CSC 7700: Scientific Computing

Introduction

Dr Gabrielle Allen

Logistics

- Web: https://wiki.cct.lsu.edu/sci-comp/
- Class will be held in 338 Johnston Hall (CCT)
  - One exception
- Lectures:
  - Tuesday 12.10-1.30pm
  - Thursday 12.10-1.30pm
- Schedule of classes
  - https://wiki.cct.lsu.edu/sci-comp/Schedule

Office Hours

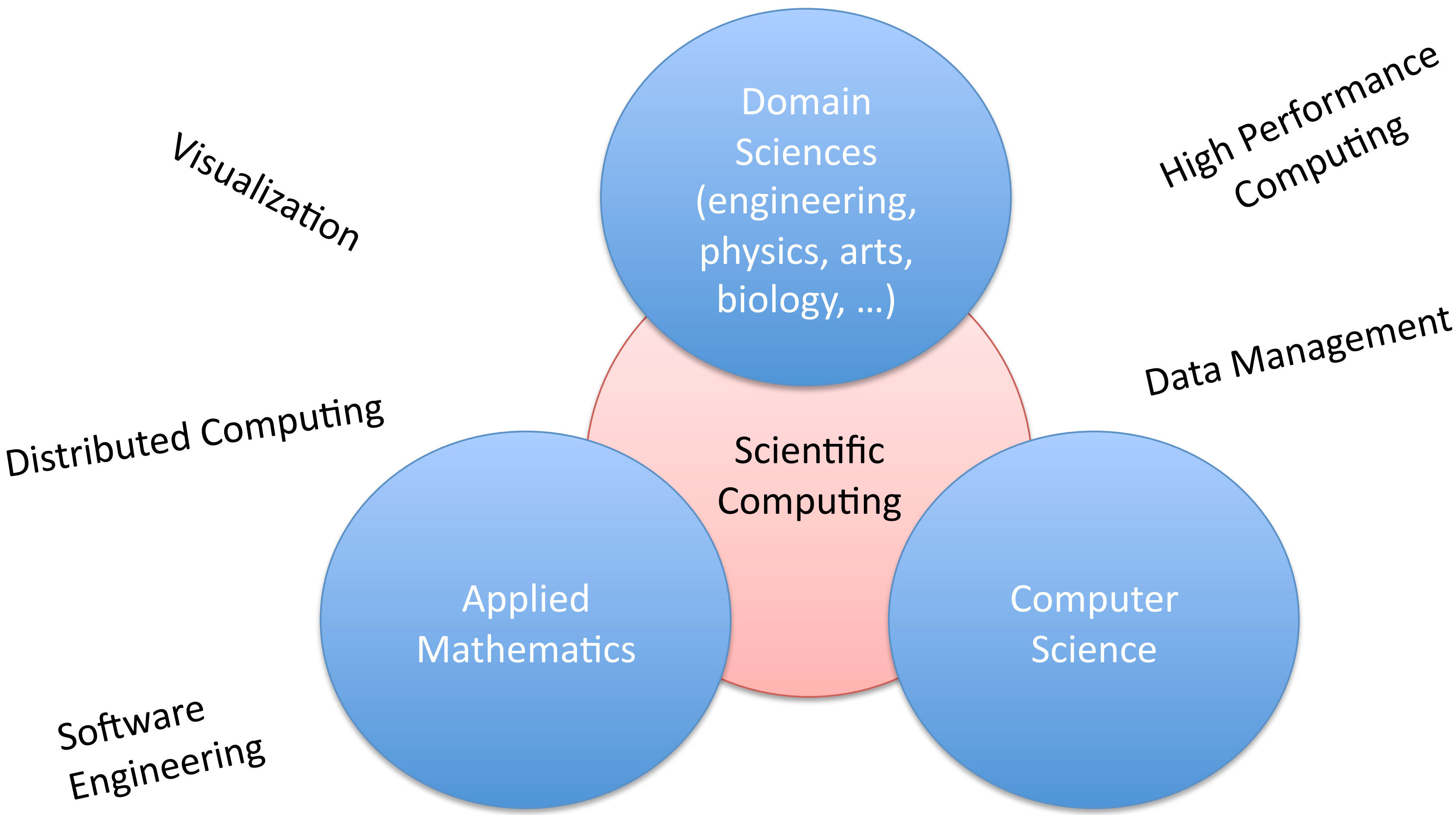
- Dr Allen
  - Wednesday 10am to noon, Johnston Office
  - Also use AIM: gridrebel or email
- Other instructors
  - Posted on web pages

Overview of Course

- Motivation and Scientific Computing
- Methodology and Modules
- Teaching Faculty
- Grading
- Cyberinfrastructure Resources

Motivation and Scientific Computing

Scientific Computing



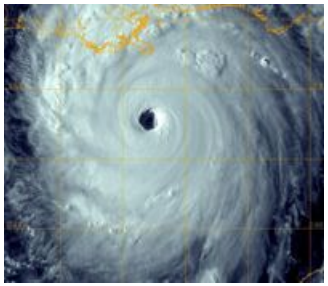
Scientific Computing (from wikipedia)

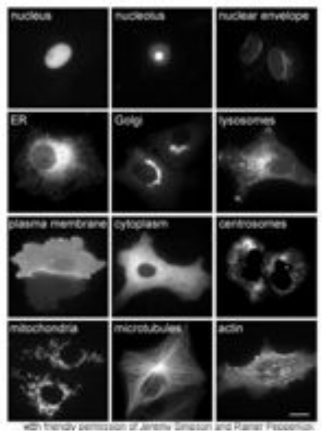
- Computational science (or scientific computing) is the field of study concerned with constructing mathematical models and quantitative analysis techniques and using computers to analyse and solve scientific problems. In practical use, it is typically the application of computer simulation and other forms of computation to problems in various scientific disciplines.
- The field is distinct from computer science (the study of computation, computers and information processing). It is also different from theory and experiment which are the traditional forms of science and engineering. The scientific computing approach is to gain understanding, mainly through the analysis of mathematical models implemented on computers.
- Scientists and engineers develop computer programs, application software, that model systems being studied and run these programs with various sets of input parameters. Typically, these models require massive amounts of calculations (usually floating-point) and are often executed on supercomputers or distributed computing platforms.

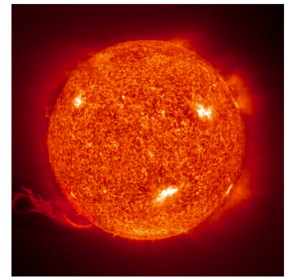
Computational Sciences

Coastal Studies | Biochemistry | Aerodynamics | Astrophysics | AVATAR

Experimental







Storyboard

Theoretical

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} + \vec{\nabla}\cdot(\rho\vec{v}) &= 0,\\
\rho\left(\frac{\partial \vec{v}}{\partial t} + \vec{v}\cdot\vec{\nabla}\vec{v}\right) &= -\vec{\nabla}p - \rho\vec{\nabla}\Phi
\end{aligned}
$$

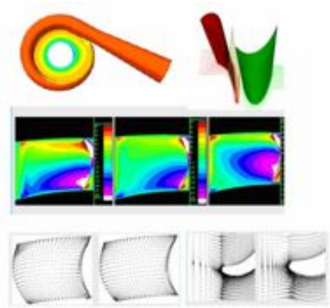
$$
\frac{d\vec{r}_i}{dt} = -\eta \sum_{\{s_i\}} \left\{ -\frac{P_\alpha}{P_0}\frac{\partial}{\partial \vec{r}_i}\left[\frac{e^{-\beta H(s,r)}}{\sum_{\{s_i\}} e^{-\beta H(s,r)}}\right]\right\} = -\eta\beta\left[<\frac{\partial H}{\partial \vec{r}_i}>_\alpha - <\frac{\partial H}{\partial \vec{r}_i}>_0\right]
$$

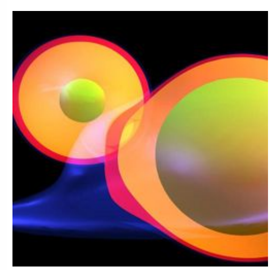
$$
\begin{aligned}
\Delta w &= \frac{1}{4\pi}\frac{\sin(\phi)}{r^2}\,\Gamma_j \cdot ds\\
w_{ij} &= \int \frac{1}{4\pi}\frac{\sin(\phi)}{r^2}\,ds \cdot \Gamma_j = A_{ij}\cdot\Gamma_j\\
A_{ij} &= \frac{1}{4\pi}\Bigg[\frac{1}{(x+h)(y-k)-(x-h)(y+k)}\left\{\frac{2h(x+h)+2k(y+k)}{R_1} - \frac{2h(x-h)+2k(y-k)}{R_2}\right\}\\
&\qquad - \frac{1}{(y+k)}\left\{1+\frac{(x+h)}{R_1}\right\} + \frac{1}{(y-k)}\left\{1+\frac{(x-h)}{R_2}\right\}\Bigg]
\end{aligned}
$$

where $R_1 = \sqrt{(x+h)^2+(y+k)^2}$ and $R_2 = \sqrt{(x-h)^2+(y-k)^2}$

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} + \vec{\nabla}\cdot(\rho\vec{v}) &= 0,\\
\rho\left(\frac{\partial \vec{v}}{\partial t} + \vec{v}\cdot\vec{\nabla}\vec{v}\right) &= -\vec{\nabla}p - \rho\vec{\nabla}\Phi\\
P &= K\rho^{1+\frac{1}{n}}, \text{ (polytropic)}\\
\nabla^2\Phi &= 4\pi G\rho,
\end{aligned}
$$

Computational





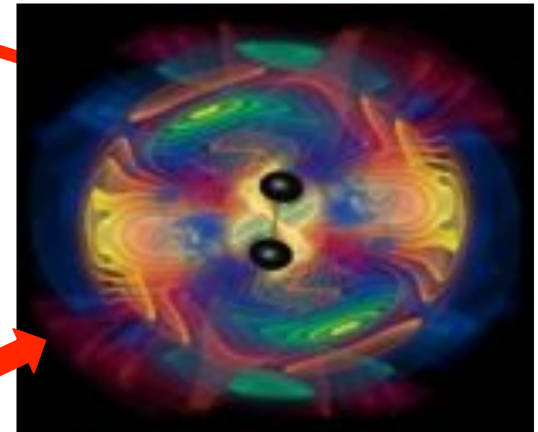
Digital Animation

LSU

Slide: Dr Joel Tohline

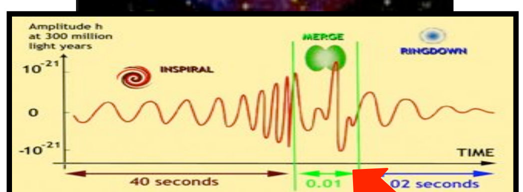
Gravitational Wave Physics

Models & Simulation



Observations

Scientific Discovery!



