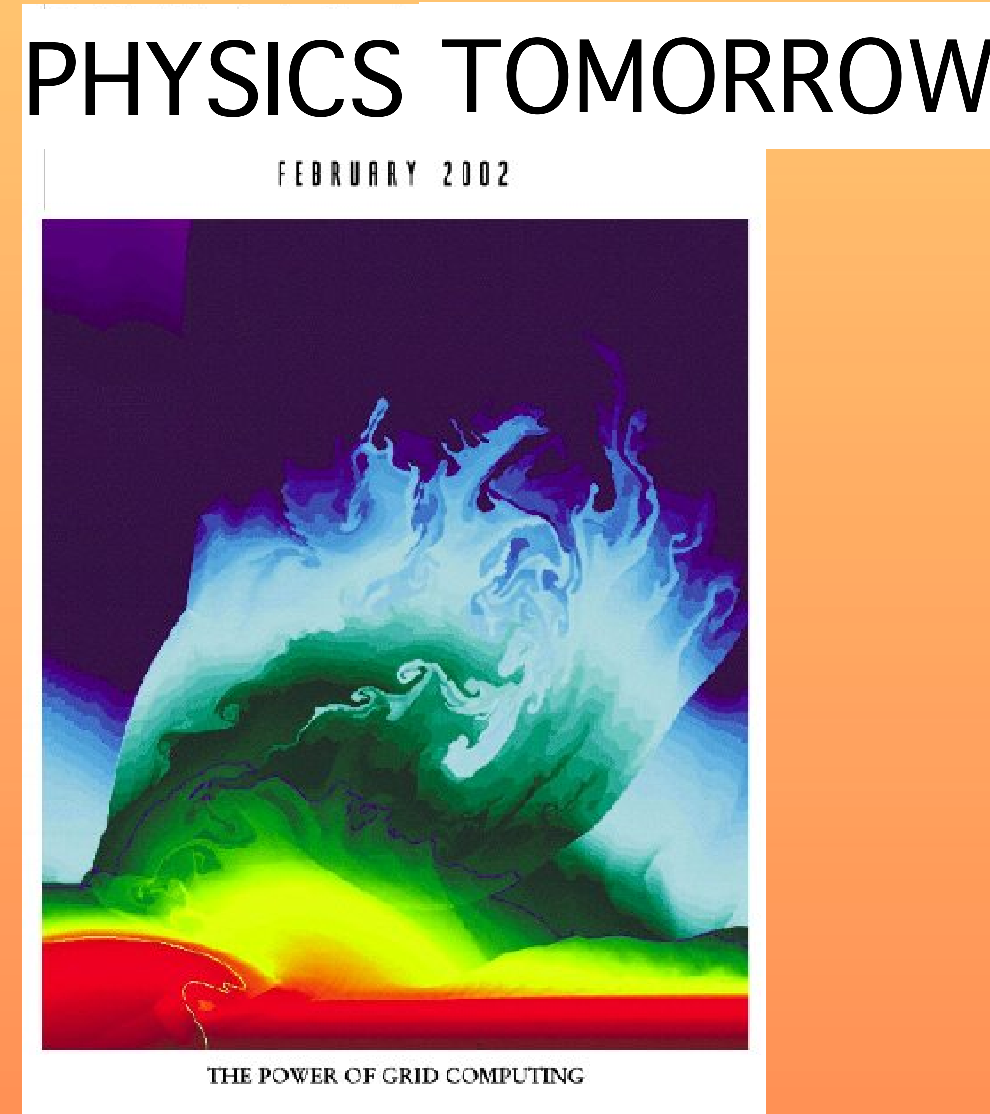
Grid Applications Today and Tomorrow



Ed Seidel

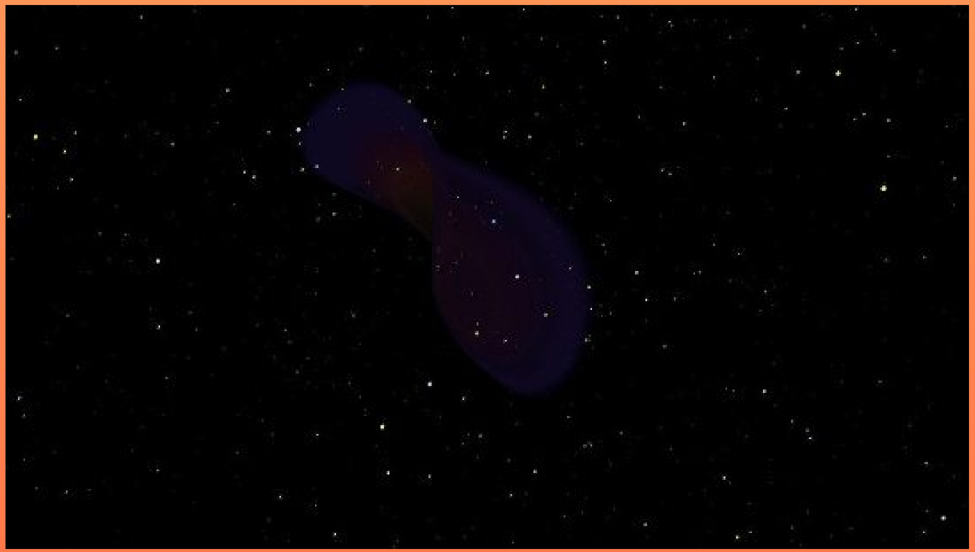
Director, Center for Computation & Technology, LSU

Co-Chair,
Applications Research Group, GGF
TWG-Apps, EU Gridstart Project

The Grid

Hot Area in Computational Science

- Computational Devices Scattered Across the World
  - Compute servers (double 18 months)
  - Handhelds
  - Networks (double each 9 months)
  - Playstations, cell phones
  - Sensors
- How to take advantage of this for science, engineering, business, art?
  - Harness multiple sites and devices
  - Make it accessible to scientists, engineers, artists
  - Integrate communities and resources



Current Grid App Types

- Comm
  - Distribu
  - Video
  - Virtual
- Data
  - Remo
  - Wea
  - Po
- Proces
  - Demanding Simulations of Science and Engineering
  - Task farming, resource brokering, distributed computations, workflow
- Interactive
  - Remote, steered, etc…

Present Examples:
- "simple but difficult"

