Introduction to Scientific and Technical computing

SSC 335/394, 2011

Victor Eijkhout

Who, what, why, how

- Instructors from Texas Advanced Computing Center
- Scientific computing
- About this course

TACC Mission

To enhance research, development, and education and to improve society through the application of advanced computing technologies.

TACC HPC & Storage Resources

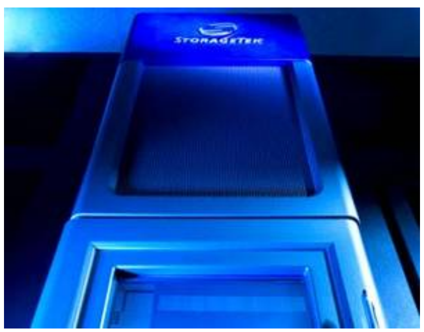
Ranger

Sun quad-socket quad-core
AMD Cluster
3900 Nodes, ~570 TFlops
1 PB memory
Infiniband Interconnect
1.7 PB Lustre File System

LONESTAR

Dell dual-socket 6-core Intel Cluster
2200 Nodes, ~300 TFlops
44 TB memory
Infiniband Interconnect
>1PB Lustre File System

Ranch



Sun StorageTek, 20 PB
max capacity

Corral

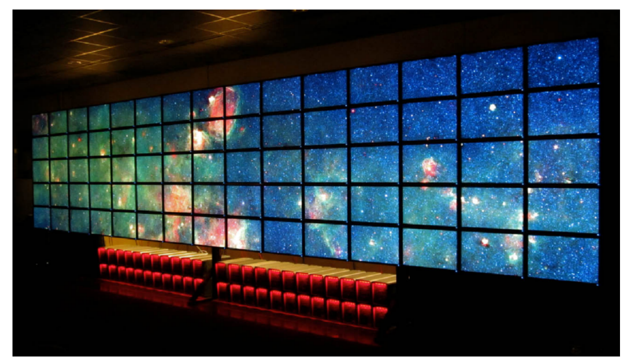
1.2Pbyte disc space
Lustre parallel file
system

Longhorn

256 Dell nodes (2
Intel quad-cores)
with 2 Nvidia FX 5800
GPUs each

Stallion

75 30” monitors,
total 300 Mpixel, still
the largest tiled
display in the world



Mathematics & Science

- In science, we use mathematics to understand physical systems.
- Different fields of science explore different ‘domains’ of the universe, and have their own sets of equations, encapsulated in theories.
- Determining the theories an governing equations requires observation or experimentation, and testing hypotheses.

THE GRAND CHALLENGE EQUATIONS

$$B_i A_i = E_i A_i + \rho_i \sum_j B_j A_j F_{ji}$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}$$

$$\vec{F} = m\,\vec{a} + \frac{dm}{dt}\,\vec{v}$$

$$dU = \left(\frac{\partial U}{\partial S}\right)_V dS + \left(\frac{\partial U}{\partial V}\right)_S dV$$

$$\nabla \cdot \vec{D} = \rho$$

$$Z = \sum_j g_j e^{-E_j/kT}$$

$$\nabla \times \vec{H} = \frac{\partial \vec{D}}{\partial t} + \vec{J}$$

$$F_j = \sum_{k=0}^{N-1} f_k e^{2\pi i jk/N}$$

$$\nabla^2 u = \frac{\partial u}{\partial t}$$

$$P(t) = \frac{\sum_i W_i B_i(t) P_i}{\sum_i W_i B_i(t)}$$

$$\nabla \cdot \vec{B} = 0$$

$$p_{n+1} = r\, p_n (1 - p_n)$$

$$-\frac{h^2}{8\pi^2 m} \nabla^2 \Psi(r,t) + V\,\Psi(r,t) = -\frac{h}{2\pi i} \frac{\partial \Psi(r,t)}{\partial t}$$

$$-\nabla^2 u + \lambda u = f$$

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u} = -\frac{1}{\rho}\nabla p + \gamma \nabla^2 \vec{u} + \frac{1}{\rho}\vec{F}$$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = f$$