Theory

$$\partial_t \alpha - \beta^i \partial_i \alpha = -\alpha^2 f(\alpha)\ \mathrm{tr}K$$

$$\tilde{\Gamma}^i = \tilde{\gamma}^{jk} \tilde{\Gamma}^i_{jk}$$

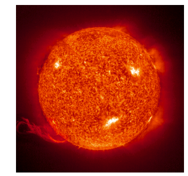
$$G_{\mu\nu} = 8\pi T_{\mu\nu}$$

Computer Science

```
do i=1,17
   xtemp = x(i)
   do j=1,17
      xsum = xsum + A(i,j)*x(j)/xtemp
   enddo
   x(i) = xsum
enddo
```

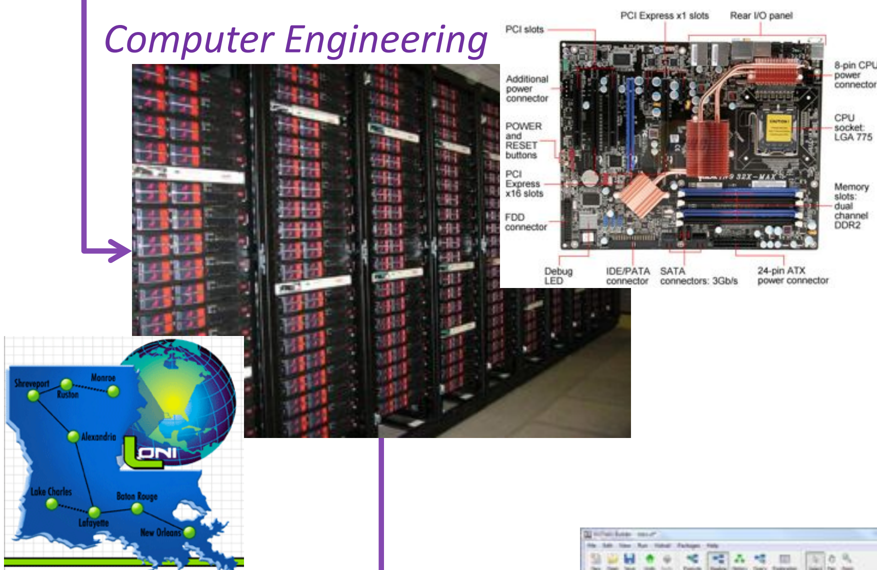
$$
\begin{aligned}
A_1x_1 + A_2x_2 + A_3x_3 + \cdots + A_{16}x_{16} &= A_{17}\\
B_1x_1 + B_2x_2 + B_3x_3 + \cdots + B_{16}x_{16} &= B_{17}\\
&\vdots\\
P_1x_1 + P_2x_2 + P_3x_3 + \cdots + P_{16}x_{16} &= P_{17}
\end{aligned}
$$

Applied Mathematics

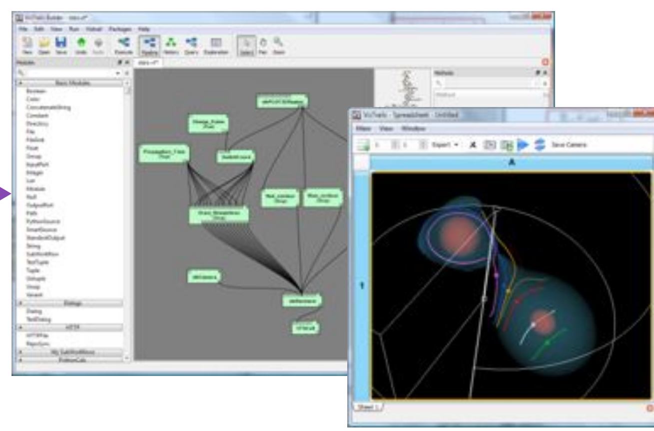


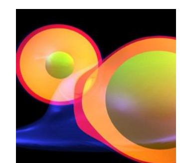
$$
\begin{aligned}
\frac{\partial\rho}{\partial t} + \vec{\nabla}\cdot(\rho\vec{v}) &= 0,\\
\rho\left(\frac{\partial\vec{v}}{\partial t} + \vec{v}\cdot\vec{\nabla}\vec{v}\right) &= -\vec{\nabla}p - \rho\vec{\nabla}\Phi\\
P &= K\rho^{1+\frac{1}{n}},\ \text{(polytropic)}\\
\nabla^2\Phi &= 4\pi G\rho,
\end{aligned}
$$

Computer Engineering

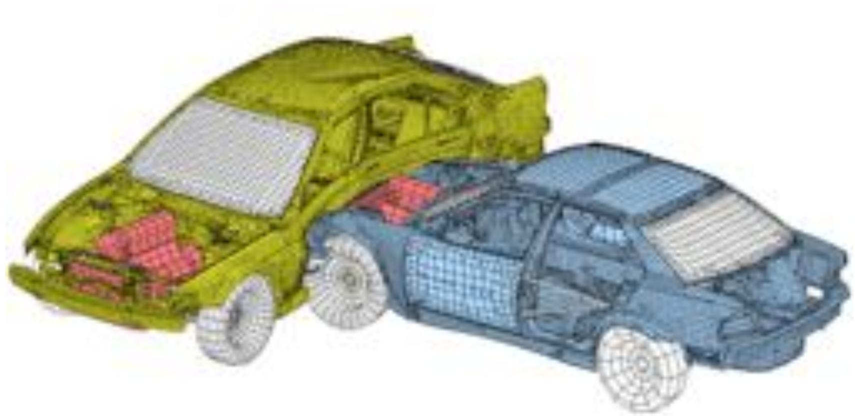


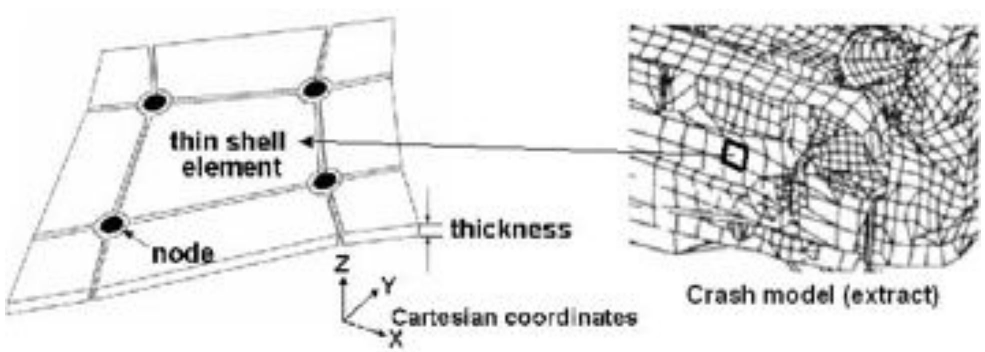
Visualization
And Analysis





Crash Simulation

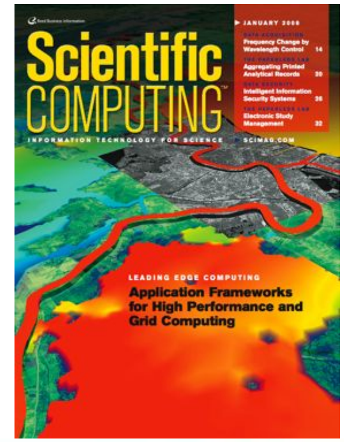


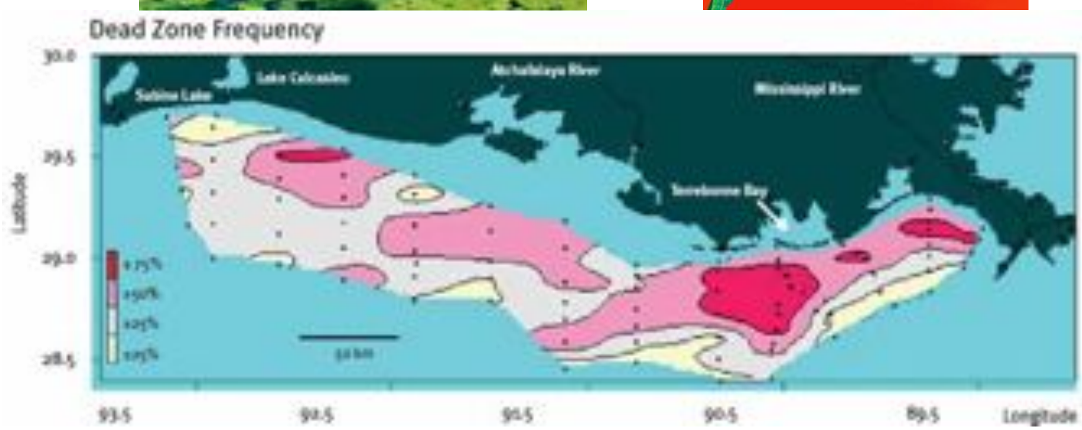


- E.g. BMW, Volkswagen, Ford, Audi, …
- Computer simulation to model safety for car/ occupants
  - Alternative to crashing cars
- Structural Analysis, usually modeled with finite elements
- Kinetic energy --> deformation energy
  - Design crumple zones, etc

Modeling Louisiana Coastal Area

- Rich dynamic environment for modeling: coupled models, multi-scale, real-time data (sensors, satellites)
  - Hurricane forecasts
  - Emergency preparedness
  - Wetland reconstruction
  - Ecological studies and fish populations
  - Oil spill mitigation
  - Levee design
  - Sea rescue
  - Hypoxia “Dead Zone”
  - Algae blooms