Future Examples:
- Dynamic
- Interacting combinations

of all types: DDDAS, etc
e.g., LEAD, SCOOP, …

From Telephone Conference Calls to Access Grid Intern'l Video Meetings

New Cyber Arts

Humans Interacting with Virtual Realities

Internet Linked Pianos

Source: Smarr

Access Grid Lea

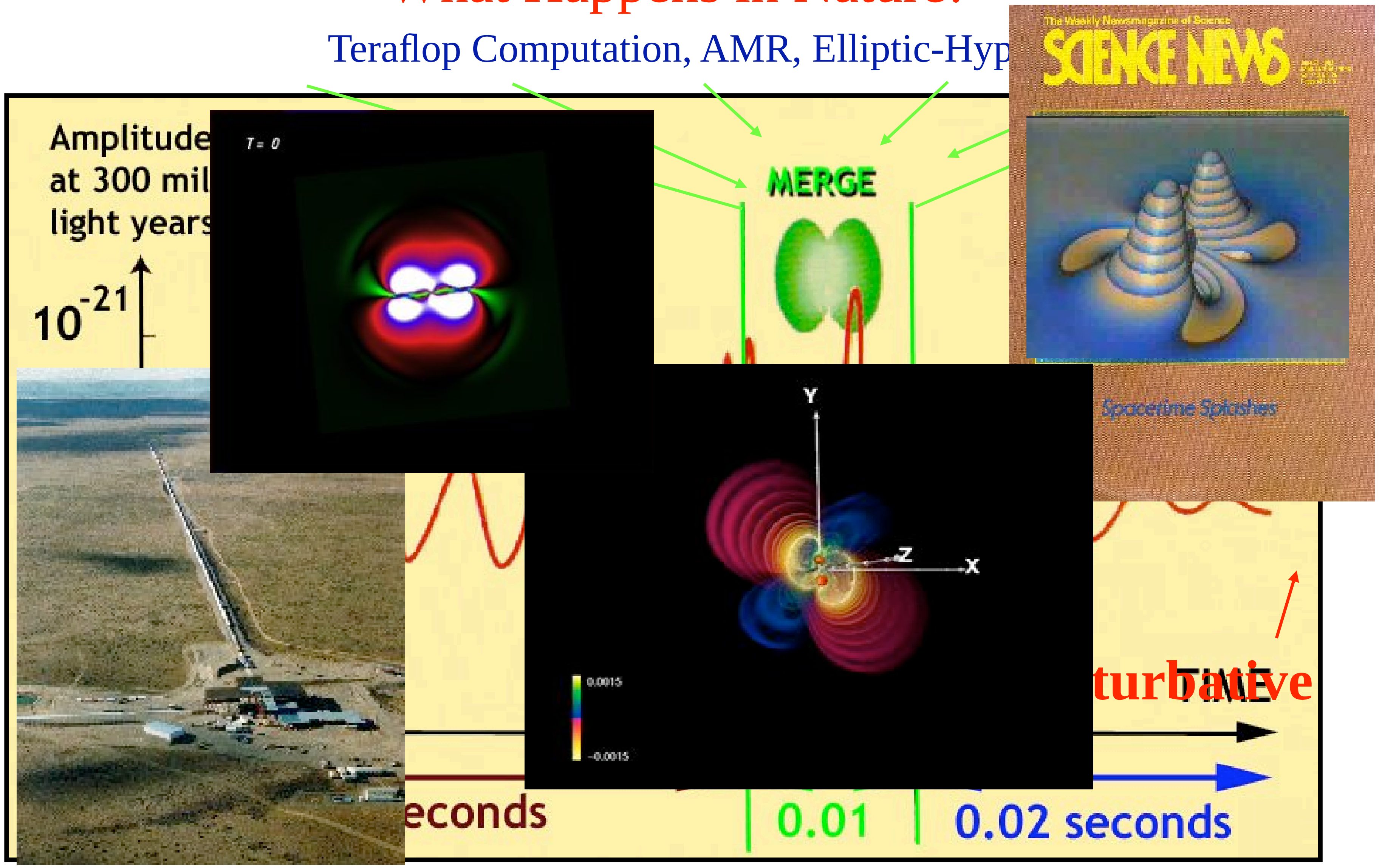
NSF STARTAP Lead-UIC's Elec. Vis. Lab

www.cactuscode.org

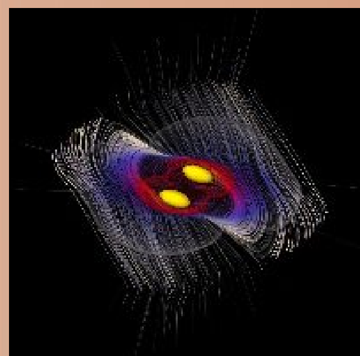
www.gridlab.org

High Capacity Computing: Want to Compute What Happens in Nature!

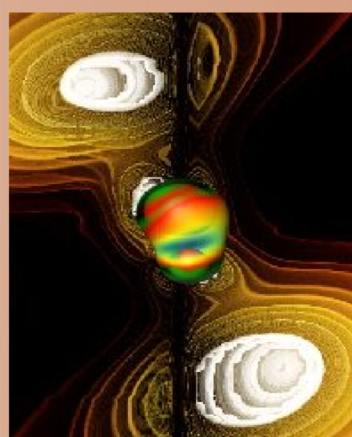
Teraflop Computation, AMR, Elliptic-Hyp



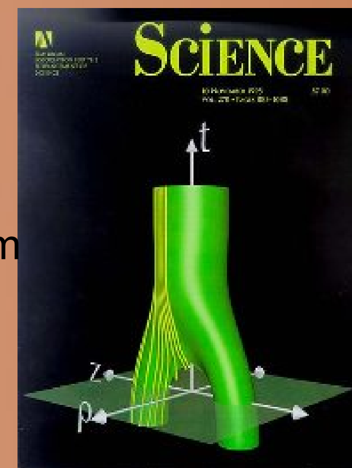
Grand Challenge Collaborations



NASA Neutron Star Grand Challenge
- 5 US Institutions
- Attack colliding neutron star problem



EU Astrophysics Network
- 10 EU Institutions
- 3 years
- Continue these problems

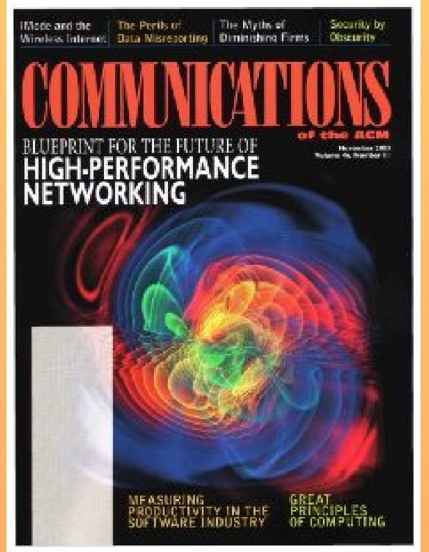


NSF Black Hole Grand Challenge
- 8 US Institutions
- 5 years
- Attack colliding black hole problem

Examples of Future of Science & Engineering
- Require Large Scale Simulations, beyond reach of any machine
- Require Large Geo-distributed Cross-Disciplinary Collaborations
- Require Grid Technologies, but not yet using them!