• NEWTON'S EQUATIONS • SCHROEDINGER EQUATION (TIME DEPENDENT) • NAVIER-STOKES EQUATION •
• POISSON EQUATION • HEAT EQUATION • HELMHOLTZ EQUATION • DISCRETE FOURIER TRANSFORM •
• MAXWELL'S EQUATIONS • PARTITION FUNCTION • POPULATION DYNAMICS •
• COMBINED 1ST AND 2ND LAWS OF THERMODYNAMICS • RADIOSITY • RATIONAL B-SPLINE •

[Courtesy of San Diego Supercomputer Center]

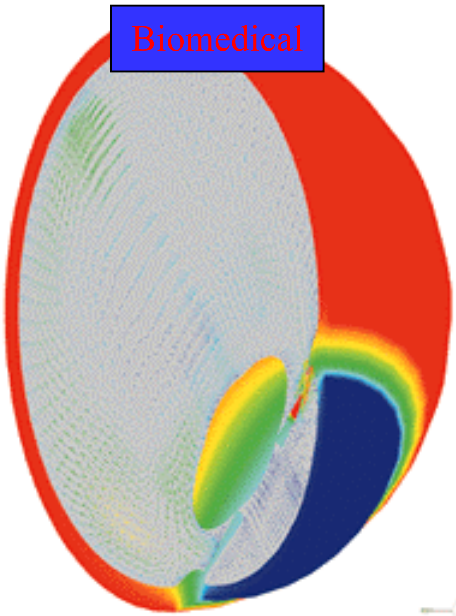
Scientific Computing

- Why should we care about scientific computing?
  - Computational research has emerged to complement experimental methods in basic research, design, optimization, and discovery in all facets of engineering and science
  - In certain cases, computational simulations are the only possible approach to analyze a problem:
    - Experiments may be cost prohibitive (eg. *flight testing a 1,000 fuselage/wing-body configurations for a modern fighter aircraft*)
    - Experiments may be impossible (eg. *interaction effects between the International Space Station and Shuttle during docking*)
  - Simulation capabilities rely heavily on the underlying compute power (eg. amount of memory, total compute processors, and processor performance)
    - Fostered the introduction and development of *super-computers* starting in the 1960's
    - Large-scale compute power is tracked around the world via the *Top500 List* (more on that later)

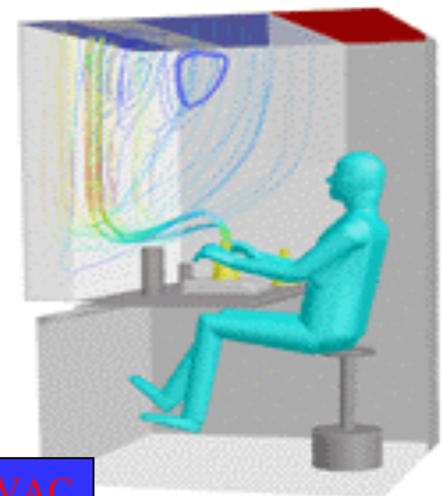
Scientific Computing: a definition

- "The efficient computation of constructive methods in applied mathematics"
  - Applied math: getting results out of application areas
  - Numerical analysis: results need to be correctly and efficiently computable
  - Computing: the algorithms need to be implemented on modern hardware

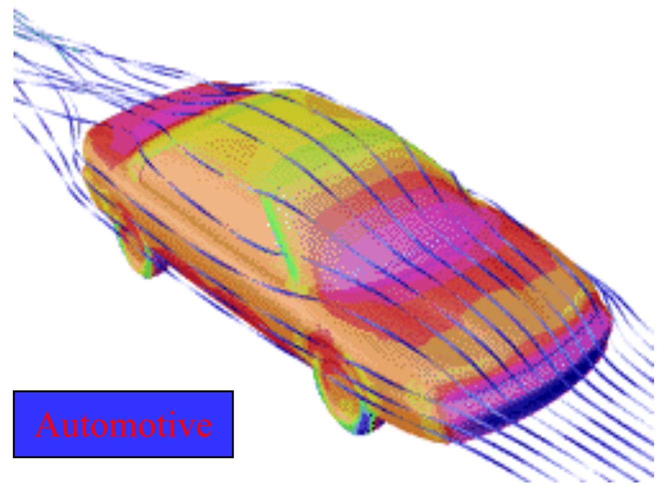
Examples of Scientific Computing
(*it really is everywhere*)



Temperature and natural convection currents in the eye following laser heating.



