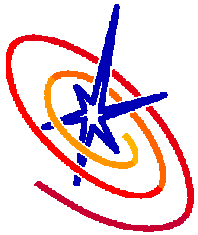


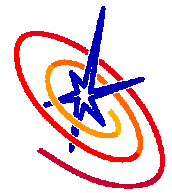
Issues in Advanced Computing: A US Perspective

Robert Rosner

Enrico Fermi Institute and Departments of Astronomy & Astrophysics and Physics, The University of Chicago
and
Argonne National Laboratory

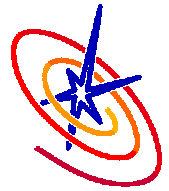
Bologna, Italy, July 5, 2002

An outline of what I will discuss



- Defining “advanced computing”
  - “advanced” vs. “high-performance”
2. Overview of scientific computing in the US today
   1. Where, with what, who pays, … ?
   2. What has been the “roadmap”?
   3. The challenge from Japan
3. What are the challenges?
   1. Technical
   2. Sociological
4. What is one to do?
   1. Hardware: What does $600M ($2M/$20M/$60M) per year buy you?
   2. Software: What does $4.0M/year for 5 years buy you?
5. Conclusions

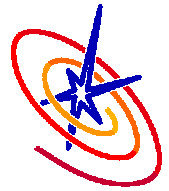
Advanced vs. high-performance computing



- "Advanced computing" encompasses frontiers of computer use
  - Massive archiving/data bases
  - High performance networks and high data transfer rates
  - Advanced data analysis and visualization techniques/hardware
  - Forefront high-performance computing (= peta/teraflop computing)

- "High-performance computing" is a tiny subset, and encompasses frontiers of
  - Computing speed ("wall clock time")
  - Application memory footprint

Ingredients of US advanced computing today

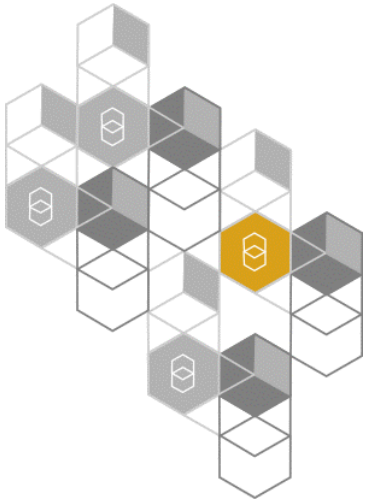


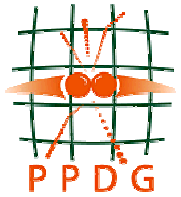
- Major program areas
  - Networking: Teragrid, IWIRE, …
  - Grid Computing: Globus, GridFTP, …
  - Scalable numerical tools: DOE/ASCI and SciDAC, NSF CS
  - Advanced visualization: Software, computing hardware, displays
  - Computing hardware: Tera/Petaflop initiatives

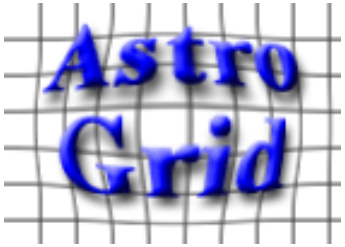
2. The major advanced computing science initiatives
   - Data-intensive science (incl. “data mining”)
     - Virtual observatories, digital sky surveys, bioinformatics, LHC science, …
   2. Complex systems science
      1. Multi-physics/multi-scale numerical simulations
      2. Code verification and validation