Numerical Relativity Applications

Study sources of Grav. Waves



- Desired BH, NS collisions/collapse
  - 10^20 Flops, multiple orbits, adaptive meshes, hydrodynamics, 1 day on 50TF machine
  - Real time scheduling across multiple resources for distributed computing
  - Lambda provisioning on demand to enable, spawning (for analysis, steering tasks), migration, interactive viz from distributed collaborations
- Output
  - Minimal 25-100TB for full analysis of waves, event horizon structure, hydrodynamics (multiple orbits)
  - May be physically distributed due to Grid activities above (spawning , migration, etc)

Issues for LIGO, GEO, LISA Gravitational Wave Astronomy

- Ensembles of simulations
  - Each simulation must be varied for accuracy, sensitivity analysis to gauge parameters, resolution, algorithms
  - Dozen simulations per physical scenario
- Parameter space!
  - Huge variations in physical scenarios for waveform catalogs
- In total: 10^3 - 10^6 simulations needed, potentially generating 25TB each, stored in federated repositories

Storage, Network Issues

- Simulation data stored across geographically distributed spaces
  - Organization, access, mining issues
- Need Lambda provisioning for
  - Grid computing scenarios
  - Remote viz and monitoring of simulations by distributed collaborations
  - Analysis of federated data sets by virtual organizations
- Data analysis of LIGO, GEO, LISA signals
  - Interacting with simulation data
  - Managing parameter space/signal analysis

Software/Toolkit Issues

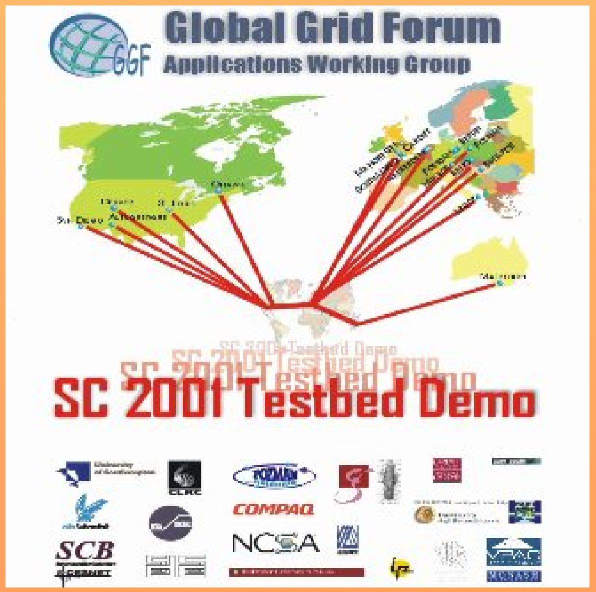
- Major problem to enable community
- Require toolkits for
  - Sharing/developing of community codes
  - AMR, remote viz, analysis
  - Logging, robust fault tolerance in grid scenarios, QoS, monitoring
  - Lambda provisioning
  - Data replication and management
- These problems are common to all!

Future view: much of it here now

- Scale of computations much larger: Grids!
- Complexity approaching that of Nature
  - Simulations of the Universe and its constituents
    - Black holes, neutron stars, supernovae
    - Human genome, human behavior
- Teams of researchers working together
  - Must support efficient, high level problem description
  - Must support collaborative computational science
  - Must support all different languages
- Ubiquitous Grid Computing
  - Resources, applications, replaced by abstraction: Grid Services
  - Very dynamic simulations, even deciding their own future
  - Apps may find services themselves: distributed, spawned, etc...
  - Must be tolerant of dynamic infrastructure
  - Monitored, viz'ed, controlled from anywhere, with colleagues elsewhere
- Applications: SCOOP, UCoMS, Hurricanes, others

Many Components for Grid Computing

- Resources: Apps Research Group
  - Worldwide “Virtual Organization” for Grid Computing
  - Over 2 dozen sites, 4 continents:
  - Alliance, NPACI, UK, Europe, Canada, Australia



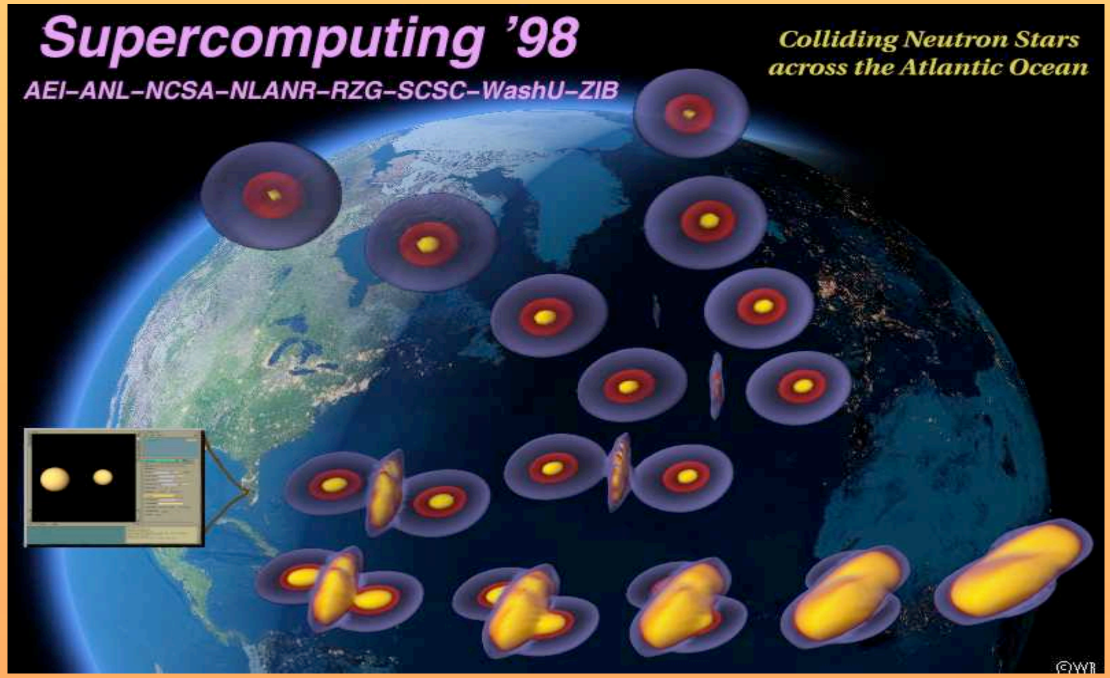
- Infrastructure: Globus, others
  - Develops fundamental technologies needed
  - to build computational grids.
  - Security: logins, data transfer
  - Communication
  - Information (GRIS, GIIS)
  - More generally: Grid Services
- Communities to use it! Virtual Organizations
  - Collaborations of groups with different expertise: no single group