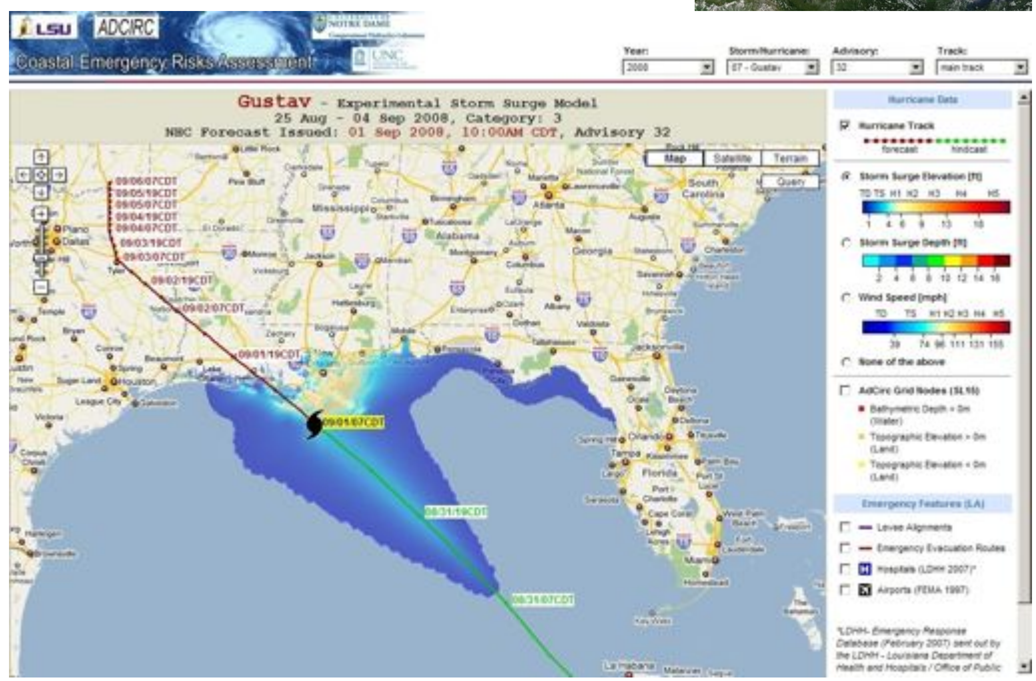




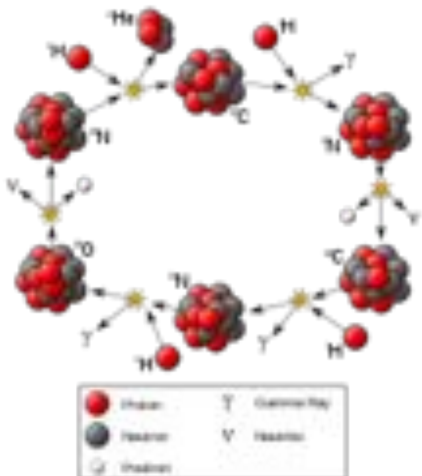
Hurricane Forecasting

- 5 days from landfall advisories from NHC provides best guess of track and intensity
- Surge/wave models prepared
- Wind forcing either from models or other methods
- Run large ensemble of models
- Archive model results
- Calculate products,
  - e.g. MOM, MOF, MEOW
  - compare with observations

Evacuation decisions 24hrs before landfall



Nuclear Fusion



Tokamak reactor

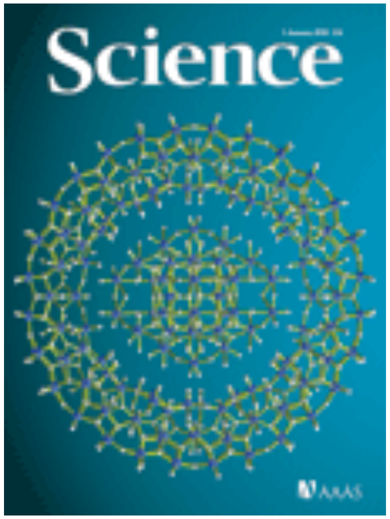
- Aim: Efficient Power generation: no emissions, no radioactive by-products
- Models investigate properties of tokamak reactor: how to build to withstand fusion environment
- Validate models against experiments

Industrial Modeling

Aerodynamics

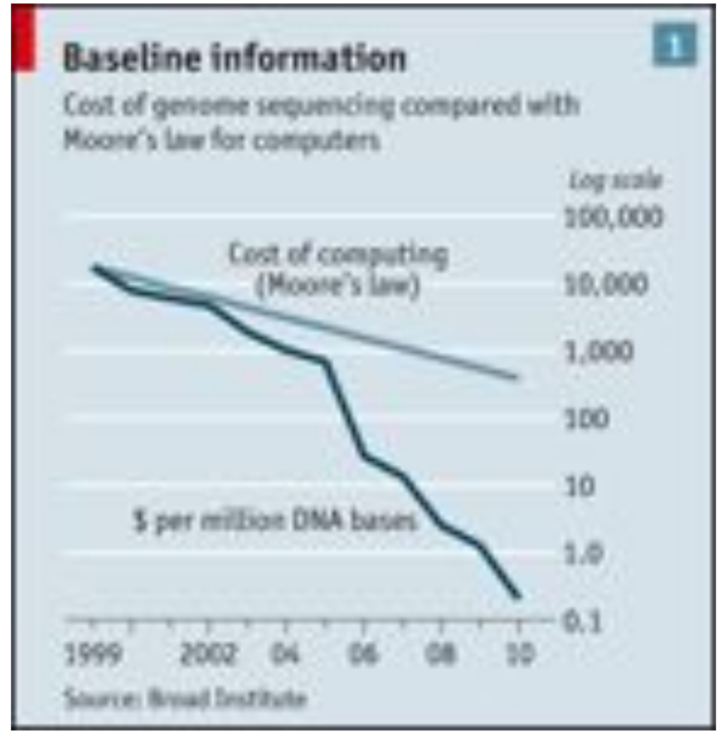
Porous Flow

Biology: The Next Big Driver of Computing?



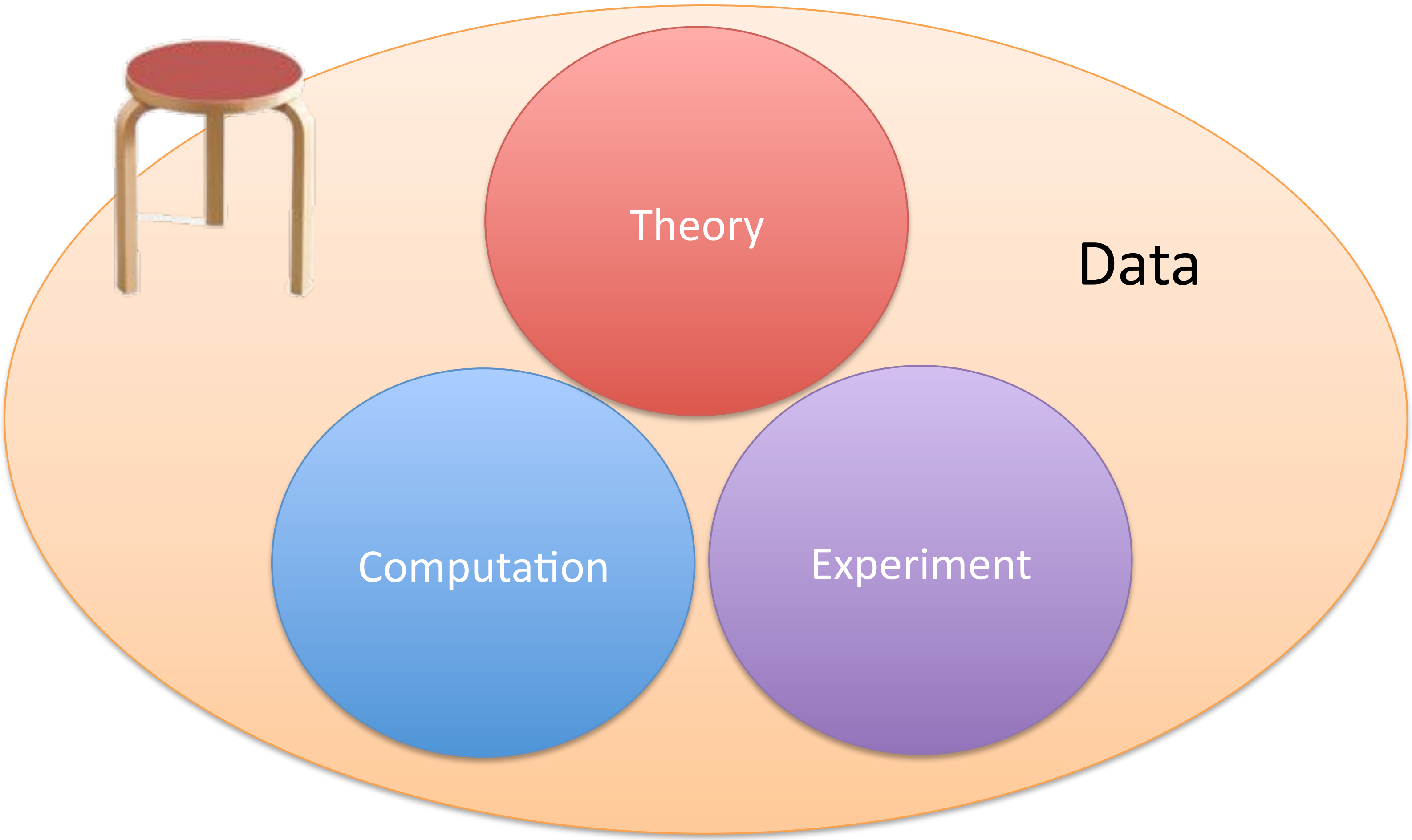
1 January 2010 Vol 327
Science, Issue 5961, Pages 36-37

Francis Collins,
Director NIH



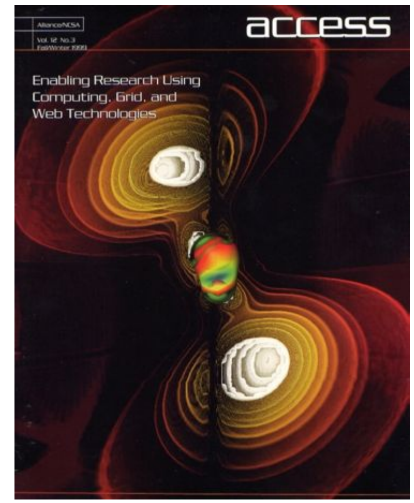
- Applying high throughput technologies
- Translating basic science discoveries into new and better treatments
- Benefiting health care reform
  - **Prevention and personalized medicine**
  - Health disparities research
  - Pharmacogenomics
  - Health research economics
- Focusing on global health

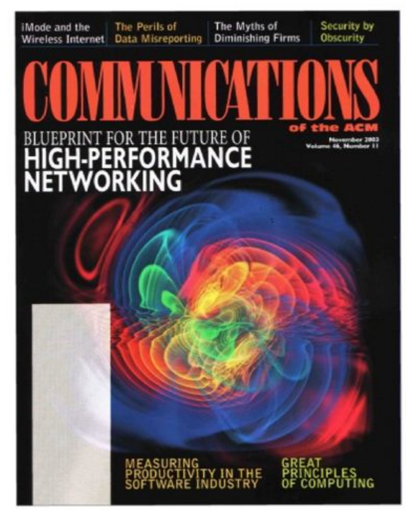
Modern Research: All Disciplines!!



