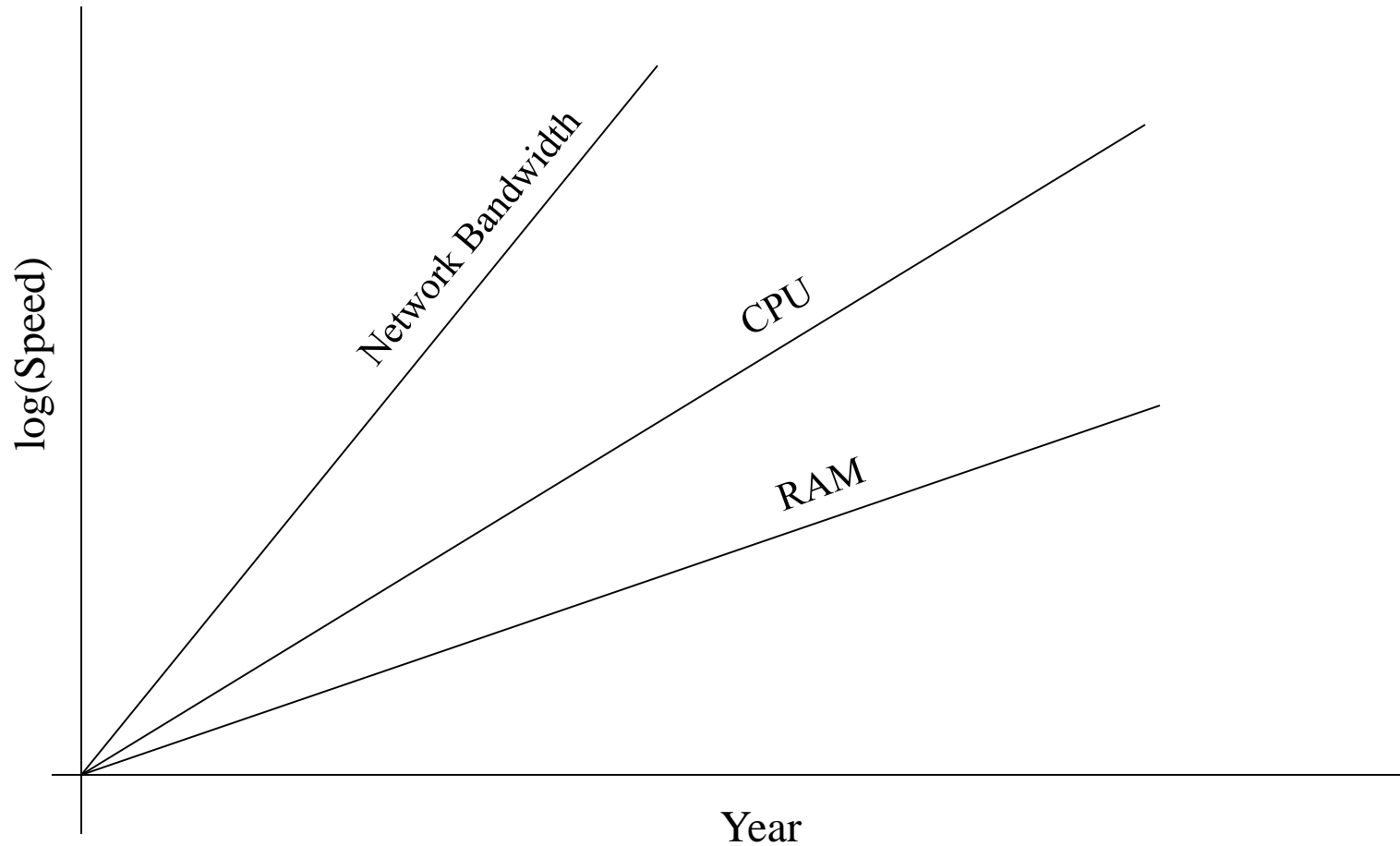
Moore's Law in Practice

Moore's Law