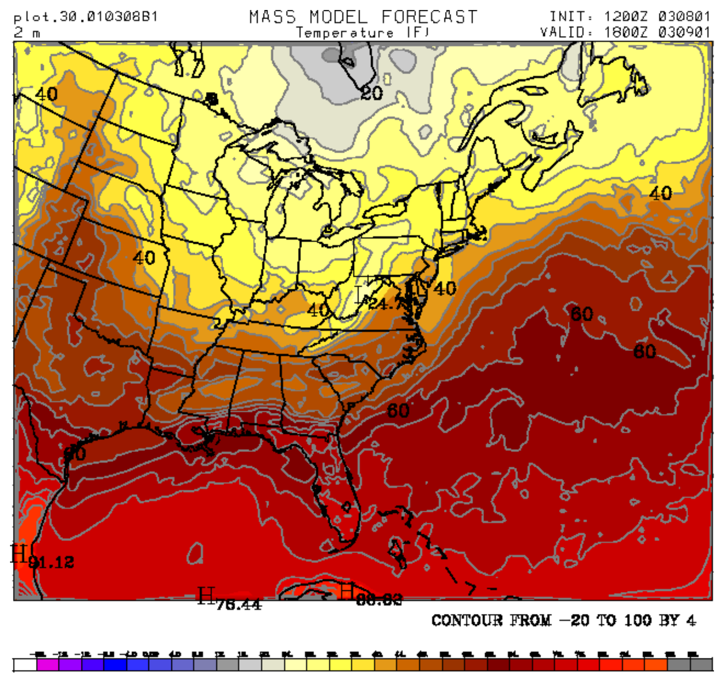
Streamlines for workstation ventilation



Examples of Scientific Computing
(*it really is everywhere*)

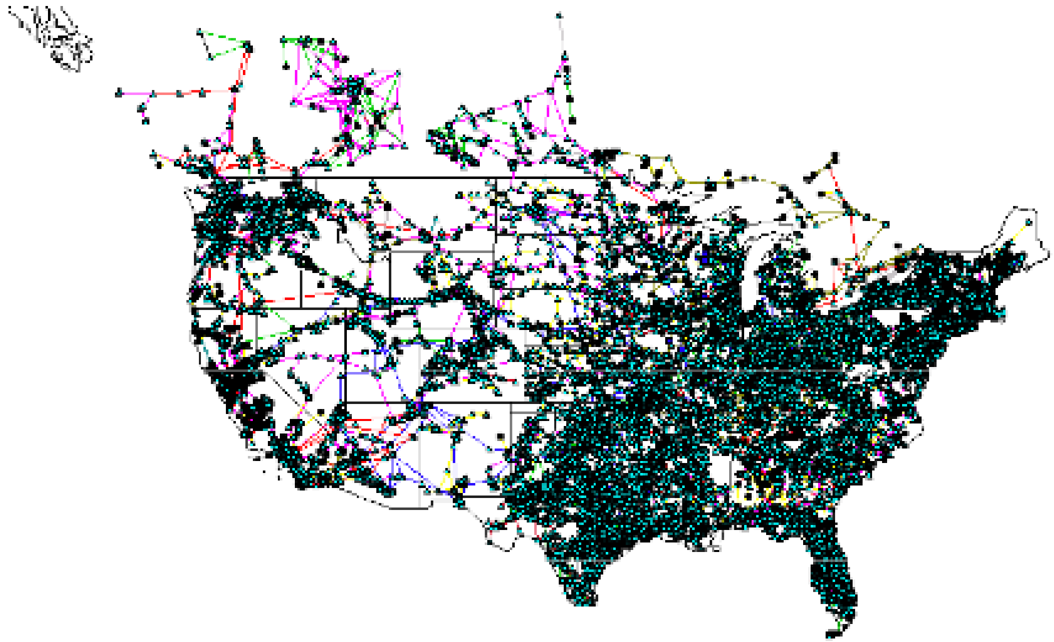
Aerospace

F18 Store Separation



Weather Forecasting

New kinds of computations



The New York Times
Thursday, September 4, 2008

Report on Blackout Is Said To Describe Failure to React

By MATTHEW L. WALD
Published: November 12, 2003

A report on the Aug. 14 blackout identifies specific lapses by various parties, including FirstEnergy's failure to react properly to the loss of a transmission line, people who have seen drafts of it say.