A Missing Component for Grid Computing

- Grid Aware Applications
  - Most apps developed before Grid
  - Have no "awareness"
  - Take limited advantage
- Develop new generation of Grid Applications
  - Not just task farming/resource selection, totally new apps
  - Apps find Grid services, or even provide them
    - Ease of Use: automatically find resources, given need!
    - Distributed simulations: use as many machines as needed!
    - Remote Viz and Steering, tracking: watch what happens!
  - Grid Enabled Modular Toolkits/Libraries for Grid Computation: Provide to Scientist/Engineer
    - Plug your Science/Eng. Applications in!

Grid Applications so far...

- Typical scenario
  - Find remote resource
    - Task farm
    - Distribute
  - Launch job(s)
    - Static, possibly coupled
  - Visualize or collect results
  -
  -

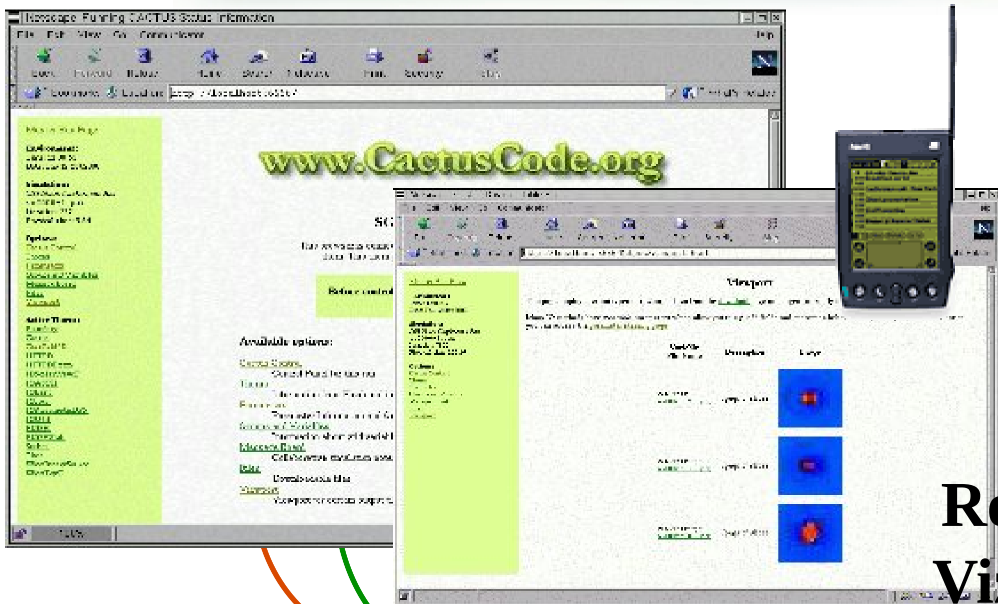


Metacomputing the Einstein Equations:
Connecting T3E's in Berlin, Garching, San Diego

- Can go far beyond this
  - Make it much, much easier
    - Portals, standards
  - Make it much more dynamic, adaptive, fault tolerant
  - Migrate this technology to general user

Remote Viz/Steering

watch/control simulation live…



GridLab

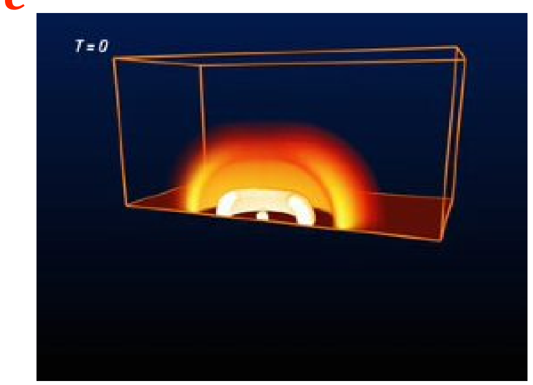
Any Viz Client:
LCA Vision, OpenDX

Remote Viz data

Changing any steerable parameter

- *Parameters*
- *Physics, algorithms*
- *Performance*

Streaming HDF5
Autodownsample



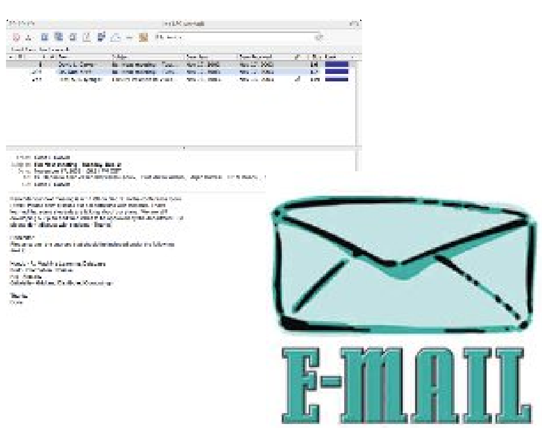
Remote Viz data

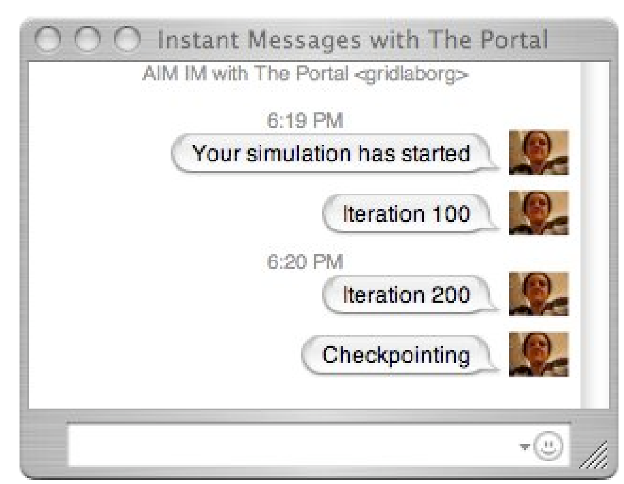
www.cactuscode.org

Amira

www.gridlab.org

Notification





Dynamic Distributed Computing

Want Apps to be able to Adapt to Changing Environment

- Many New Ideas
  - Consider: the Grid IS your computer
    - Networks, machines, devices come and go
    - Dynamic codes, aware of their environment, seek out resources
    - Distributed and Grid-based thread parallelism
  - Begin to change the way you think about problems: think global, solve much bigger problems
- Many old ideas
  - 1960's all over again
  - How to deal with dynamic processes
  - Processor management
  - Memory hierarchies, etc