in Practice
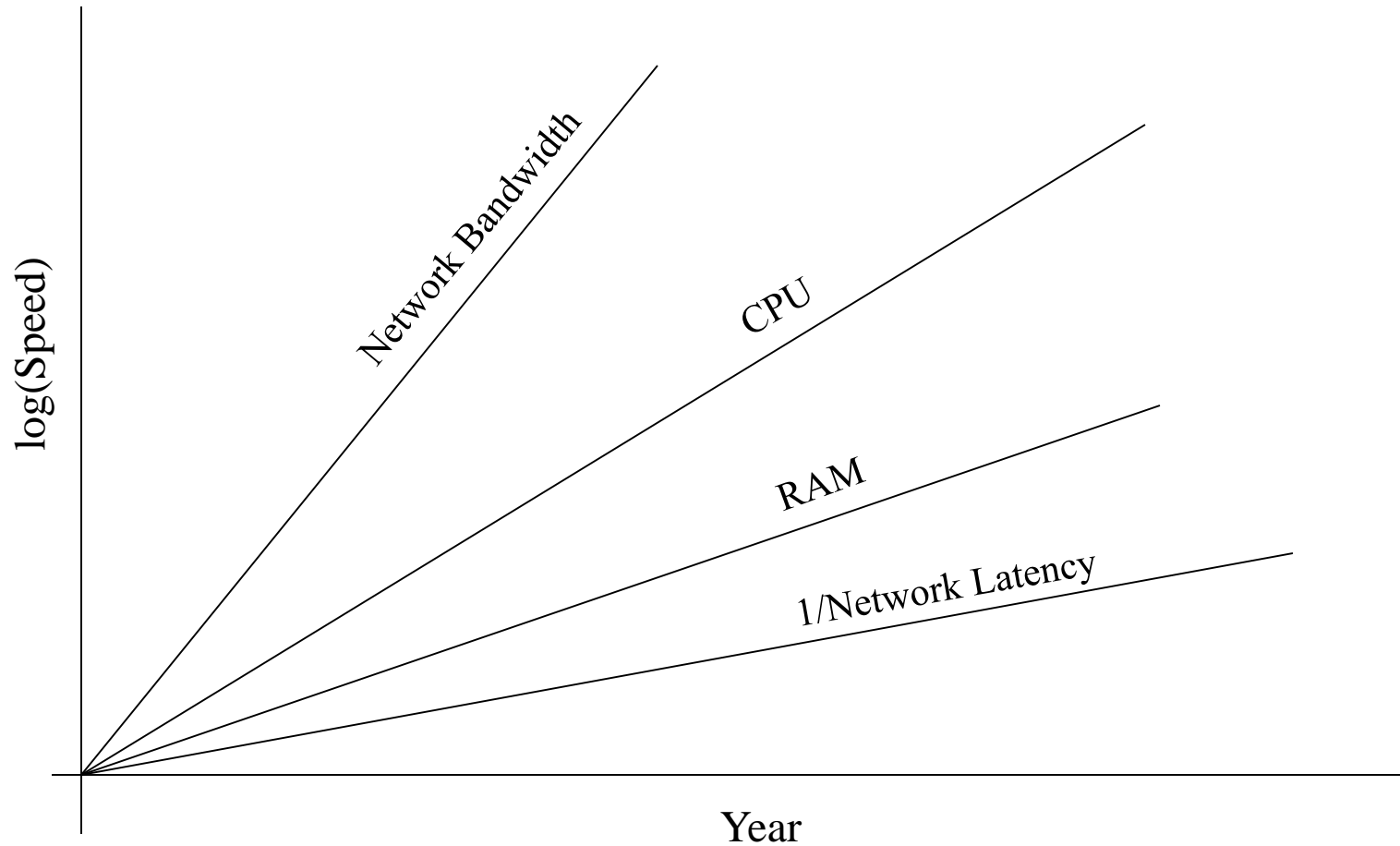
Moore's Law in Practice