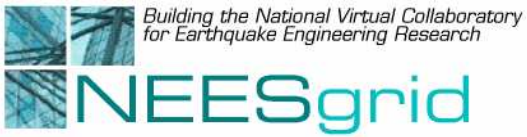
Example: Grid Science

EUROGRID









Specific Example: Sloan Digital Sky Survey Analysis

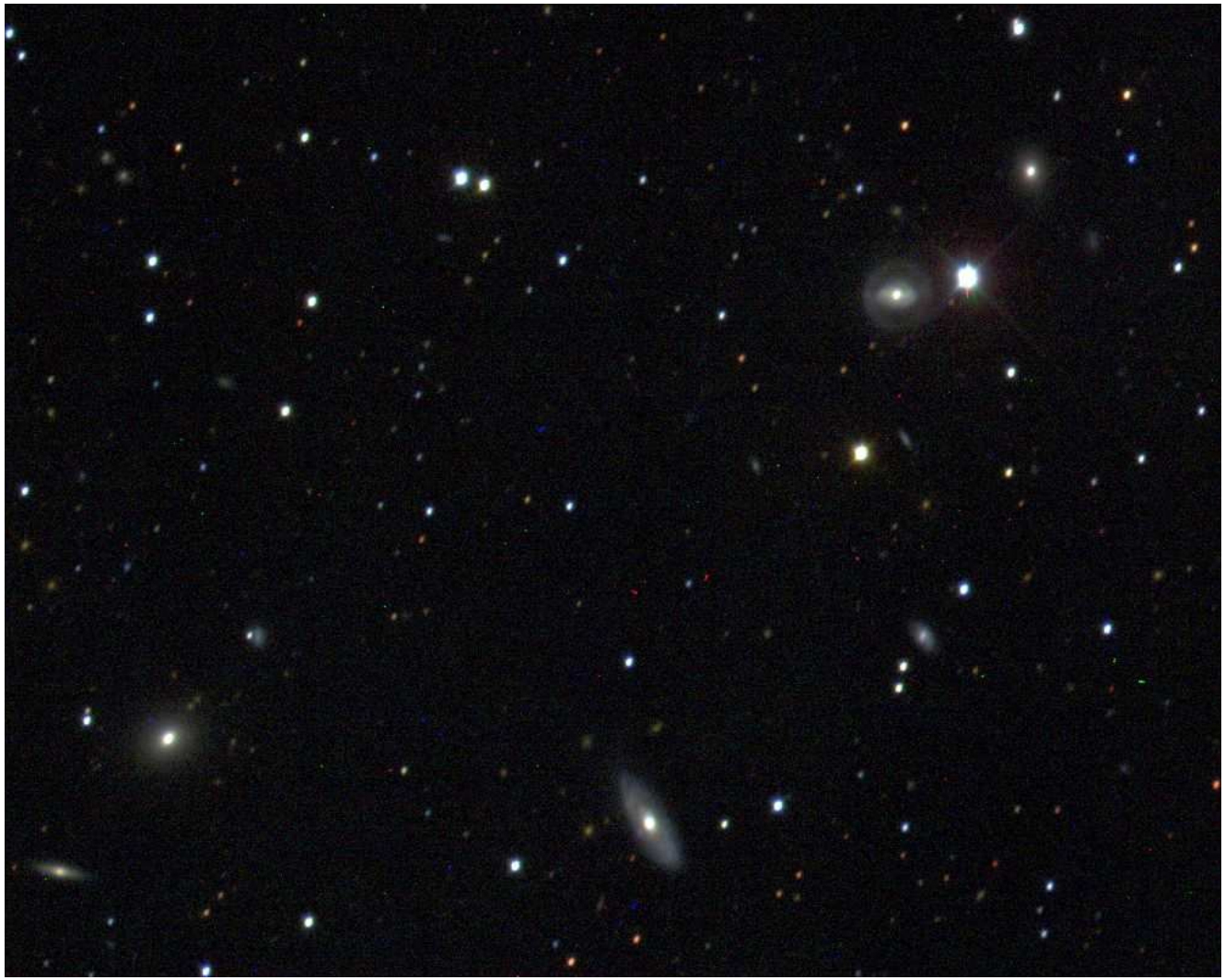
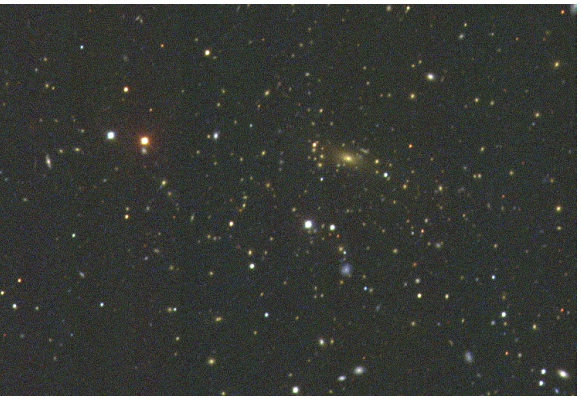