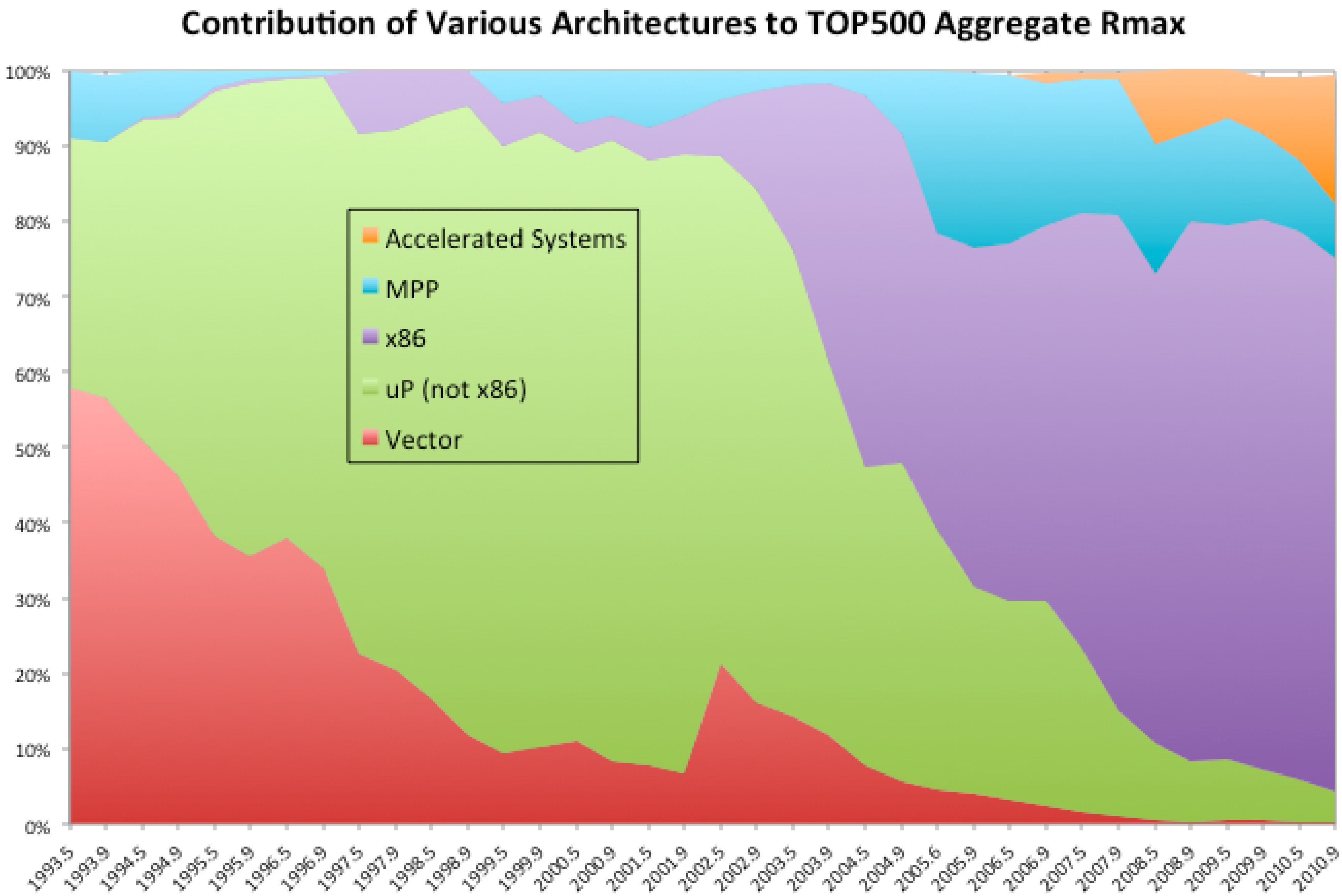
A working group of experts from eight states and Canada will meet in private on Wednesday to evaluate the report, people involved in the