GridLab

www.gridlab.org

- Resources, applications, etc replaced by abstract notion: Grid Services
- Grid is very dynamic, services come and go
  - Develop protection against this
  - Use to advantage: matches nature of many applications
- Very dynamic simulations, even deciding their own future
- Apps may find the services themselves: distributed, spawned, etc...
  - Must be tolerant of dynamic infrastructure
  - Monitored, viz'ed, controlled from anywhere, with colleagues elsewhere
- Developing tools/services to create this world
  - Grid Application Toolkit (GAT)
  - Need to become standardized

New Paradigms for Dynamic Grids

- Code/User/Infrastructure should be aware of environment
  - What Grid Sevices are available??
    - Resources available NOW, and what is their current state?
    - What is my allocation?
    - What is the bandwidth/latency between sites?
    - Where do I get my data from sensor network?
- Code/User/Infrastructure should be able to make decisions
  - A slow part of simulation runs asynchronously…spawn it off!
  - New, more powerful resources available…migrate there!
  - Machine went down…reconfigure and recover!
  - Need more memory (or less!): get it by adding(dropping) machines!
- Code/User/Infrastructure should be able to publish to central server for tracking, monitoring, steering…
  - Unexpected event…notify users!
  - Collaborators from anywhere all connect, examine simulation.

New Grid Applications

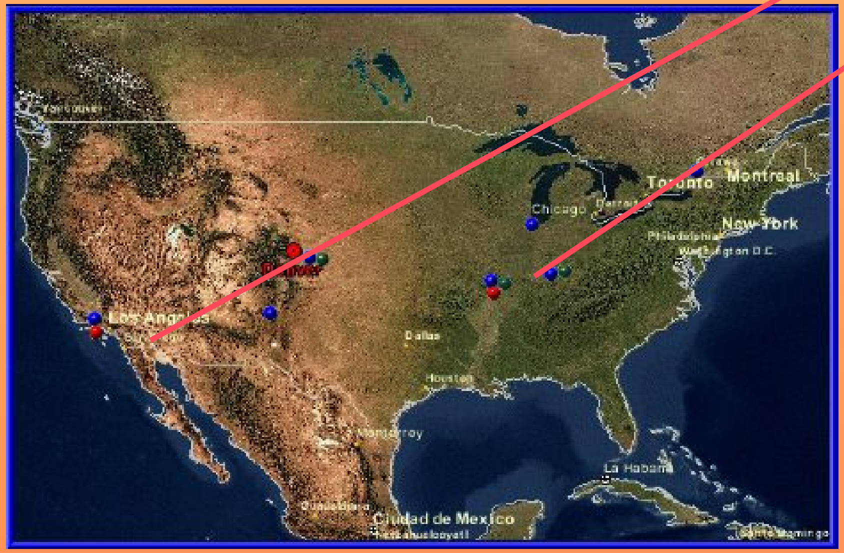
- Intelligent Parameter Surveys, Monte Carlos
  - May control other simulations!
- Dynamic Staging: move to faster/cheaper/ bigger machine ("Grid Worm")
- Multiple Universe: clone to investigate steered parameter ("Gird Virus")
- Automatic Component Loading
  - Needs change, discover/load/execute component
- Automatic Convergence Testing
- Look Ahead
  - Spawn off and run coarser resolution to predict likely future
- Spawn Independent/Asynchronous Tasks
- Dynamic Load Balancing: inhomogeneous loads, multiple grids
- Inject dynamically aqcuired data

What can be done now?

- Some Current Examples, work NOW as prototypes: Building blocks for the future
  - Dynamic, Adaptive Distributed Computing
    - Increase scaling from 15 - 70%
  - Migration: Cactus Worm
  - Spawning
  - Task Farm/Steering Combination
- If these can be done now, think what you could do tomorrow
- We are developing tools to enable such scenarios for any applications