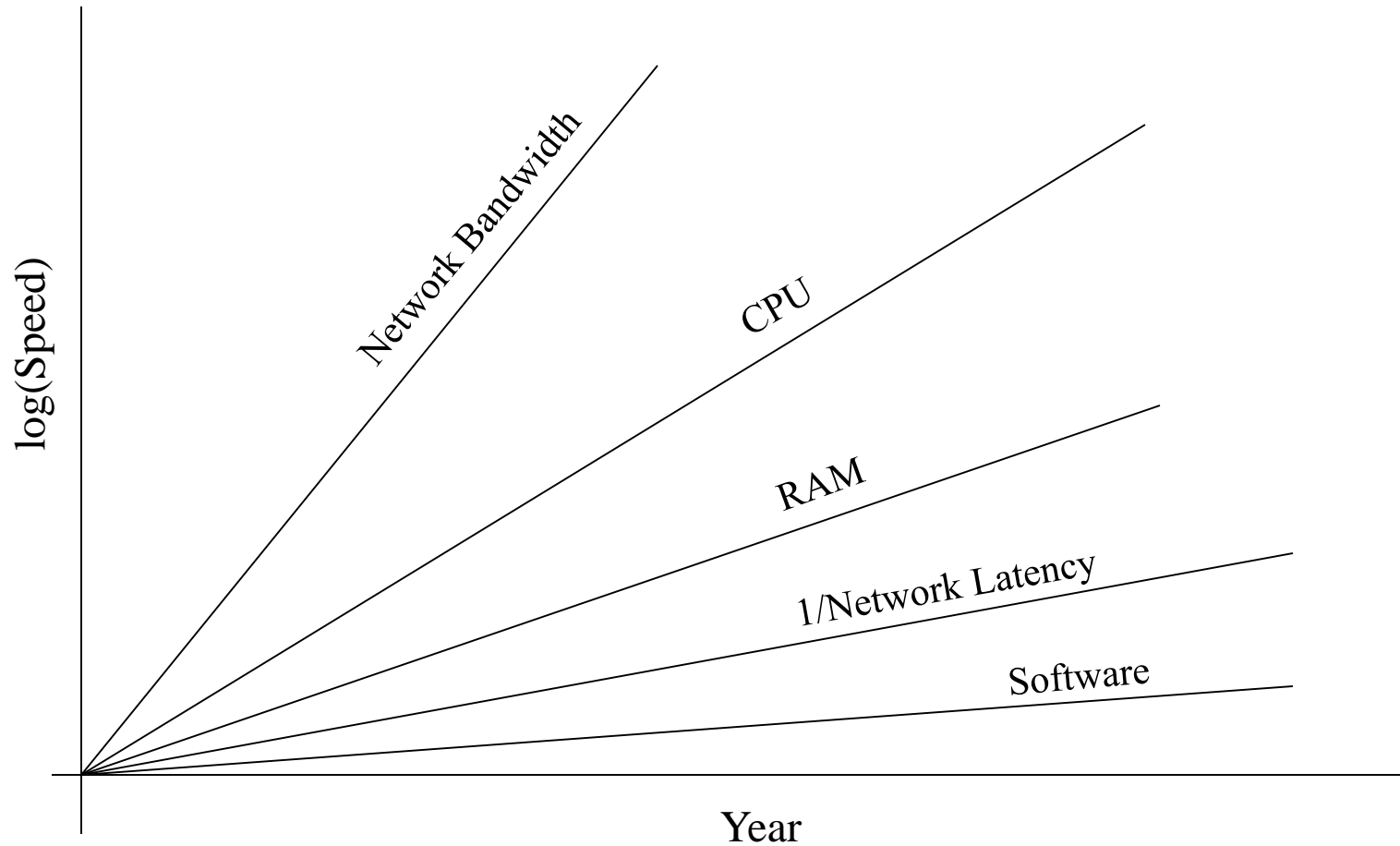# Moore's Law on Gene Sequencers

Increases 10x every 16 months, compared to 2x every 23 months for CPUs.
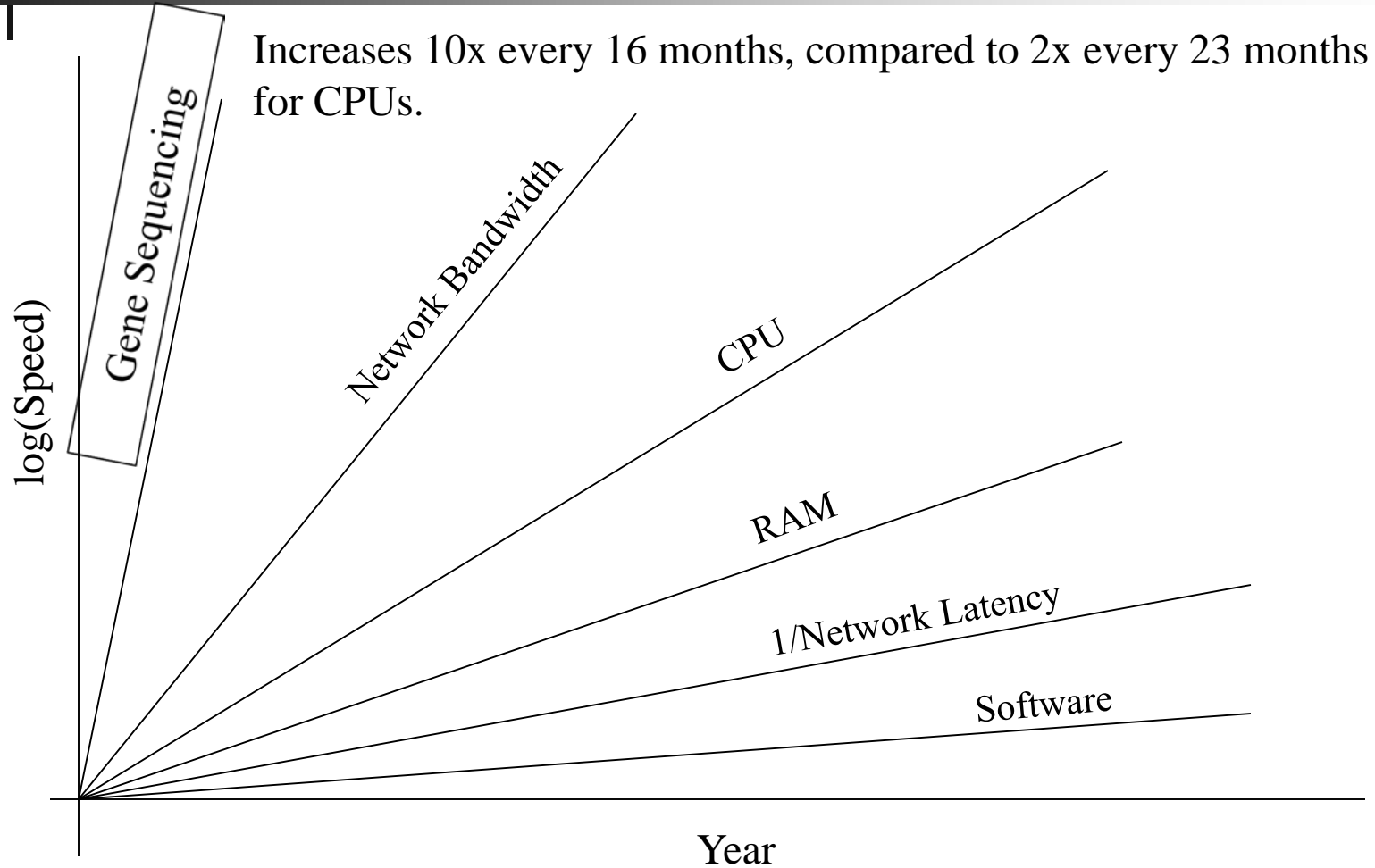
What does 1 TFLOPs Look Like?

2002: Row

2012: Card

1997: Room

ASCI RED[13]
Sandia National Lab

Chip?
Maybe 2016/17

`boomer.oscer.ou.edu`
In service 2002-5: 11 racks

AMD FirePro W9000[14]

NVIDIA Kepler K20[15]

Intel MIC Xeon PHI[16]

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 and on your cell phone in 25 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.