E-MAIL
PRINT
SINGLE-PAGE
REPRINTS
SAVE
SHARE

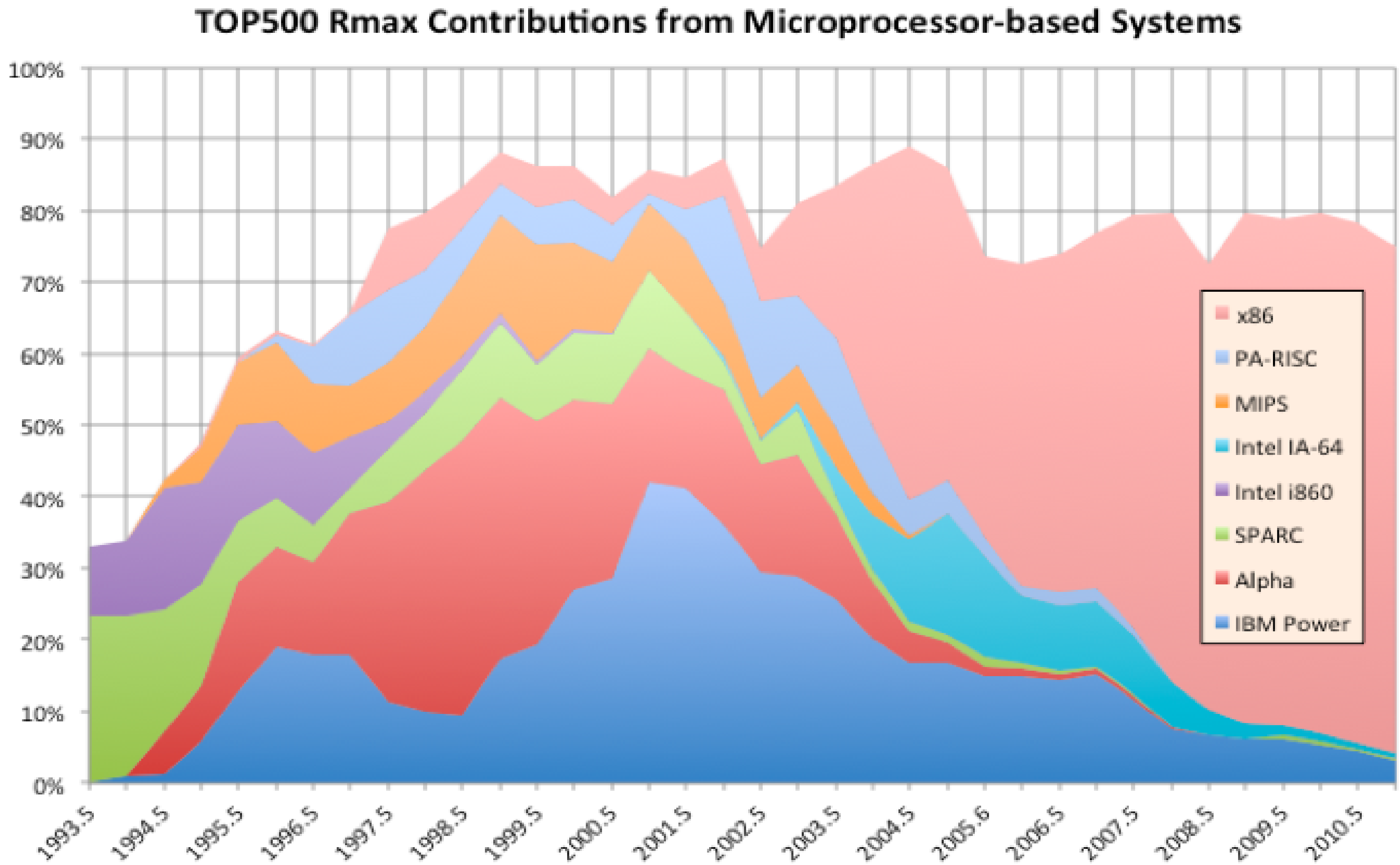
The Top500 List

- http://www.top500.org
- Owner submitted benchmark performance sine 1993
  - based on a dense linear system solve
  - http://www.netlib.org/benchmark/hpl/