Spawning on ARG Testbed

Main Cactus BH Simulation starts here

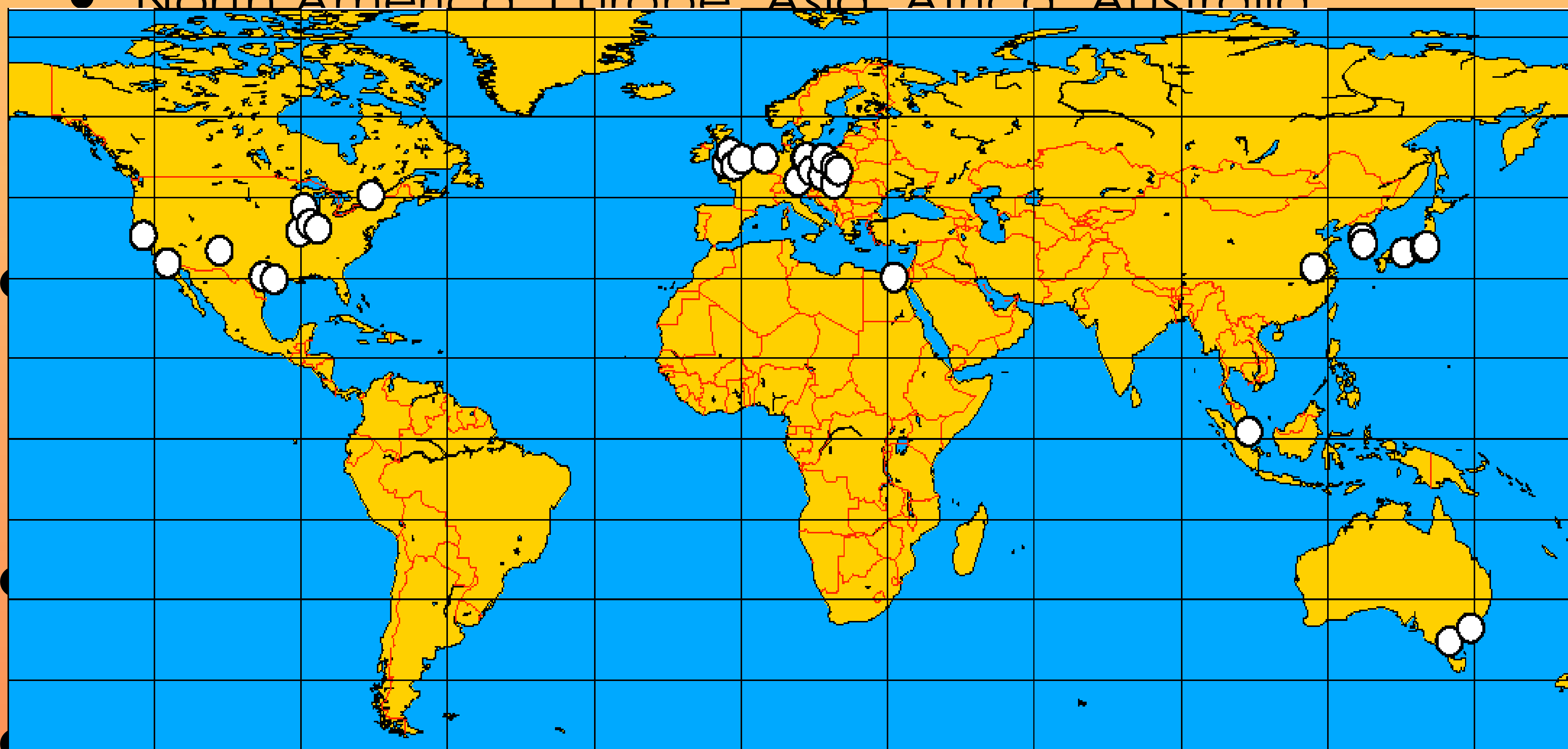


All analysis tasks spawned automatically to free resources worldwide

User only has to invoke "Spawner" thorn…

Global Grid Testbed Collaboration

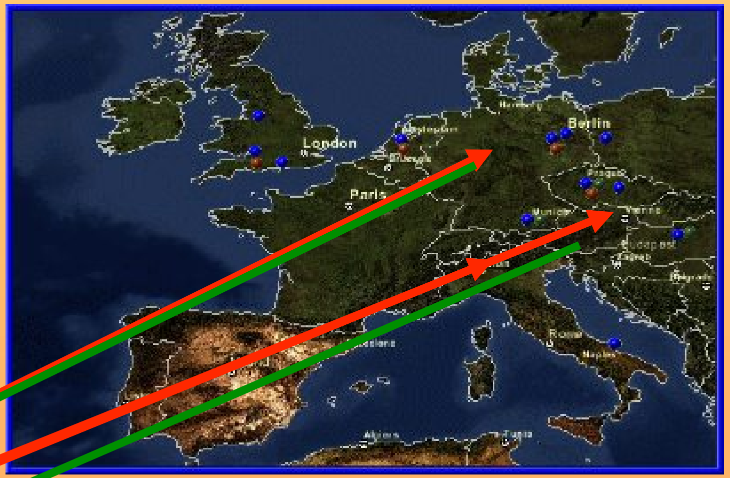
- 5 continents
  - North America, Europe, Asia, Africa, Australia

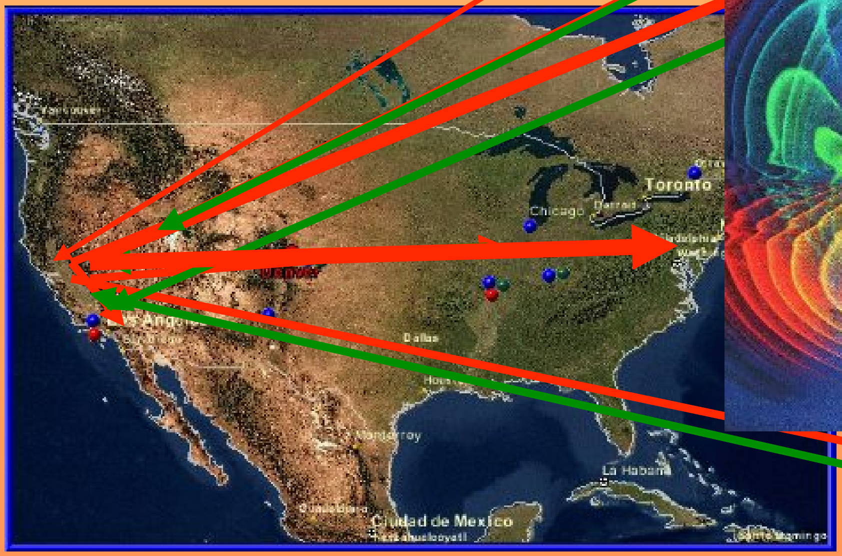


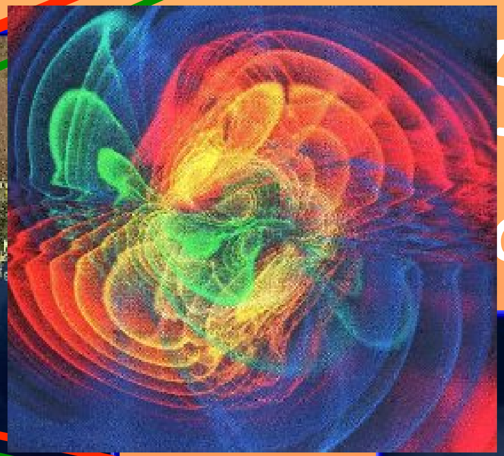
- And yet: all run same Grid infrastructure, and many can be used for applications…
- 4 different applications