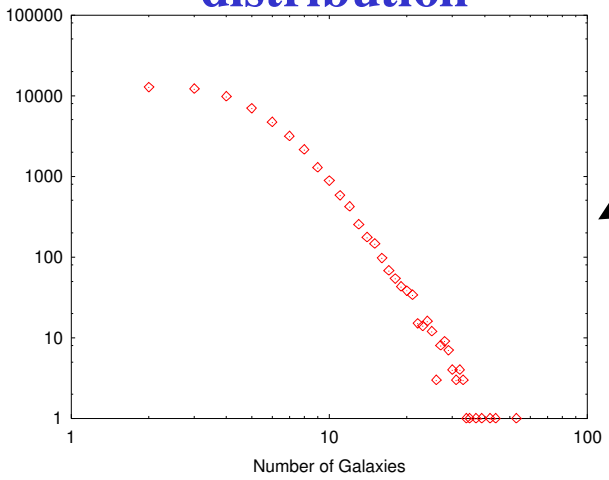
Image courtesy SDSS

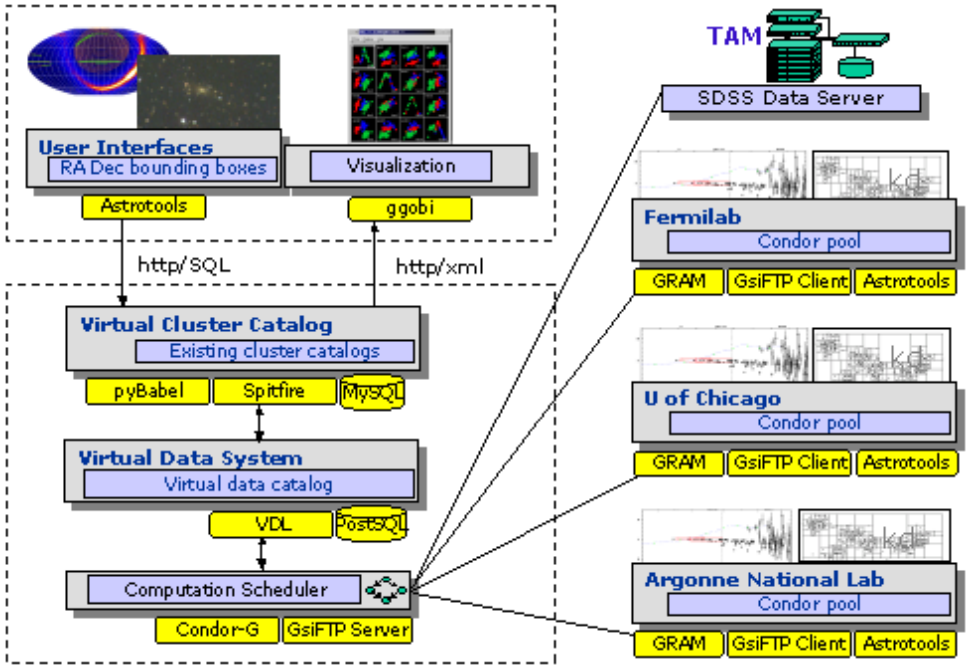
Specific Example: Sloan Digital Sky Survey Analysis

Size distribution of galaxy clusters?