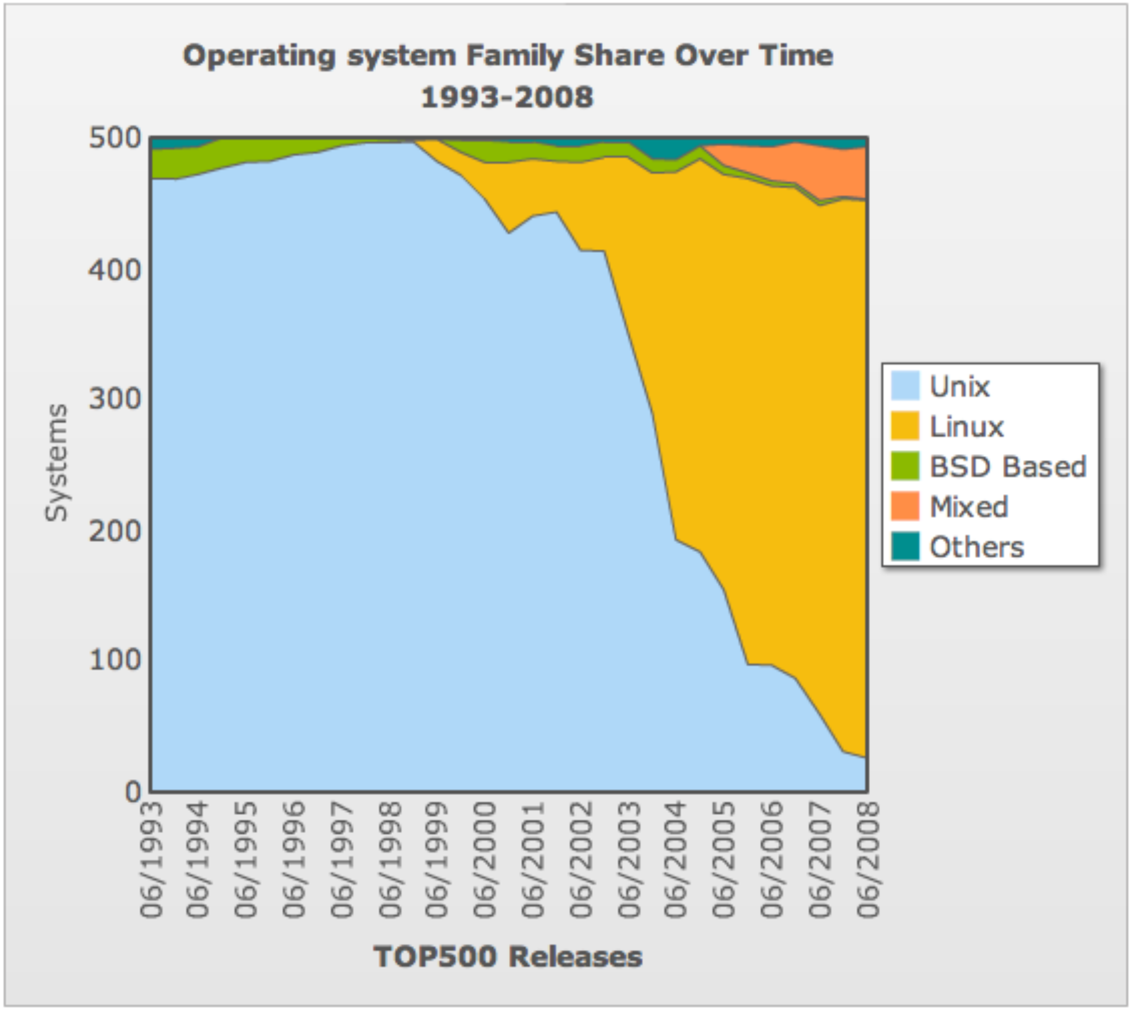
Top500 by Overall Architecture



Top 500 by microprocessor



Top500 by Operating System

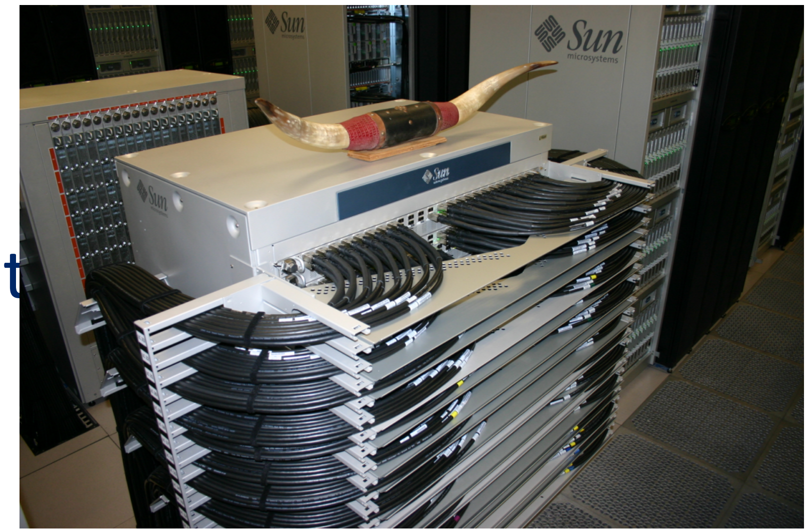


Ranger: What is it?

- Ranger is a unique instrument for computational scientific research housed at UT's PRC
- Results from over 2 ½ years of initial planning and deployment efforts
- Funded by the National Science Foundation as part of a unique program to reinvigorate High Performance Computing in the United States
- Oh yeah, it's a Texas-sized supercomputer

How Much Did it Cost and Who's Involved?

- TACC selected for very first NSF 'Track2' HPC system
  - $30M system acquisition
  - Sun Microsystems is the vendor
  - We competed against almost every open science HPC center



- TACC, ICES, Cornell Theory Center, Arizona State HPCI are teamed to operate/support the system four 4 years ($29M)

Ranger System Summary

- Compute power - 579 Teraflops
  - 3,936 Sun four-socket blades
  - 15,744 AMD “Barcelona” processors
    - Quad-core, four flops/cycle (dual pipelines)
- Memory - 123 Terabytes
  - 2 GB/core, 32 GB/node
  - 132 GB/s aggregate bandwidth
- Disk subsystem - 1.7 Petabytes
  - 72 Sun x4500 “Thumper” I/O servers, 24TB each
  - 40 GB/sec total aggregate I/O bandwidth
  - 1 PB raw capacity in largest filesystem
- Interconnect - 10 Gbps / 2.8 μsec latency
  - Sun InfiniBand-based switches (2), up to 3456 4x ports each