Computational Science Needs

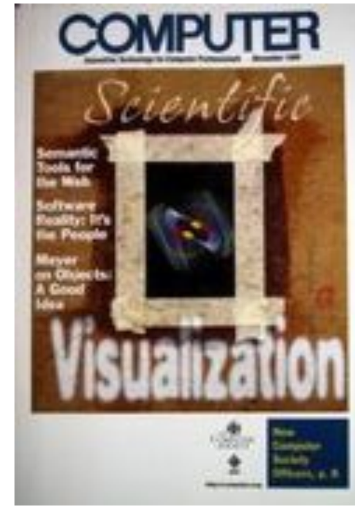
- Requires large mix of technologies & expertise!
- Collaboration across physics, engineering, computer science, applied mathematics, …
- numerical algorithms
  - Finite difference, spectral, monte carlo, elliptic
  - Multiscale, adaptive mesh refinement
- Different computational components
  - Large scale parallellization, new architectures
  - I/O, visualization
- New challenges
  - Petascale, data, complexity
- Complex infrastructure!
  - LONI, TeraGrid, DEISA, PRACE, Blue Waters, FutureGrid, Open Science Grid, …
- How to achieve all this? How to train students?

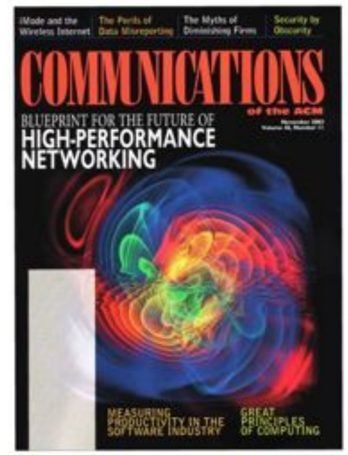


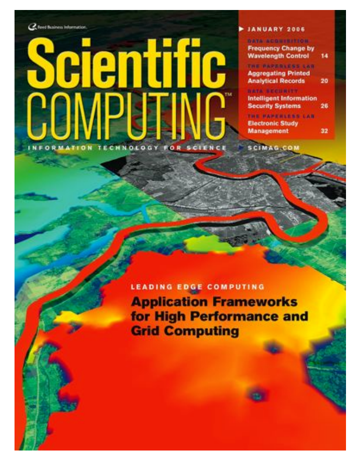


Many Research Issues

- Highly scalable algorithms (Petascale, exascale computing, heterogeneous architectures)
- Storage & networks: provenance, metadata
- Visualization (Large data, interactive, remote, AMR)
- Software engineering (code generation, verification & validation, interfaces, interoperability, etc)
- Workflows, Grid & distributed computing
- Education





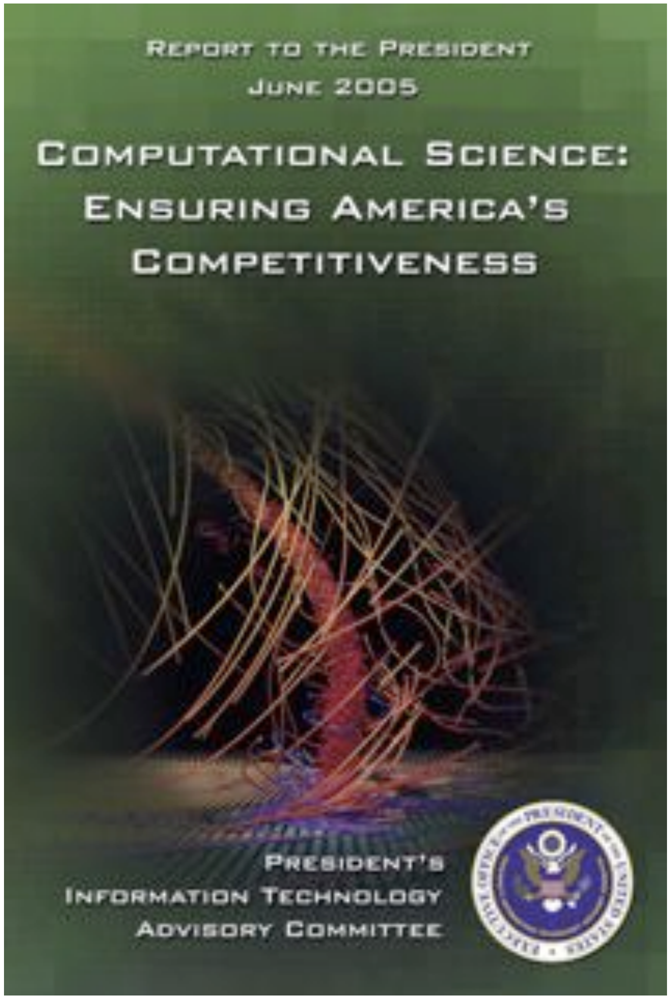


Nationally Relevant:
e.g. PITAC Report Summary:

"Computational science -- the use of advanced computing capabilities to understand and solve complex problems -- is critical to scientific leadership, economic competitiveness, and national security. It is one of the most important technical fields of the 21st century because it is essential to advances throughout society."

"Universities must significantly change organizational structures: multidisciplinary & collaborative research are needed [for US] to remain competitive in global science"

Collaborations for Complex problems: Innovations will occur at boundaries



CSC 7700 Motivation

- Generally recognized that we do not have curricula in place to educate students in computational science
  - Universities are changing: e.g. MS at LSU
- Lack of "real world" education puts our students at a disadvantage for modern research projects in scientific computing
- Experiences of the teaching faculty for this course!

Methodology and Modules

CSC 7700 Approach

- Design a course that prepares students for working in scientific computing research projects in academia or industry …
  - For CS/EE students: understand issues of scientific computing, current technologies, get real experience
  - For domain students: understand CS/EE research directions, current technologies, get real experience
  - Learn to collaborate
- Teaching faculty already working together (e.g. NSF CyberTools)
  - Regular meetings over summer to discuss curricula and approach

Modules

- Each focused on one specific area. Emphasis on
  - Broad overview relevant to modern scientific computing, more depth into selected specific technologies
  - Real world use of current technologies
- Curricula, lectures, coursework, exam by leader in the field
  - Overseen by Drs. Allen/Jha
- New course and curricula developed
  - Teaching team have weekly meetings to ensure coordination between modules

Modules

- A: Basic Skills (Loeffler)
- B: Networks and Data (Hutanu)
- C: Simulations and Application Frameworks (Schnetter)
- D: Scientific Visualization (Benger)
- E: Distributed Scientific Computing (Jha)

A: Basic Skills

- This module will review of basic knowledge and prerequisites for scientific computing from a computer science and cyberinfrastructure perspective.
- Topics that are likely to be covered include:batch systems; unix, shells and ssh; parallelization; compilation and make; performance analysis and strong/weak scaling; verification and validation; sources of errors and numerical precision; overview of numerical methods and partial differential equations.

B: Networks and Data

- This module covers network basics, advanced network topics, middleware and distributed computing basics, data-intensive computing and grid-based visualization.
- Practical experience: iperf, GridFTP, high-speed networks, web services.

C: Simulation and Application Frameworks

- This module covers: What is simulation (focus on IVPs), e.g. basic schedule, components involves. High level overview of PDEs and numerical approximation. Design of applications, focusing on application frameworks. Case study for Cactus, including interfaces, capabilities, complexities involved in cutting edge applications, current needs.
- Practical work will include: Compile and run Cactus applications, understand Cactus configuration files, write simple Cactus component including verification.

D: Scientific Visualization

- Covers: Scientific Visualization vs. Visualization - scope and concepts; the visualization pipeline; differential geometry and differentiable manifolds; vector calculus; geometric algebra; discretized manifolds; topology; object-oriented and generic programming in C++; rendering: raytracing vs. rasterization; OpenGL; vertex buffer objects; vertex, geometry and fragment shaders; shading models; data types for scientific visualization; file formats and I/O; HDF5; modular programming; software components
- Practical experience: Able to use Vish to visualize data, ability to modify Vish components for visualization. Simple data reader? Something with data description?

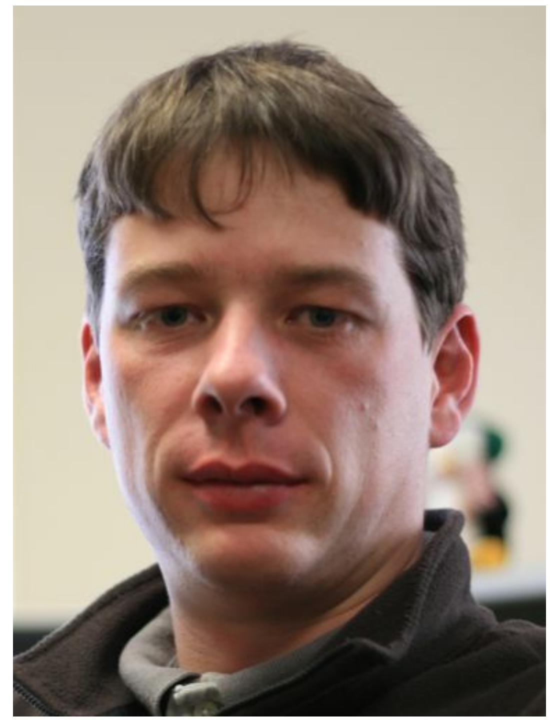
E: Distributed Scientific Computing

- This module will cover the theory and practice of Distributed Computing as applied to Scientific Applications. We will cover traditional Grids -- high-throughput as well as high-performance, as well as emerging infrastructure such as Clouds (eg Amazon EC2, Azure). We will use FutureGrid -- NSF's Track-2d Experimental Grid System as the test-bed.