TENTATIVE Schedule

Tue Jan 20: Overview: What the Heck is Supercomputing?

Tue Jan 27: The Tyranny of the Storage Hierarchy

Tue Feb 3: Instruction Level Parallelism

Tue Feb 10: Stupid Compiler Tricks

Tue Feb 17: Shared Memory Multithreading

Tue Feb 24: Distributed Multiprocessing

Tue March 3: Applications and Types of Parallelism

Tue March 10: Multicore Madness

Tue March 17: **NO SESSION** (OU's Spring Break)

Tue March 24: **NO SESSION** (Henry has a huge grant proposal due)

Tue March 31: High Throughput Computing

Tue Apr 7: GPGPU: Number Crunching in Your Graphics Card

Tue Apr 14: Grab Bag: Scientific Libraries, I/O Libraries, Visualization

Thanks for helping!

- OU IT
  - OSCER operations staff (Brandon George, Dave Akin, Brett Zimmerman, Josh Alexander, Patrick Calhoun)
  - Horst Severini, OSCER Associate Director for Remote & Heterogeneous Computing
  - Debi Gentis, OSCER Coordinator
  - Jim Summers
  - The OU IT network team
- James Deaton, Skyler Donahue, Jeremy Wright and Steven Haldeman, OneNet
- Kay Avila, U Iowa
- Stephen Harrell, Purdue U

Coming in 2015!

Red Hat Tech Day, Thu Jan 22 2015 @ OU

http://goo.gl/forms/jORZCz9xh7

Linux Clusters Institute workshop May 18-22 2015 @ OU

http://www.linuxclustersinstitute.org/workshops/

Great Plains Network Annual Meeting, May 27-29, Kansas City

Advanced Cyberinfrastructure Research & Education Facilitators (ACI-REF) Virtual Residency May 31 - June 6 2015

XSEDE2015, July 26-30, St. Louis MO

https://conferences.xsede.org/xsede15

IEEE Cluster 2015, Sep 23-27, Chicago IL

http://www.mcs.anl.gov/ieeecluster2015/

OKLAHOMA SUPERCOMPUTING SYMPOSIUM 2015, **Sep 22-23 2015** @ OU

SC13, Nov 15-20 2015, Austin TX

http://sc15.supercomputing.org/

OK Supercomputing Symposium 2015

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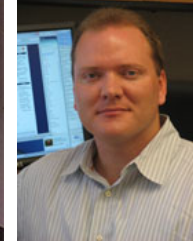Science & Engineering
Assistant Director

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
Cyberinfrastructure

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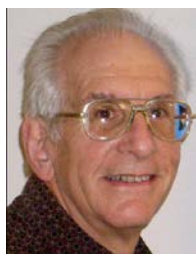Director
Texas Advanced
Computing Center
U. Texas Austin

2008 Keynote:
José Munoz
Deputy Office
Director/Senior
Scientific Advisor
NSF Office of
Cyberinfrastructure

2009 Keynote:
Douglass Post
Chief Scientist
US Dept of Defense
HPC Modernization
Program

2010 Keynote:
Horst Simon
Deputy Director
Lawrence Berkeley
National Laboratory

2011 Keynote:
Barry Schneider
Program Manager
National Science
Foundation

2012 Keynote:
Thom Dunning
Director
National Center for
Supercomputing
Applications

2013 Keynote:
John Shalf
Dept Head CS
Lawrence
Berkeley Lab
CTO, NERSC

2014 Keynote:
Irene Qualters
Division Director
Advanced
Cyberinfarstructure
Division, NSF

**FREE!
Wed Sep 23 2015
@ OU**

**Reception/Poster Session
Tue Sep 22 2015 @ OU
Symposium
Wed Sep 23 2015 @ OU**

Supercomputing in Plain English: Overview
Tue Jan 20 2015

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.samsungssd.com/meetssd/techspecs`

[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/`