Galaxy cluster size distribution



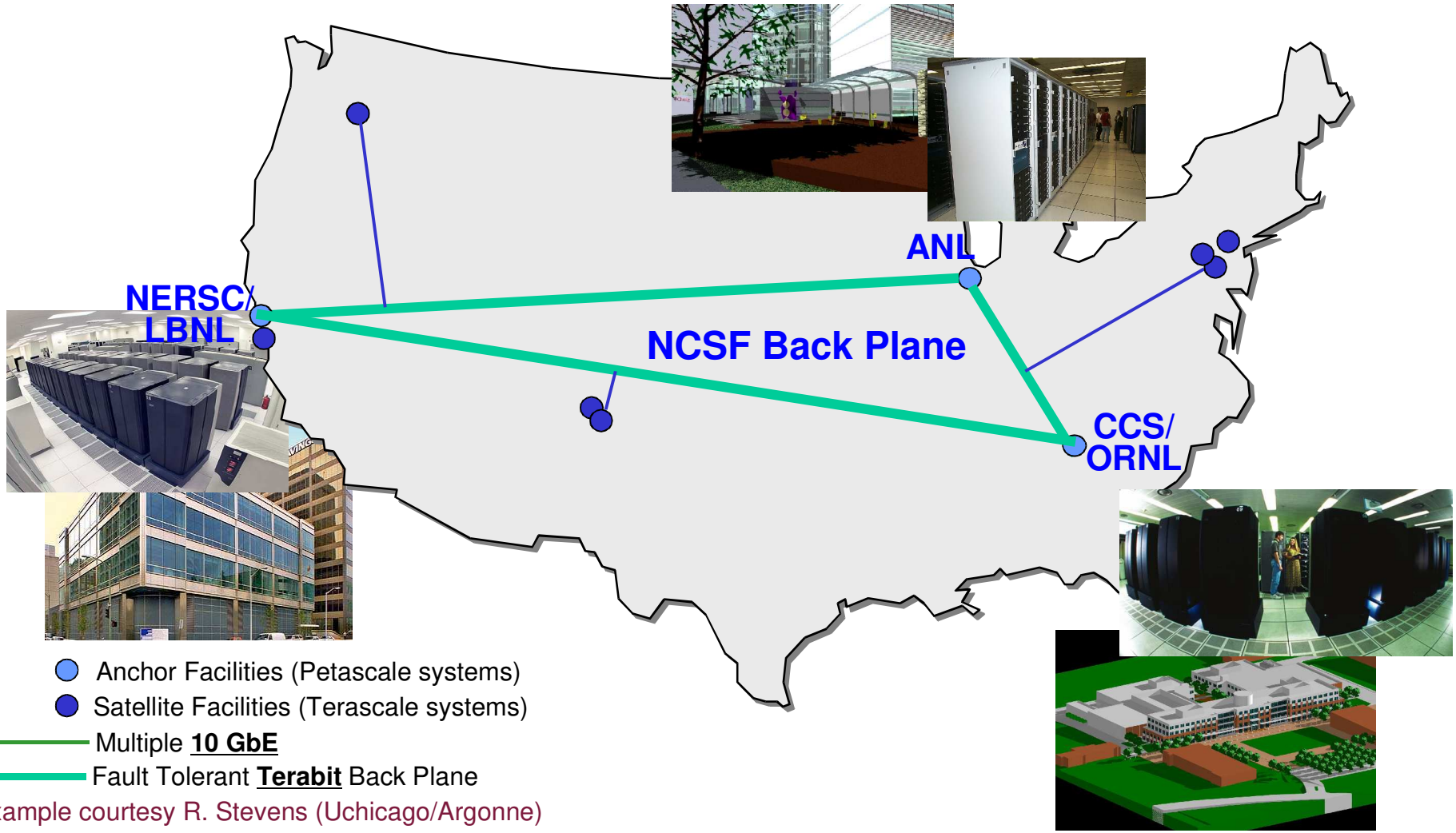


Chimera Virtual Data System + iVDGL Data Grid (many CPUs)

Example courtesy I. Foster (Uchicago/Argonne)