Task Farm/Steering Combo

Main Cactus BH Simulation
starts in Berkeley







zens of smal. jobs
t out to test
ameters

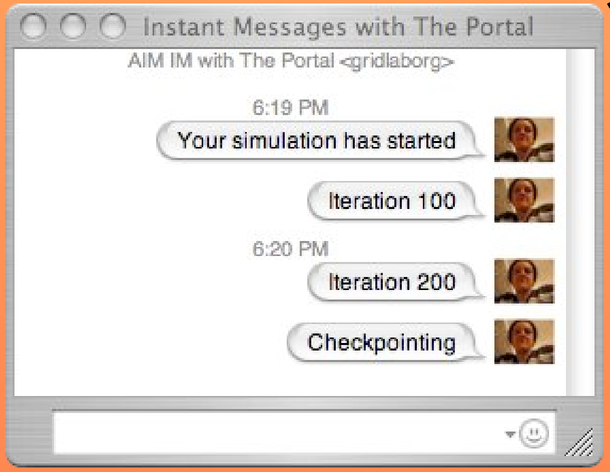
Data returned for main job

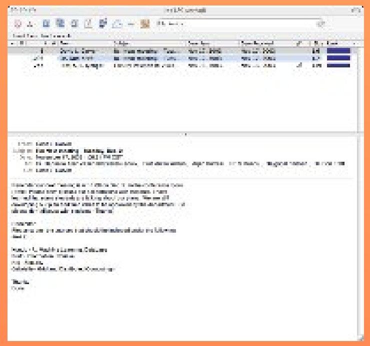
Huge job generates remote data
to be visualized in Baltimore

Notification

Local monitoring: Clients on machines provide information

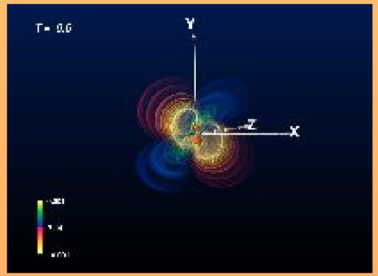
Portal stores users details and notification preferences

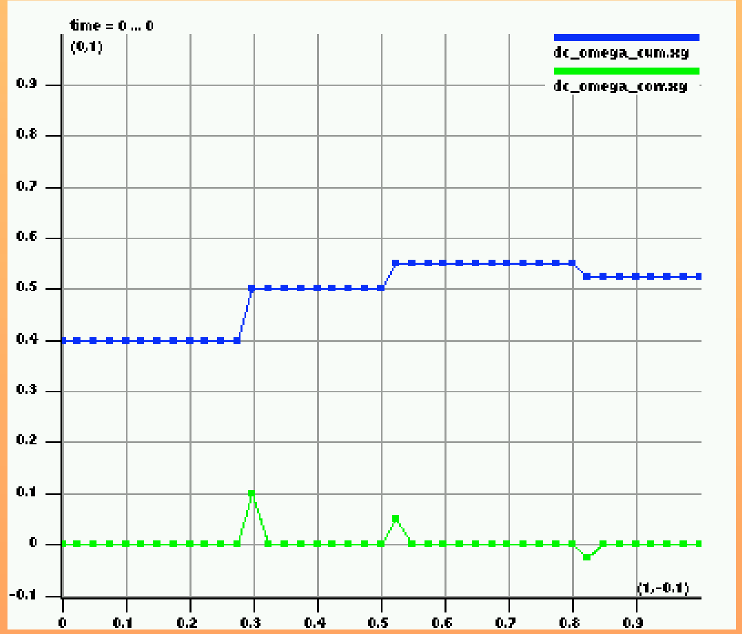


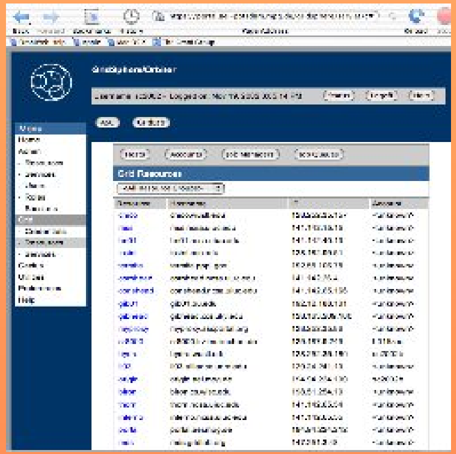


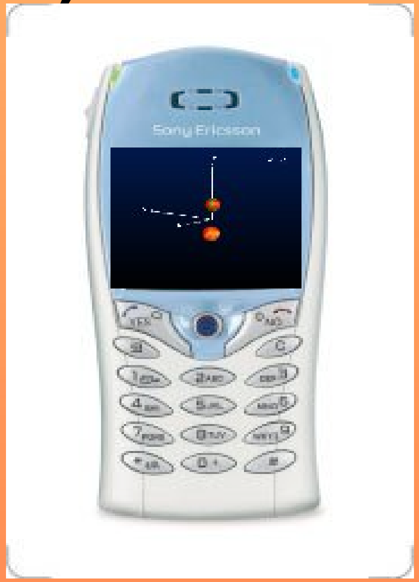
Task Farm/Remote Viz/ Steer Demo

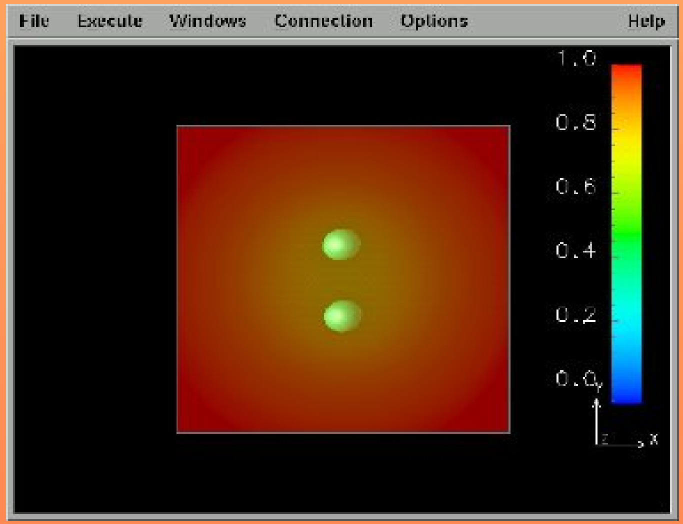
Big BH Sim
(LBL, NCSA, PSC, …)