  - Full non-blocking 7-stage Clos fabric
  - Mellanox ConnectX InfiniBand

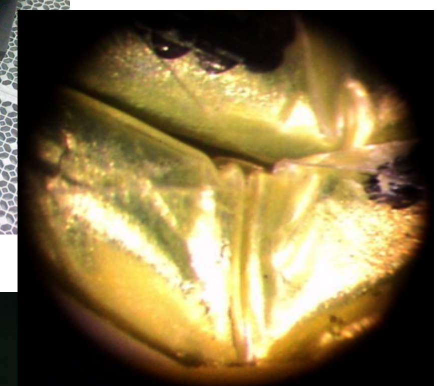
External Power and Cooling Infrastructure

Switches in Place

InfiniBand Cabling in Progress

Ranger Cable Envy?

- On a system like Ranger, even designing the cables is a big challenge (1 cable can transfer data ~3000 times faster then your best ever wireless connection)
- The cables on Ranger were the first demonstration of their kind and are part of a new standard for InfiniBand cabling (1 cable is really 3 cables inside)
- Routing them to all the various components is no fun either



Class Goals/Topics

- Remember that definition "The efficient computation of constructive methods in applied mathematics"
  - Numerical analysis/algorithms, (parallel) computation, and how to combine them
- Theory topics: architecture, numerical analysis, implementing the one on the other
- Practical skills: the tools of scientific computing

Class Goals/Topics

- UNIX Exposure
  - shells/command line
  - environment
  - compilers
  - libraries
- Good practices for scientific software engineering
  - version control
  - build systems
  - Data storage
  - debugging skills

Class Setup

- Theory classes on Tuesday
- Introduction to a practical topic on Thursday, you do a self-guided tutorial at home, recap and discussion the next Thursday.
- There will be mid-term and final exam
- Homework both theory and programming
- Project!