Specific Example: Toward Petaflop Computing

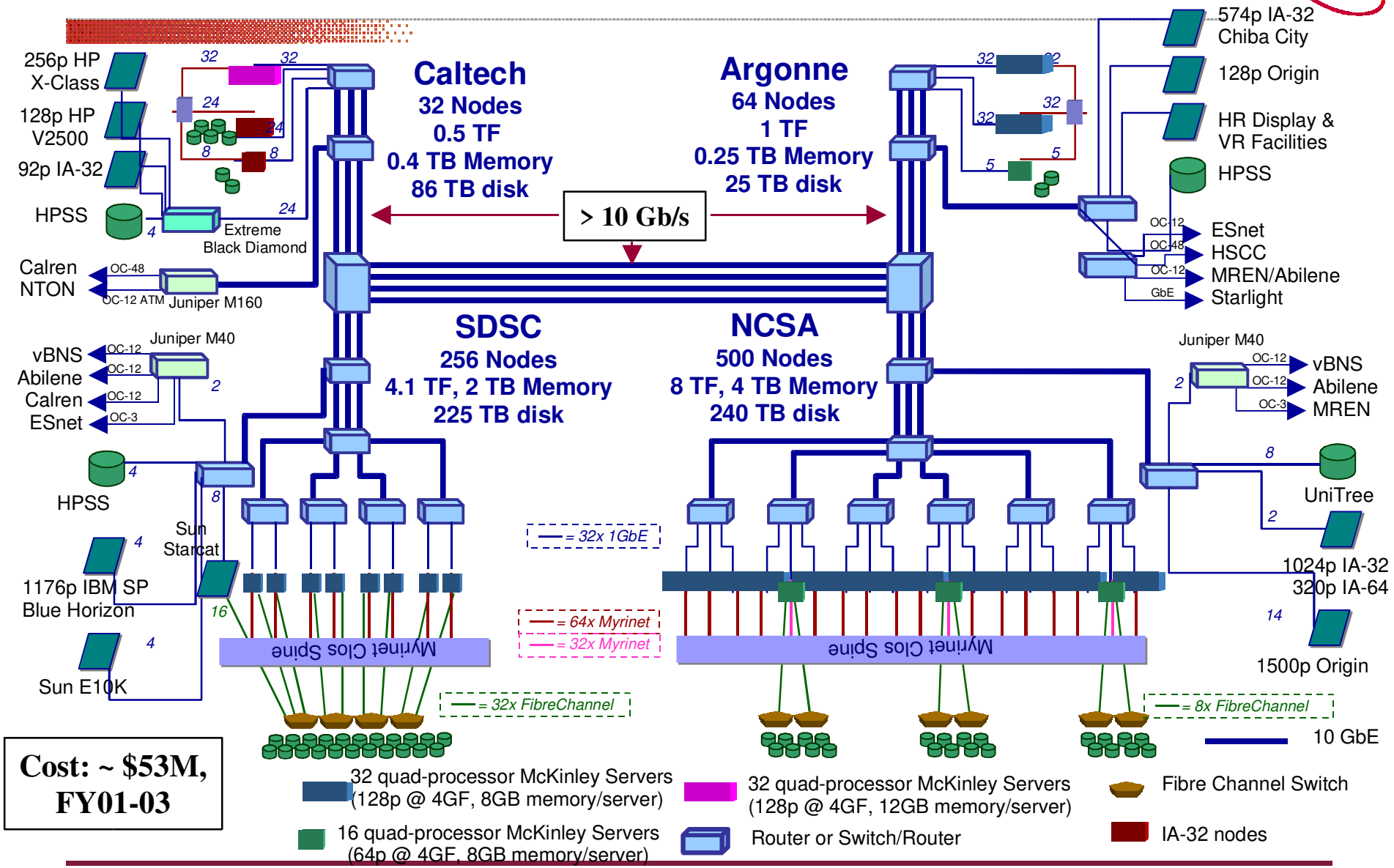
Proposed DOE Distributed National Computational Sciences Facility



- Anchor Facilities (Petascale systems)
- Satellite Facilities (Terascale systems)
- Multiple **10 GbE**
- Fault Tolerant **Terabit** Back Plane

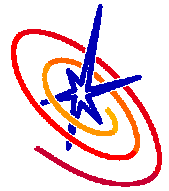
Example courtesy R. Stevens (Uchicago/Argonne)

Specific Example: NSF-funded 13.6 TF Linux TeraGrid



Cost: ~ $53M, FY01-03

Re-thinking the role of computing in science

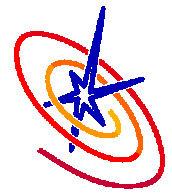


- Computer science (= informatics) research is typically carried out as a traditional academic-style research operation
  - Mix of basic research (applied math, CS, …) and applications (PETSc, MPICH, Globus, …)
  - Traditional "outreach" meant providing packaged software to others