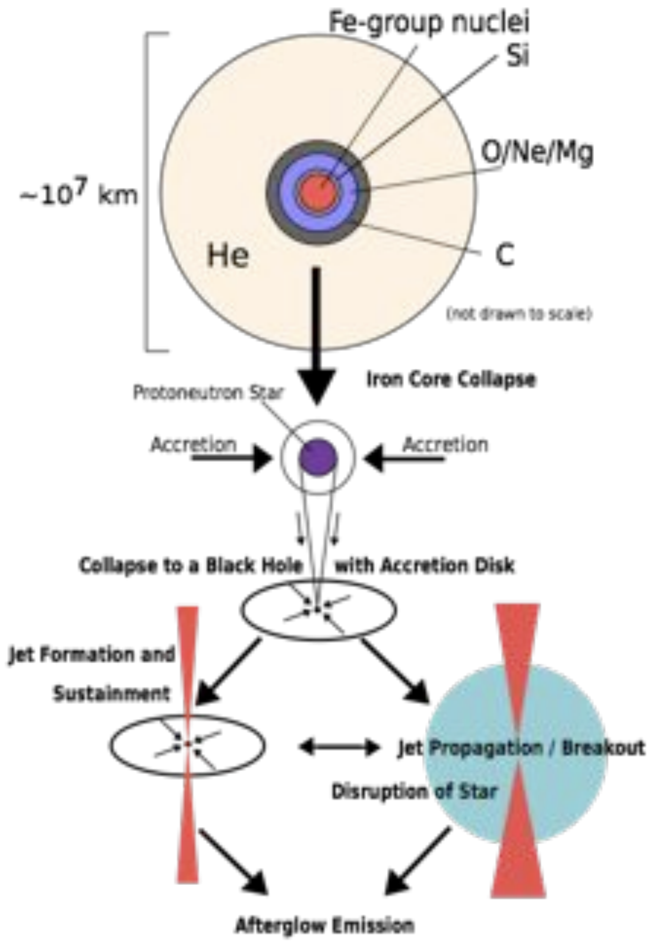
Teaching Faculty

Dr Frank Loeffler

- Module A: Basic Skills
- Postdoc, Center for Computation & Technology
- Relativistic astrophysics and high performance computing, lead developer for Einstein Toolkit Consortium



Gamma-Ray Burst Grand Challenge

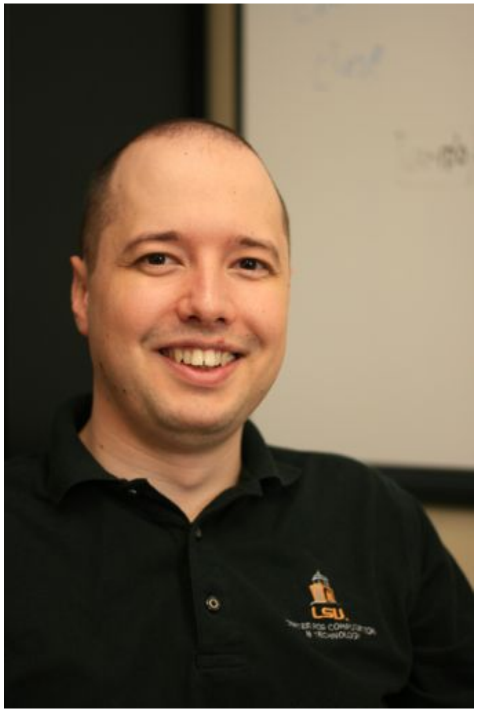


- <u>Most energetic events</u> in the universe
- Mechanism still a riddle; grand challenge in astrophysics
- Modeling requires expertise in many fields of physics (general relativity, magneto-hydrodynamics, neutrinos, ...)
- Requires petascale computing and collaboration

Einstein Toolkit
http://www.einsteintoolkit.org/

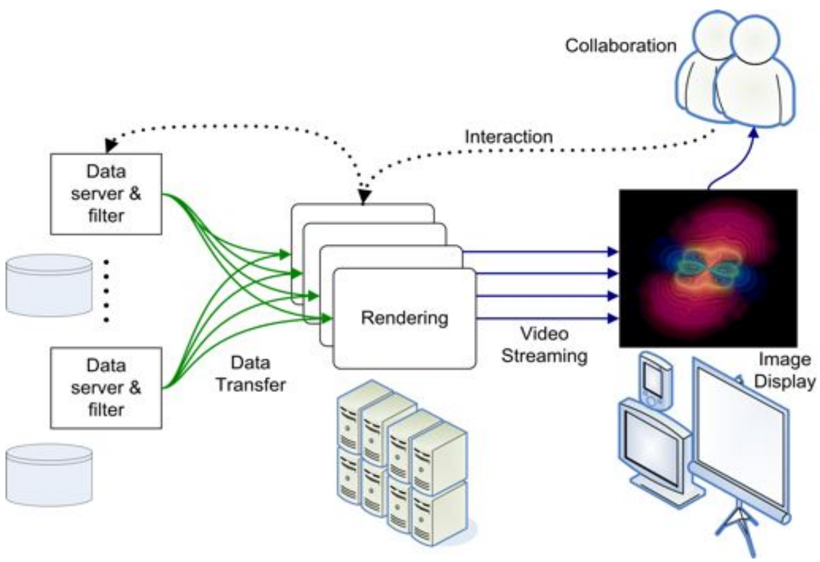
Dr Andrei Hutanu

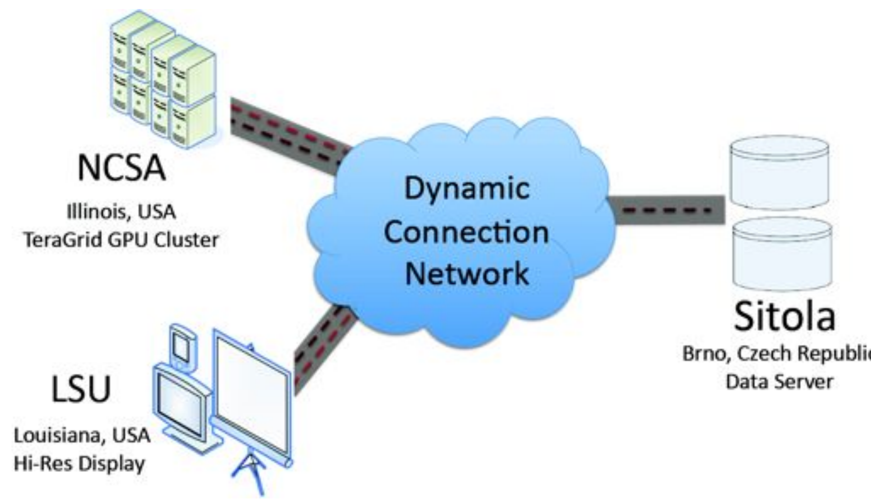
- Module B: Networks & Data
- Research Scientist, Center for Computation & Technology
- Applications of high speed networks, network testbeds, lead of eaviv project
- Lead of $300K NSF EAGER project



Remote Interactive Viz: eaviv
https://wiki.cct.lsu.edu/eaviv

- NSF Project (LSU/NCSA): Use new optical networks/services for dynamic configuration
- Target: Truly interactive visualization of large data using remote resources



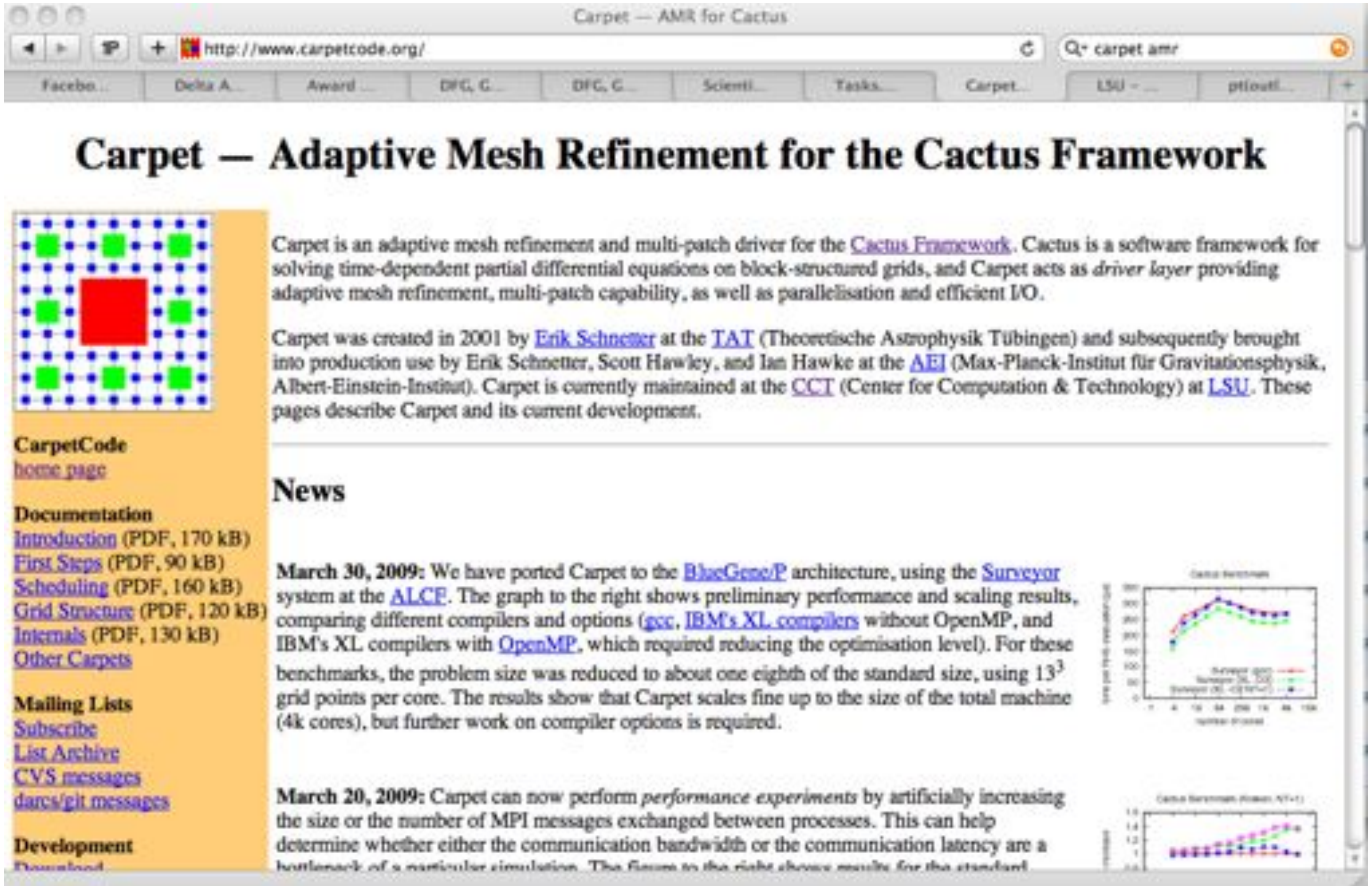


Dr Erik Schnetter

- Module C: Simulations & Application Frameworks
- Assistant Research Professor, Physics & Astronomy
- Relativistic astrophysics, Cactus Framework lead architect, originator of Carpet AMR infrastructure
- PI of $1.5M NSF PetaApps project

Cactus Framework
http://www.cactuscode.org

Carpet AMR Infrastructure http://www.carpetcode.org



Carpet — AMR for Cactus

http://www.carpetcode.org/

Carpet — Adaptive Mesh Refinement for the Cactus Framework

CarpetCode
home page

Documentation
Introduction (PDF, 170 kB)
First Steps (PDF, 90 kB)
Scheduling (PDF, 160 kB)
Grid Structure (PDF, 120 kB)
Internals (PDF, 130 kB)
Other Carpets

Mailing Lists
Subscribe
List Archive
CVS messages
darcs/git messages

Development

Carpet is an adaptive mesh refinement and multi-patch driver for the Cactus Framework. Cactus is a software framework for solving time-dependent partial differential equations on block-structured grids, and Carpet acts as *driver layer* providing adaptive mesh refinement, multi-patch capability, as well as parallelisation and efficient I/O.