Computer Accounts

- Longhorn
  - we will have class accounts on a powerful academic systems
  - similar user environment on both Lonestar/Ranger
- Your own machine
  - More convenient for development
  - You need a Unix version
  - Linux: check.
  - Mac: make sure X11 is installed
  - Windows: see next slide

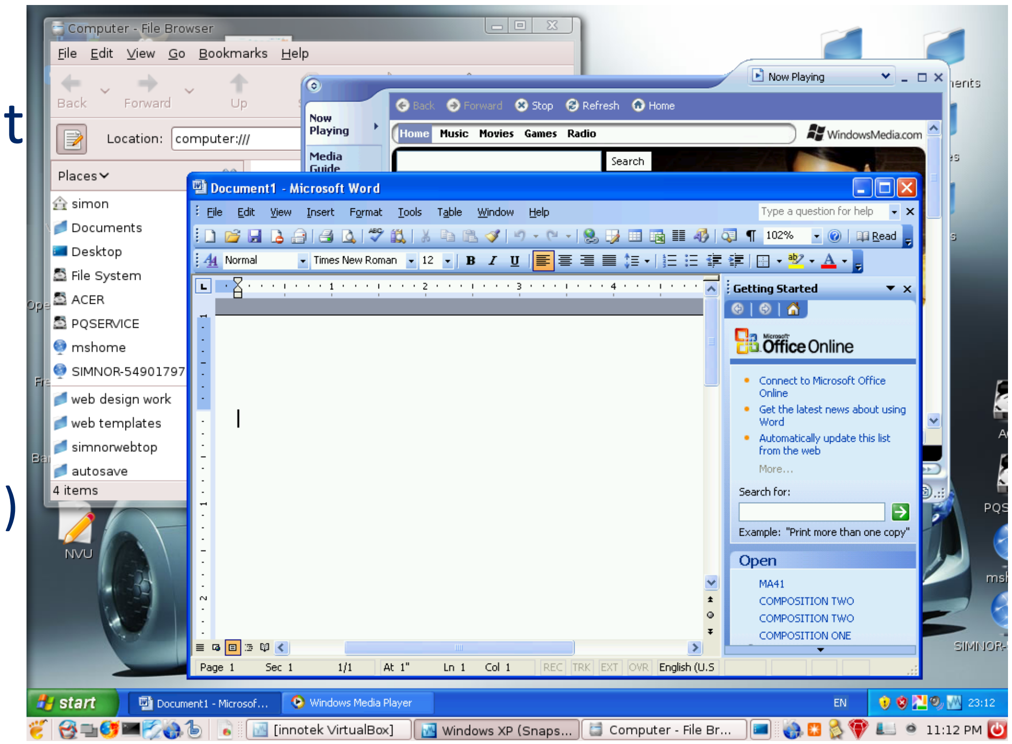
Cygwin – a Unix Environment on Windows

- http://www.cygwin.com/
- What is it?
  - A DLL (cygwin1.dll) which acts as a Linux API emulation layer providing substantial Linux API functionality.
  - A collection of tools, which provide Linux look and feel.
- What isn't it?
  - Cygwin is not a way to run native Linux apps on Windows. You have to rebuild your application from source if you want it to run on Windows.
  - Cygwin is not a way to magically make native Windows apps aware of UNIX functionality. Again, you need to build your apps from source if you want to take advantage of Cygwin functionality.

Other UNIX/Windows Options

- You can use virtual machine software to support a virtualized Unix environment (eg. install Linux within Windows)

- Available with products like:
  - VMWARE (www.vmware.com)
  - VirtualBox (www.virtualbox.org)



“Production”

- Jobs run in a managed environment
  - login to the login node
  - submit jobs to the scheduler
  - wait
  - collect results
- Running programs on the login node highly discouraged
  - avoid resource intensive tasks
  - exceptions include compilers, “standard” UNIX commands (ls, mkdir, cp, mv, etc.)

Remote Login

- Only SSH access is allowed
  - UNIX users: type *ssh* *username@lonestar.tacc.utexas.edu* at the command line
  - *or username@ranger.tacc.utexas.edu*
  - Windows users: Get a client
    - PuTTY (http://www.chiark.greenend.org.uk/~sgtatham/putty/)
  - or use Cygwin and follow the UNIX instructions