2. New intrusiveness/ubiquity of computing
   - Opportunities
     1. E.g., integrate computational science into the natural sciences
     2. Computational science as the fourth component of astrophysical science:

        Observations ⟷ Theory ⟷ Experiment ⟷ Computational science
   2. **The key step:**
      1. **To motivate and drive informatics developments by the applications discipline**

What are the challenges? The hardware …

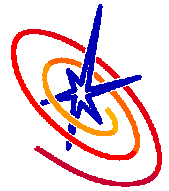


- Staying along Moore's Law trajectory
- Reliability/redundancy/"soft" failure modes
  - The ASCI Blue Mountain experience …
3. Improving "efficiency"
   1. "efficiency" = actual performance/peak performance
   2. Typical #s on tuned codes for US machines ~ 5-15% (!!)
   3. Critical issue: memory speed vs. processor speed
   4. US vs. Japan: do we examine hardware architecture?
4. Network speed/capacity
5. Storage speed/capacity
6. Visualization
   1. Display technology
   2. Computing technology (rendering, ray tracing, …)

What are the challenges? The software …

- Programming models
  1. MPI vs. OpenMP vs. …
  2. Language interoperability (F77, F90/95, HPF, C, C++, Java, …)
     - Glue languages: scripts, Python, …
2. Algorithms
  1. Scalability
  2. Reconciling time/spatial scalings (example: rad. hydro)
  3. Data organization/data bases
  4. Data analysis/visualization
- Coding and code architecture
  1. Code complexity (debugging, optimization, code repositories, access control, V&V)
  2. Code reuse & code modularity
  3. Load balancing

What are the challenges? The sociology …



How do we get astronomers, applied mathematicians, computer scientists, … to talk to one another productively?

- Overcoming cultural gap(s): language, research style, …
- Overcoming history
- Overcoming territoriality: who's in charge?
  - Computer scientists doing astrophysics?
  - Astrophysicists doing computer science?
- Initiation: top-down or bottom-up?
  - Anectodal evidence is that neither works well, if at all
- Possible solutions include:
  - Promote acculturation (mix): Theory institutes and Centers
  - Encourage collaboration: Institutional incentives/seed funds
  - Lead by example: construct "win-win" projects, change "other" to "us"
    - ASCI/Alliance centers at Caltech, Chicago, Illinois, Stanford, Utah

The Japanese example: focus

Information courtesy: Keiji Tani, Earth Simulator Research and Development Center, Japan Atomic Energy Research Institute

Atmospheric and oceanographic science

High resolution global models

Predictions of global warming, etc.

High resolution regional models

Predictions of El Niño events and Asian monsoons, etc.

High resolution local models

Predictions of weather disasters (typhoons, localized torrential downpours, downbursts, etc.)

Solid earth science

Global dynamic model

Describing the entire solid earth as a system

Regional model

Description of crust/mantle activity in the Japanese Archipelago region

Simulation of earthquake generation processes, seismic wave tomography

Other HPC applications: biology, energy science, space physics, etc.

Using the science to define requirements

Requirements for the Earth Simulator

- Necessary CPU capabilities for atmospheric circulation models:

		Present	Earth Simulator	CPU ops ratio
Horizontal mesh	Global model	50-100 km	5-10 km	~100
	Regional model	20-30 km	1 km	few 100s
Layers		several 10s	100-200	few 10s
Time mesh		1	1/10	10

- Necessary memory footprint for 10 km-mesh:

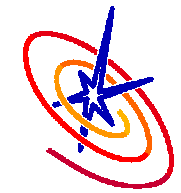
  Assume: 150-300 words for each grid point:
  4000×2000×200×(150-300)×2×8 = 3.84 - 7.68 TB

CPU must be at least 20 times faster than those of present computers for atmospheric circulation models; memory comparable to NERSC Seaborg.

- Effective performance, NERSC Glenn Seaborg: ~0.05*5 Tops ~ 0.25 Tops
- Effective performance of E.S.: > 5 Tops
- Main memory of E.S. : > 8 TB

What is the result, ~$600M later?