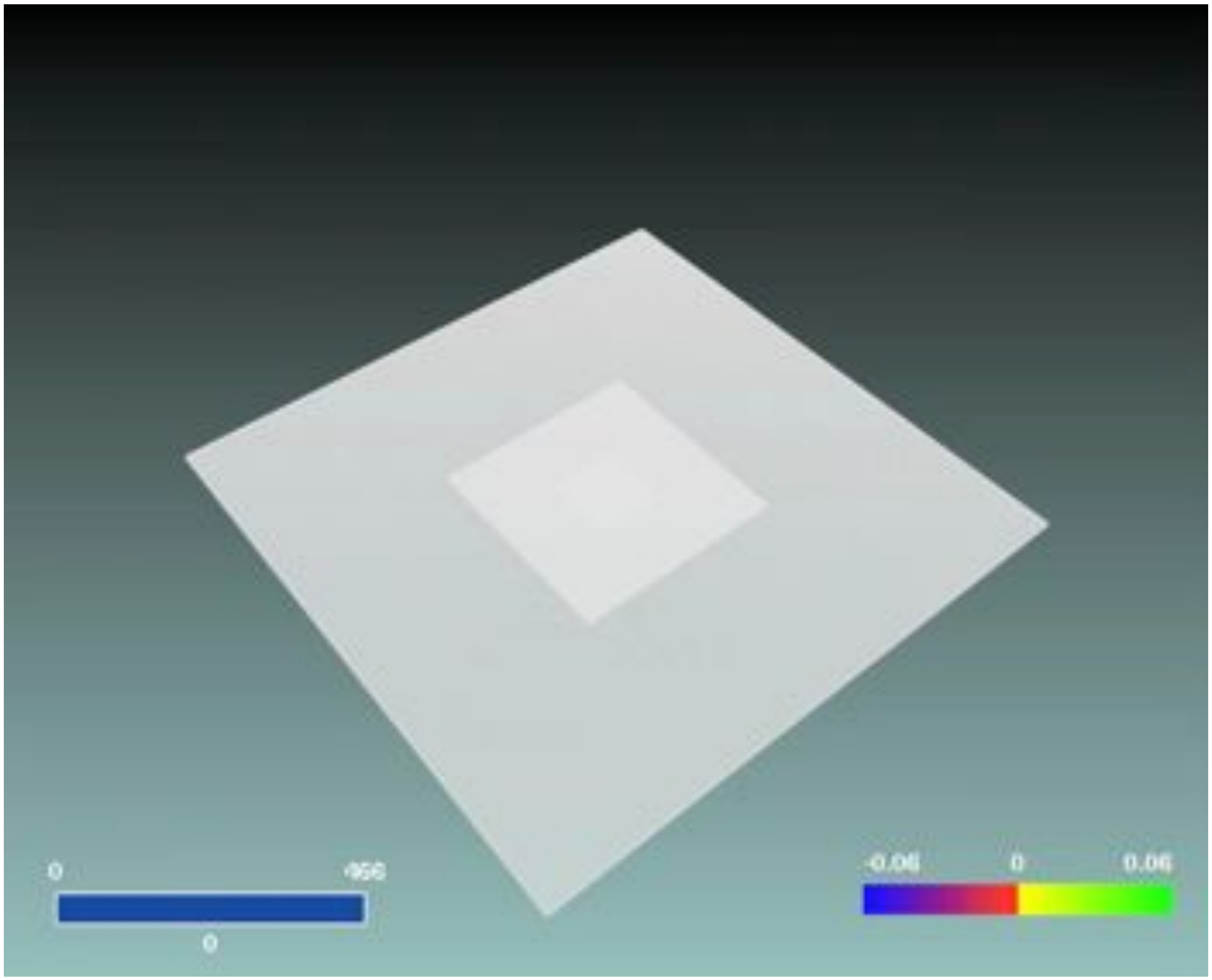
Carpet was created in 2001 by Erik Schnetter at the TAT (Theoretische Astrophysik Tübingen) and subsequently brought into production use by Erik Schnetter, Scott Hawley, and Ian Hawke at the AEI (Max-Planck-Institut für Gravitationsphysik, Albert-Einstein-Institut). Carpet is currently maintained at the CCT (Center for Computation & Technology) at LSU. These pages describe Carpet and its current development.

News

March 30, 2009: We have ported Carpet to the BlueGene/P architecture, using the Surveyor system at the ALCF. The graph to the right shows preliminary performance and scaling results, comparing different compilers and options (gcc, IBM's XL compilers without OpenMP, and IBM's XL compilers with OpenMP, which required reducing the optimisation level). For these benchmarks, the problem size was reduced to about one eighth of the standard size, using 13^3 grid points per core. The results show that Carpet scales fine up to the size of the total machine (4k cores), but further work on compiler options is required.

March 20, 2009: Carpet can now perform *performance experiments* by artificially increasing the size or the number of MPI messages exchanged between processes. This can help determine whether either the communication bandwidth or the communication latency are a

Carpet AMR Infrastructure

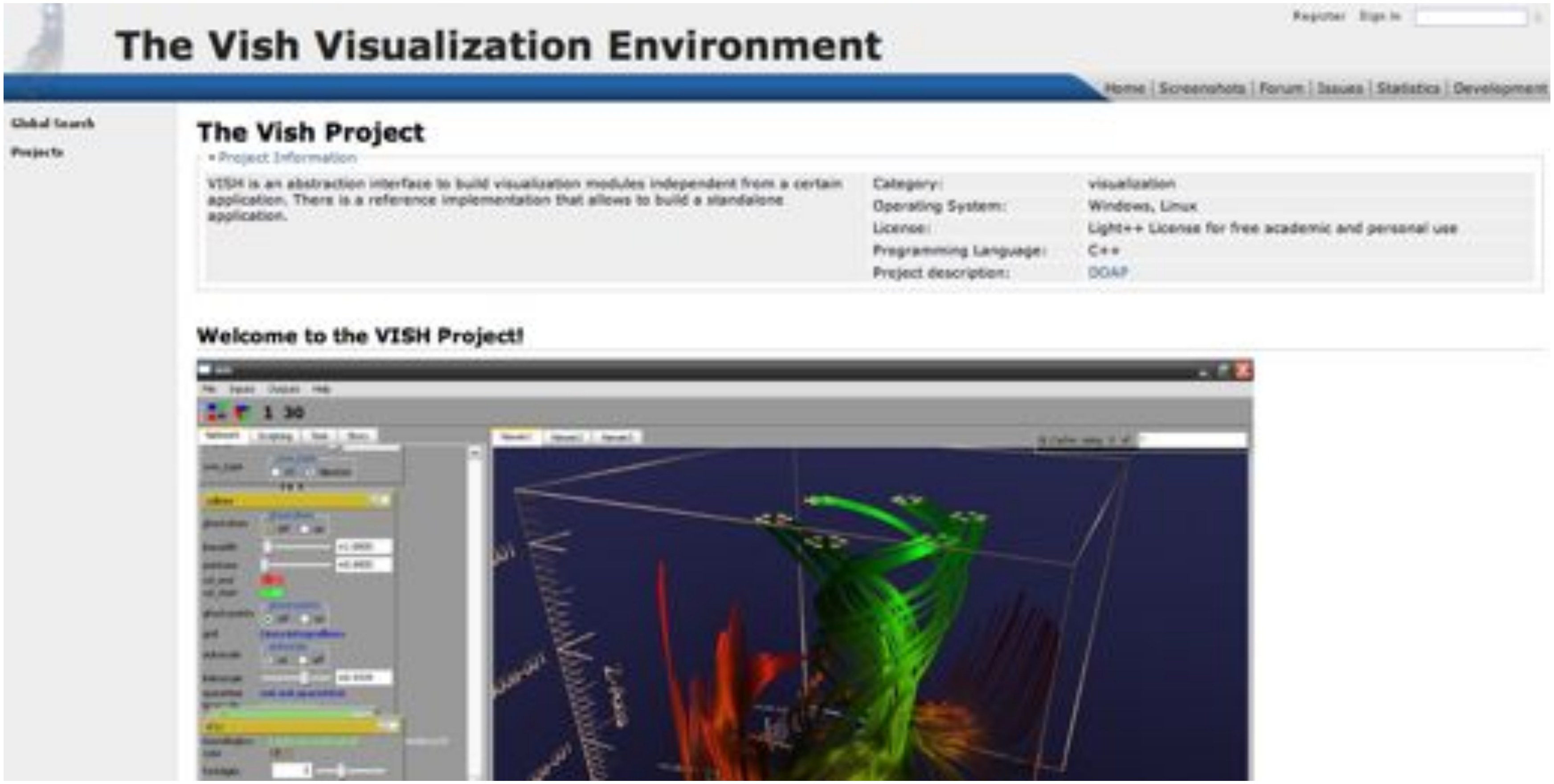


AEI (Rezzolla, Kaehler)

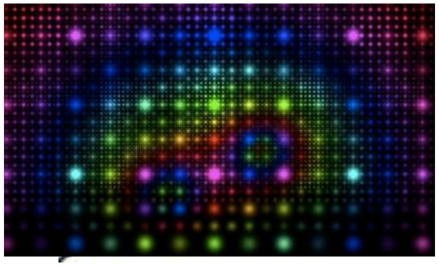
Dr Werner Benger

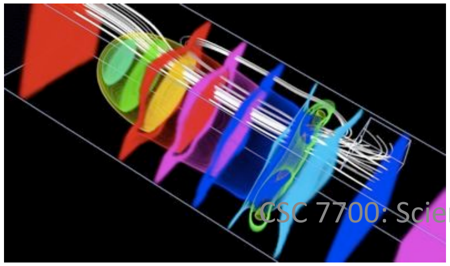
- Module D: Advanced Visualization
- Research Scientist, Center for Computation & Technology
- Originator of the Vish visualization framework, expert in data models, advanced viz techniques etc.
- Created most of the images you will see around CCT & in these slides.

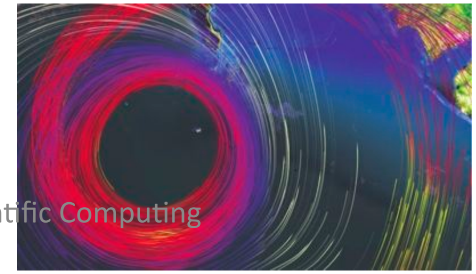
Vish

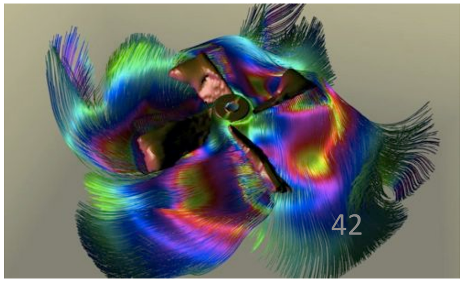


Register Sign In
The Vish Visualization Environment
Home | Screenshots | Forum | Issues | Statistics | Development
Global Search
Projects
The Vish Project
Project Information
VISH is an abstraction interface to build visualization modules independent from a certain application. There is a reference implementation that allows to build a standalone application.
Category: visualization
Operating System: Windows, Linux
License: Light++ License for free academic and personal use
Programming Language: C++
Project description: DOAP
Welcome to the VISH Project!