What is the result, ~$600M later?



- Architecture: MIMD-type, distributed memory, parallel system, consisting of computing nodes with tightly coupled vector-type multi-processors which share main memory
- Performance: Assuming an efficiency ε ~ 12.5%, the peak performance is ~ 40 TFLOPS (recently, achieved ε well over 30% [!!])
- The effective performance for atmospheric circulation model > 5 TFLOPS

	Earth Simulator	Seaborg
Total number of processor nodes:	640	208
Number of PE's for each node:	8	16
Total number of PE's:	5120	3328
Peak performance of each PE:	8 Gops	1.5 Gops
Peak performance of each node:	64 Gops	24 Gops
Main memory:	10 TB (total)	> 4.7 TB
Shared memory / node:	16 GB	16-64 GB
Interconnection network:	Single-Stage Crossbar Network	

The US Strategy: "Layering"

Small System
Computing Capability
0.1 GF – 10GF

Local (university)
resources

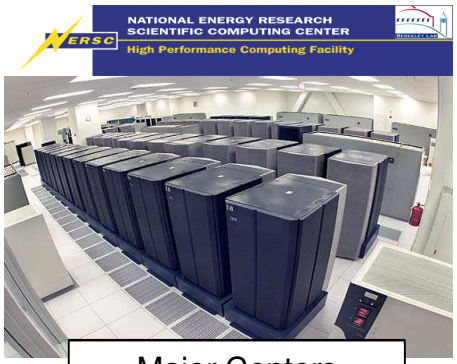
$3-5K capital costs
<$0.5K operating costs

Mid-Range
Computing/Archiving
Capability
~1.0 TF/~100 TB Archive

Local Centers
Example: Argonne

$3-5M capital costs
~$2-3M operating costs

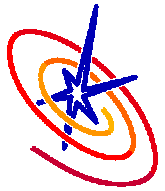
High-End
Computing Capability
10+ TF



Major Centers
Example: NERSC

>$100M capital costs
~$20-30M operating costs

The US example: focusing software advances



The DOE/ASCI challenge: how can application software development be sped up, and take advantage of latest advances in physics, applied math, computer science, … ?

The ASCI solution: do an experiment

- Create 5 groups at universities, in a variety of areas of "multi-physics"
  - Astrophysics (Chicago), shocked materials (Caltech), jet turbines (Stanford), accidental large-scale fires (U. Utah), solid fuel rockets (U. Illinois/Urbana)
- Fund well, at ~$20M total for 5 years (~$45M for 10 years)
- Allow each Center to develop its own computing science infrastructure
- Continued funding contingent on meeting specific, pre-identified, goals
- Results? See example, after 5 years!

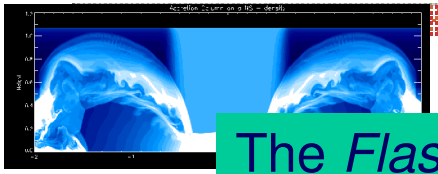
The SciDAC solution: do an experiment

- Create a mix of applications and computer science/applied math groups
- Create funding-based incentives for collaborations, forbid "rolling one's own" solutions
  - Example: application groups funded at ~15-30% of ASCI/Alliance groups
- Results? Not yet clear (effort ~ 1 year old)

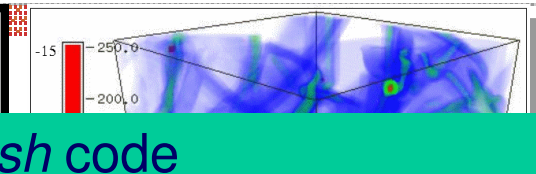
Example: The Chicago ASCI/Alliance Center