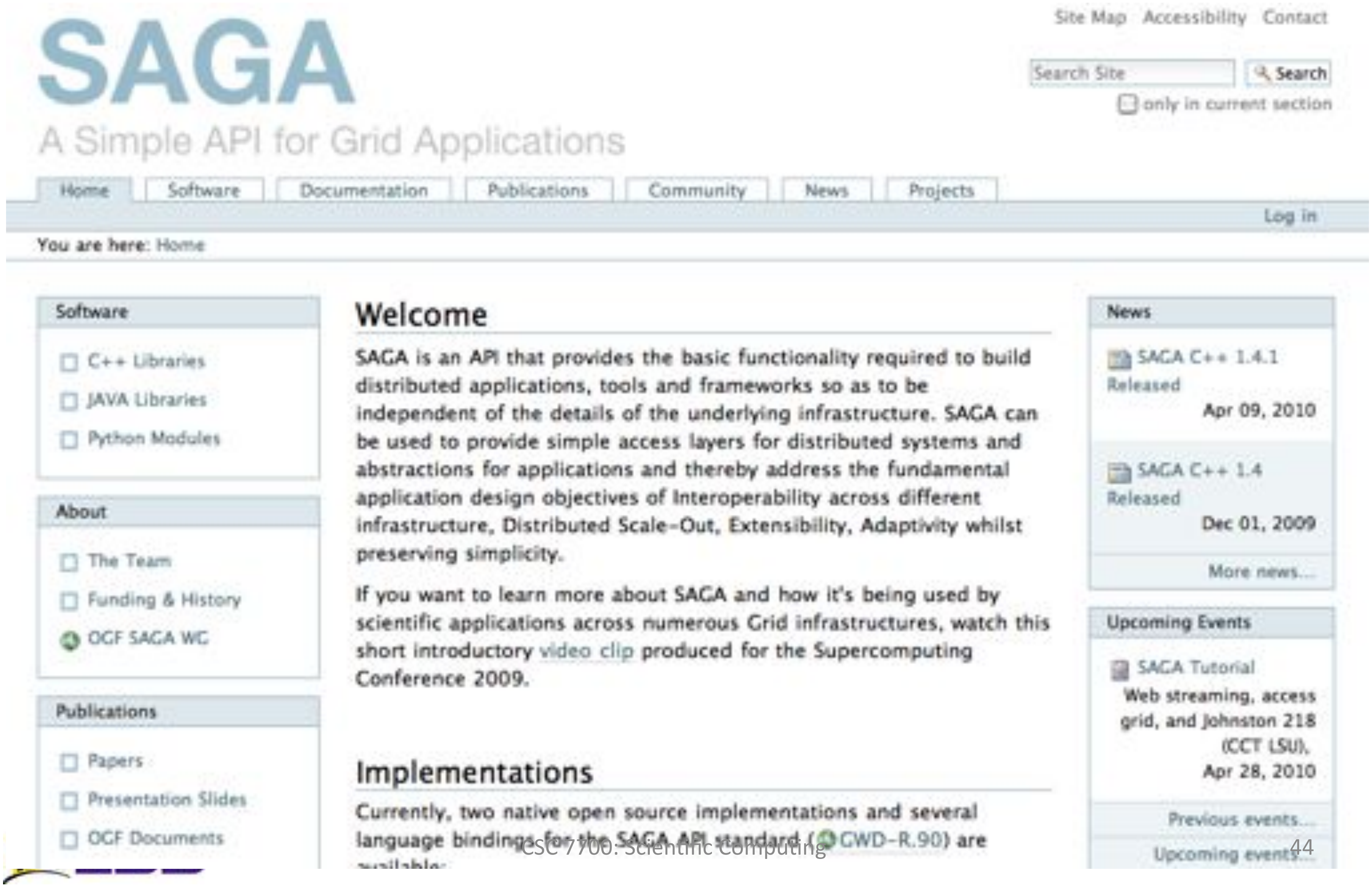




Dr Shantenu Jha

- Module E: Distributed Scientific Computing
- Assistant Research Professor, Computer Science, Director for Cyberinfrastructure Development CCT
- Lead of the SAGA project

Simple API for Grid Applications http://saga.cct.lsu.edu/

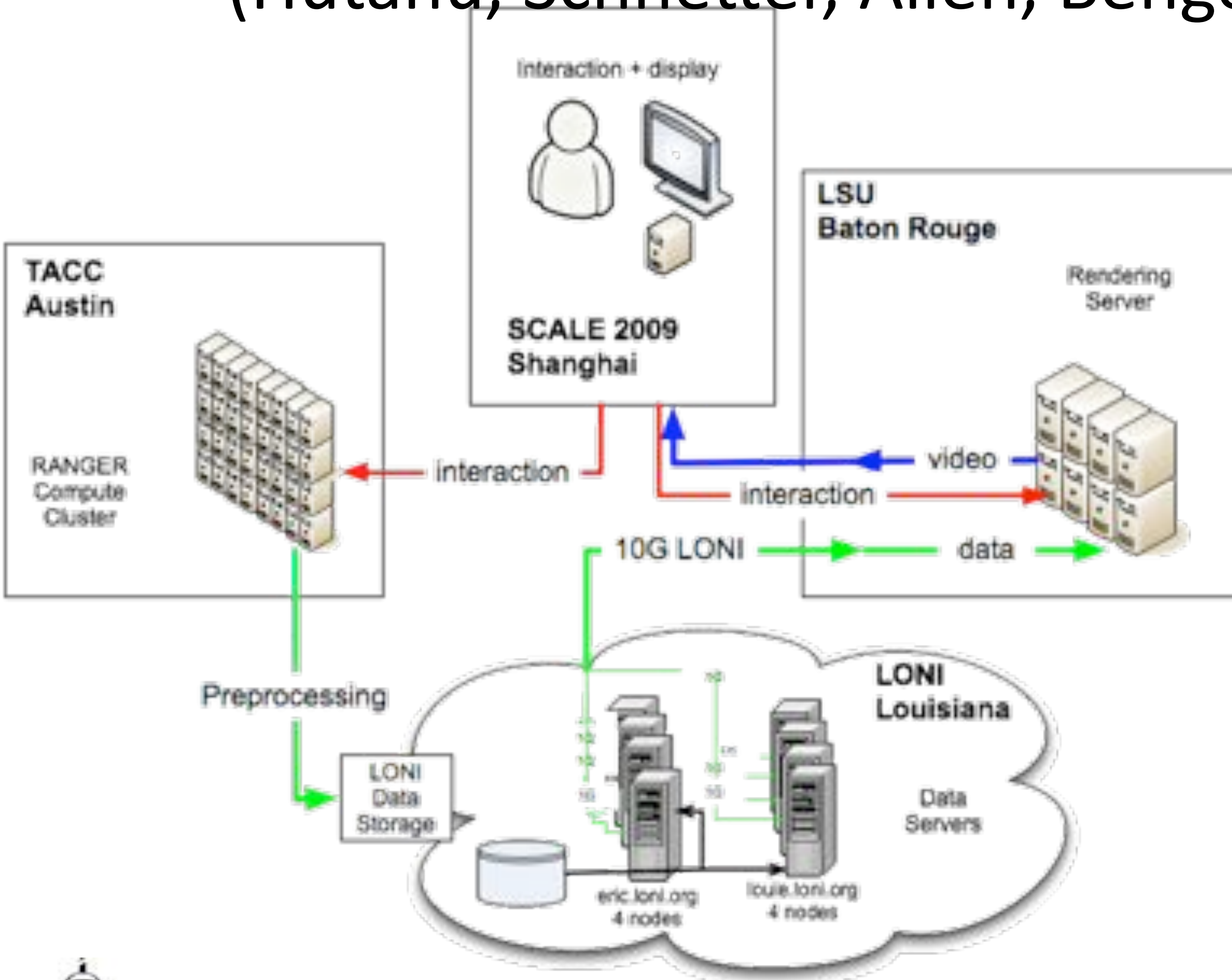


Common Goals

- Technology enabling scientific research
- Science motivating technology research
- Contributing to education

- Example: Cactus Framework
  – Over 200 science publications using Cactus
  – Over 50 student thesis

Example: Winners IEEE SCALE09!
(Hutanu, Schnetter, Allen, Benger, et al)



- 2048 core simulation code
- 8 node renderer at LSU
- Remote LONI nodes as data servers
- 1024^3 spatial resolution (1GByte/timestep)
- 20 timesteps cached remotely
- Scalable Approach!!!

Grading

CSC 7700: Grading

- Emphasis is on practical work and experience
- Aim is not to have tricky exam questions, but to convince us that you have acquired real experience
- This is the real world! Don't leave coursework till the last night, you will need to put in real hours and solve issues yourself or with colleagues

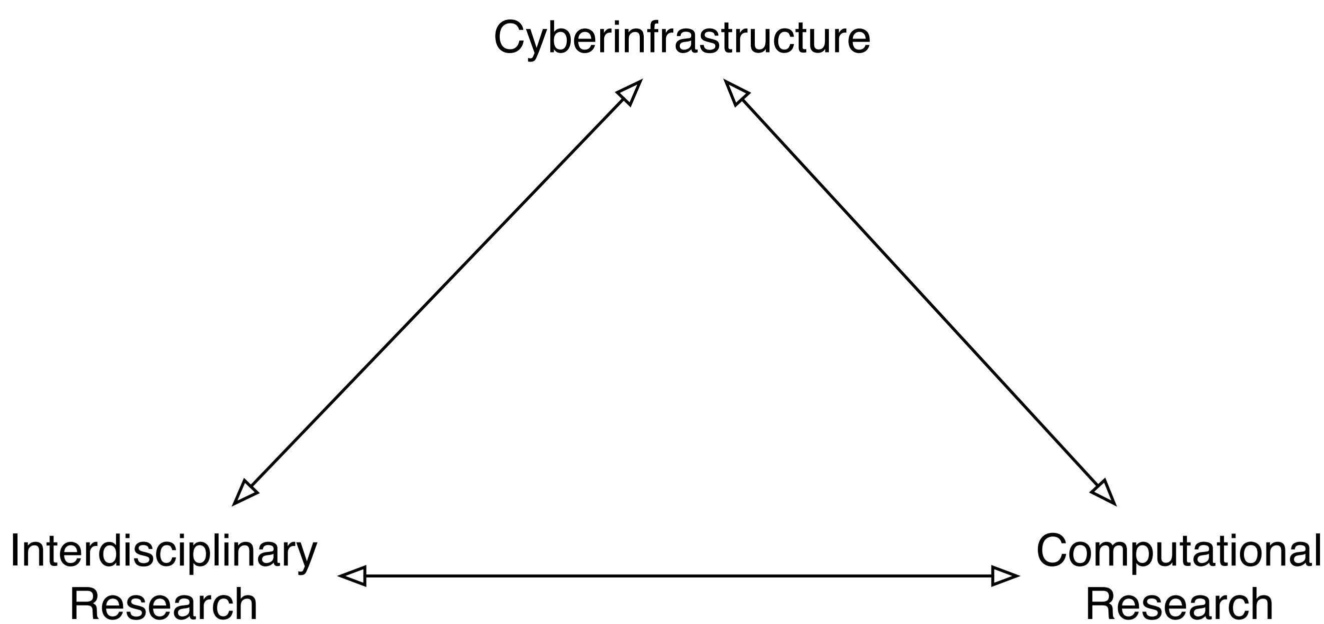
CSC 7700: Grading

- The class will be graded by module with a final exam which will cover all modules.
- The breakdown of grades between modules and the final exam is
  - 20% Final exam
  - 20% Module B: Networks and Data
  - 20% Module C: Simulations and Application Frameworks
  - 20% Module D: Scientific Visualization
  - 20% Module E: Distributed Scientific Computing
  - Within each module, half of the grades will be for coursework, with the other half awarded either by project, essay, or exam depending on the module.