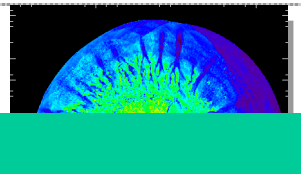
- Funded starting Oct. 1, 1997, 5-year anniversary Oct. 1, 2002, w/ possible extension for another 5 years
- Collaboration between
  - University of Chicago (Astrophysics, Physics, Computer Science, Math, and 3 Institutes [Fermi Institute, Franck Institute, Computation Institute])
  - Argonne National Laboratory (Mathematics and Computer Science)
  - Rensselear Polytechnic Institute (Computer Science)
  - Univ. of Arizona/Tuscon (Astrophysics)
  - "Outside collaborators": SUNY/Stony Brook (Relativistic rad. hydro), U. Illinois/Urbana (rad. hydro), U. Iowa (Hall mhd), U. Palermo (solar/time-dependent ionization), UC Santa Cruz (flame modeling), U. Torino (mhd, relativistic hydro)
- Extensive "validation" program with external experimental groups
  - Los Alamos, Livermore, Princeton/PPPL, Sandia, U. Michigan, U. Wisconsin

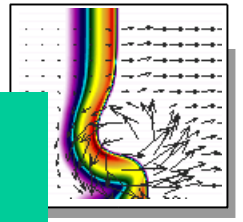
What does $4.0M/yr for 5 years buy?



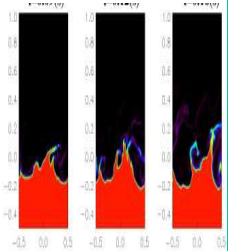
Relativistic accretio



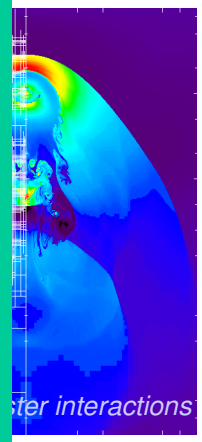




rtex interactions

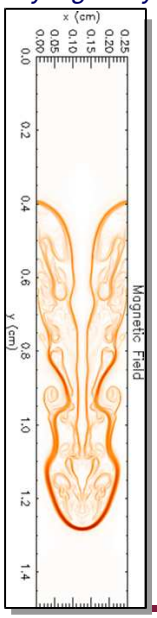


Wave breaking or



ter interactions

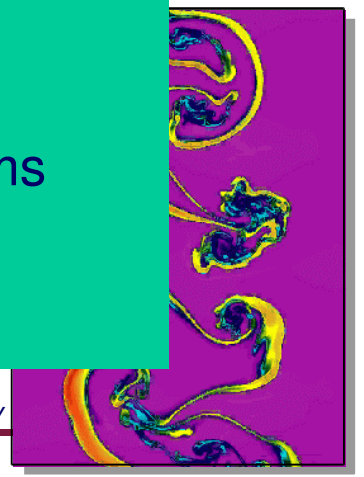
Magnetic Rayleigh-Taylor



The *Flash* code

2. Is modular
3. Has a modern CS-influenced architecture
4. Can solve a broad range of (astro)physics problems
5. Is highly portable
   a. Can run on all ASCI platforms
   b. Runs on all other available massively-parallel systems
5. Can utilize all processors on available MMPs

- Scales well, and performs well
- Is extensively (and constantly) verified/validated
- Is available on the web: http://flash.uchicago.edu
- Has won a major prize (Gordon Bell 2001)
- Has been used to solve significant science problems
  - (nuclear) flame modeling
  - Wave breaking

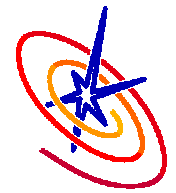


Cellular detonation

Helium burning on neutron stars

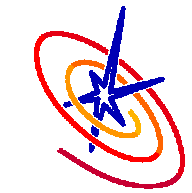
vortex
Richtmyer-Meshkov instability

Conclusions



- Key first step:
  1. **Answer the question**: Is the future imposed? planned? opportunistic?
  2. **Answer the question**: What is the role of various institutions, and of individuals?
  3. **Agree on specific science goals**
     - What do you want to accomplish?
     - Who are you competing with?
2. Key second steps:
   1. **Insure funding support for long-term (= expected project duration)**
   2. **Construct science "roadmap"**
   3. **Define specific science milestones**
- Key operational steps
  1. **Allow for early mistakes**
  2. **Insist on meeting specific science milestones by mid-project**

And that brings us to …



Questions and Discussion