Cyberinfrastructure Resources

Center for Computation & Technology:
Computational Science Initiative at LSU

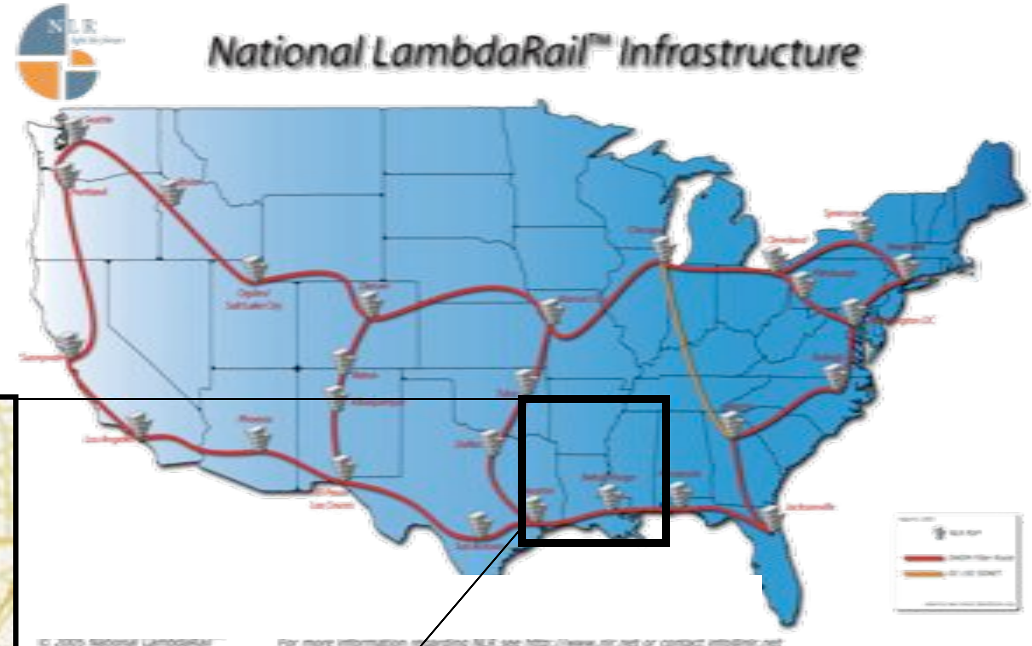
- Established 2003: Vision 20/20 in Louisiana
- Over 35 joint faculty with many departments

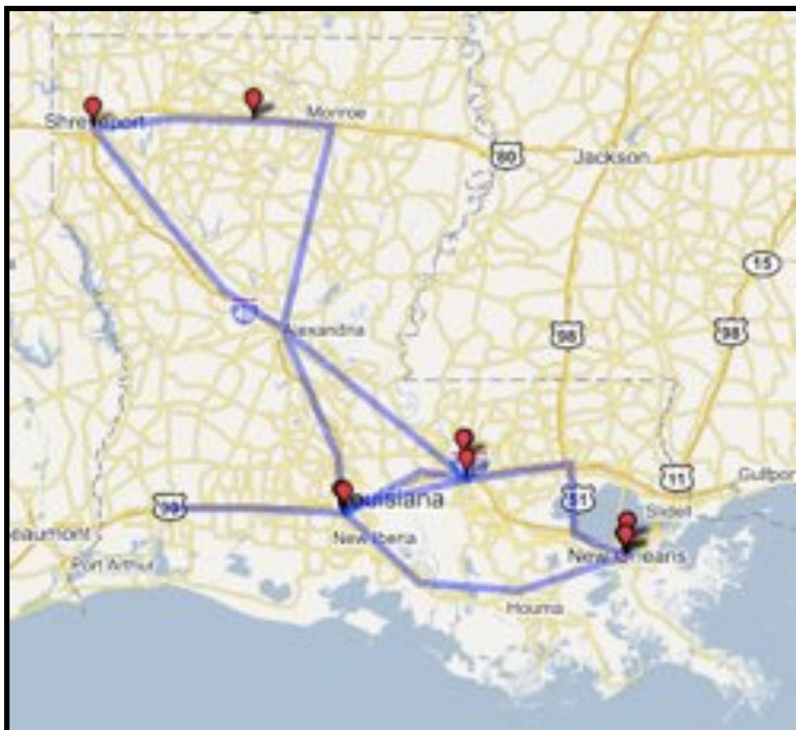


Louisiana Optical Network (LONI)
http://www.loni.org

State initiative ($50M) to support research (2004):

- 40 Gbps optical network + NLR
- Connects state universities, health science centers



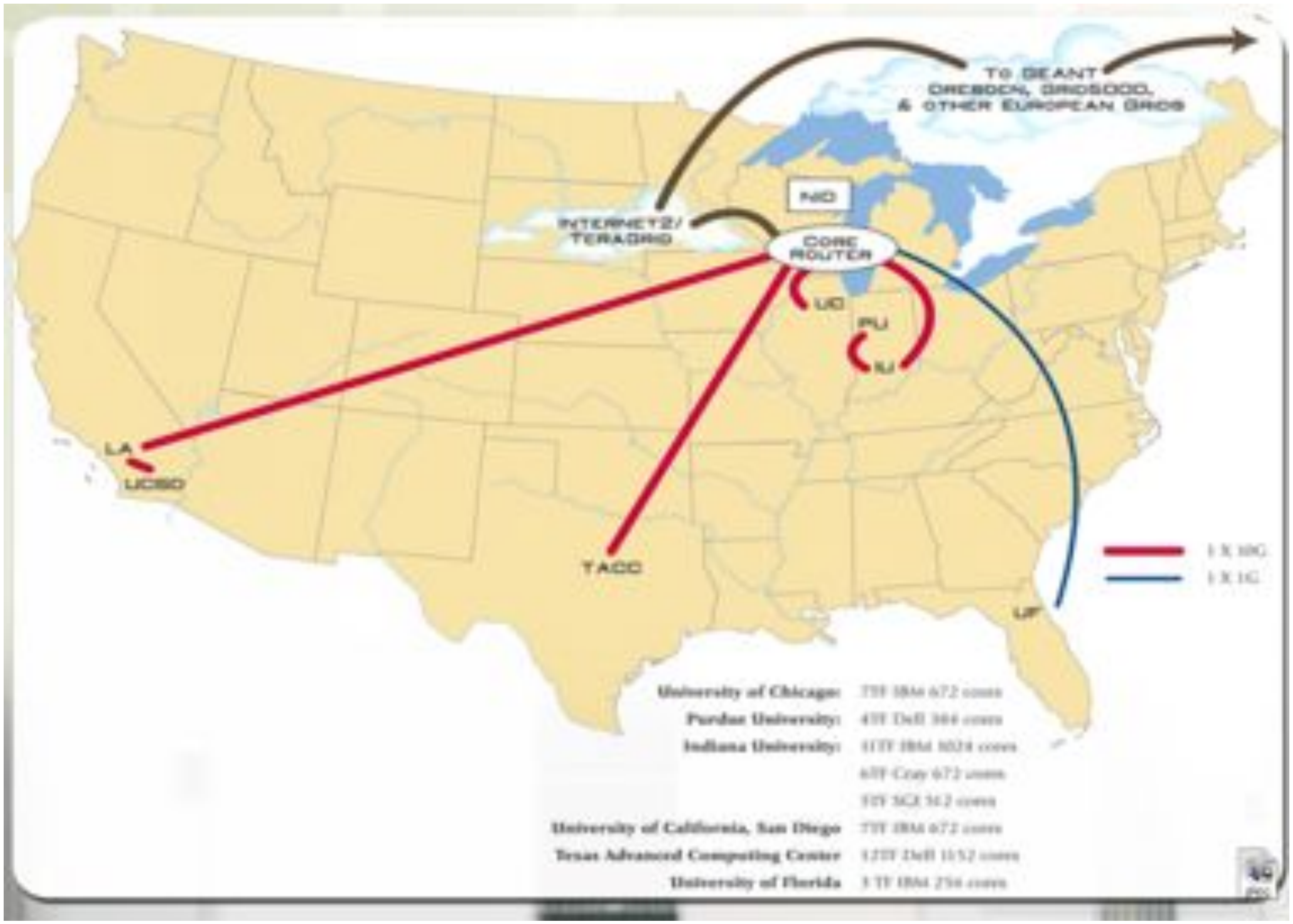


- Compute resources: ~100 Tflops across state
- Data resources ~500TB with NSF PetaShare
- LONI customers: MS universities, K12, hospitals, LPB

NSF TeraGrid
http://www.teragrid.org

TeraGrid™

NSF FutureGrid
http://www.futuregrid.org



Accounts for Class

- Educational Allocation for TeraGrid
  - https://wiki.cct.lsu.edu/sci-comp/TeraGrid_machines
  - Modules A, B, C
  - Abe, Lonestar, Ranger, Queen Bee, Steele, MSS
  - Most accounts available now (hand out)
  - Only use for this class!!!!
- Will have allocation on FutureGrid
  - Module E
  - Accounts being organized

This Class is for You!

- Ask questions, give us suggestions, let us know what you don't know or understand
- Let us know right away about any problems particularly as we start the practical work
- Plan to request your participation in initial and final survey so we can analyze the effect of this class.