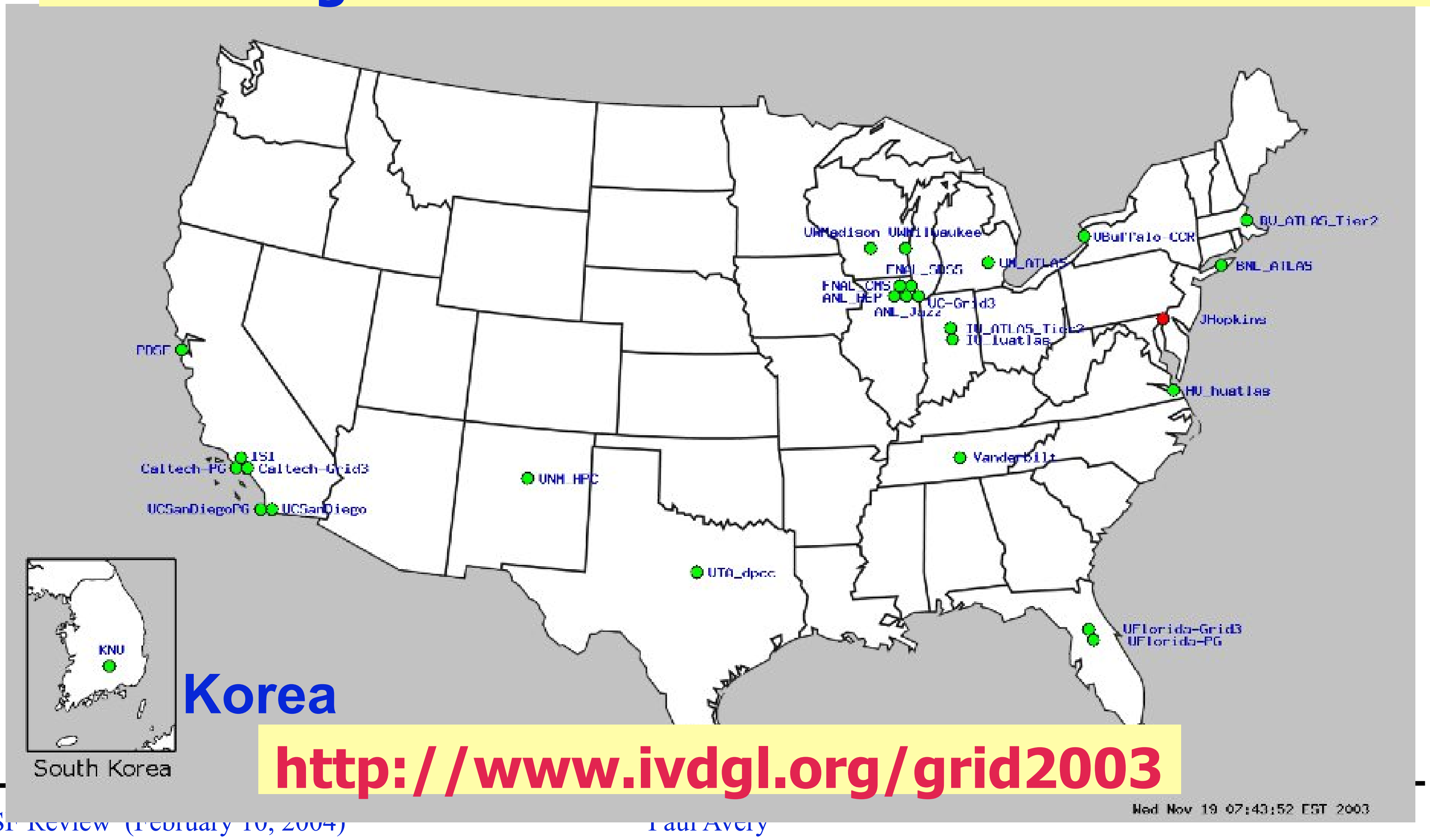




Task Farm Manager Status in portal…

Grid2003: An Operational Grid

- **28 sites (2100-2800 CPUs)**
- **400-1300 concurrent jobs**
- **10 applications**
- **Running since October 2003**



Korea

http://www.ivdgl.org/grid2003

Text

Grid2003: A Necessary Step

- Learning how to operate a Grid
  - Add sites, recover from errors, provide info, update, test, etc.
  - Need tools, services, procedures, documentation, organization
  - Need reliable, intelligent, skilled people
- Learning how to cope with "large" scale
  - "Interesting" failure modes as scale increases
  - Increasing scale must not overwhelm human resources
- Learning how to delegate responsibilities
  - Multiple levels: Project, Virtual Org., service, site, application
  - Essential for future growth

- **Grid2003 experience critical for building "useful" Grids**
- **Frank discussion in "Grid2003 Project Lessons" doc**

DØ Southern Analysis Region (DØSAR)

- Consortium coordinating activities to maximize computing and analysis resources
- UTA, OU, LTU, LU, SPRACE, Tata, KSU, KU, UMiss, CSF
- MC farm clusters - dedicated and multi-purpose, rack mounted and desktop, 10's -100's CPU's
- http://www-hep.uta.edu/d0-sar/d0-sar.html

ST4.1 Maxillo-facial Surgery Simulation

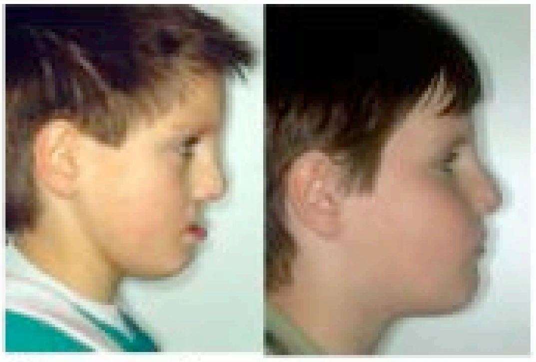
Distraction osteogenesis: new treatment for facial malformations.

NEC
Empowered by Innovation

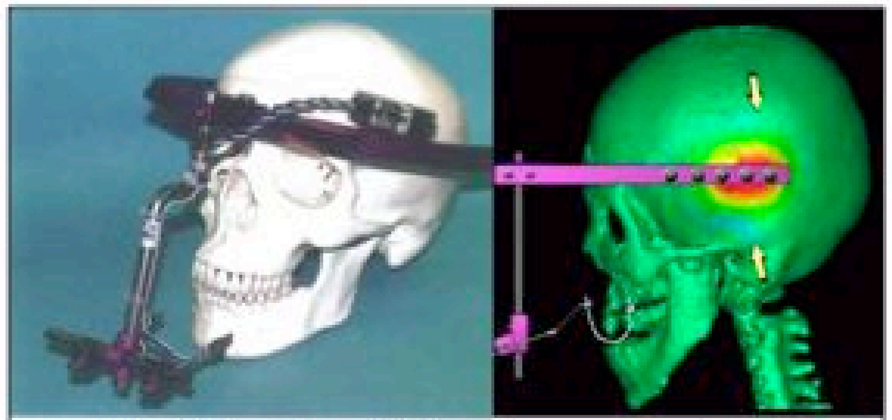
Provide a virtual try-out space for the pre-operative planning of maxillo-facial surgery.

Numerical modelling & simulation can improve surgical planning and predict outcome.

Requires large resources & parallel computers: distributed on-demand supercomputing.



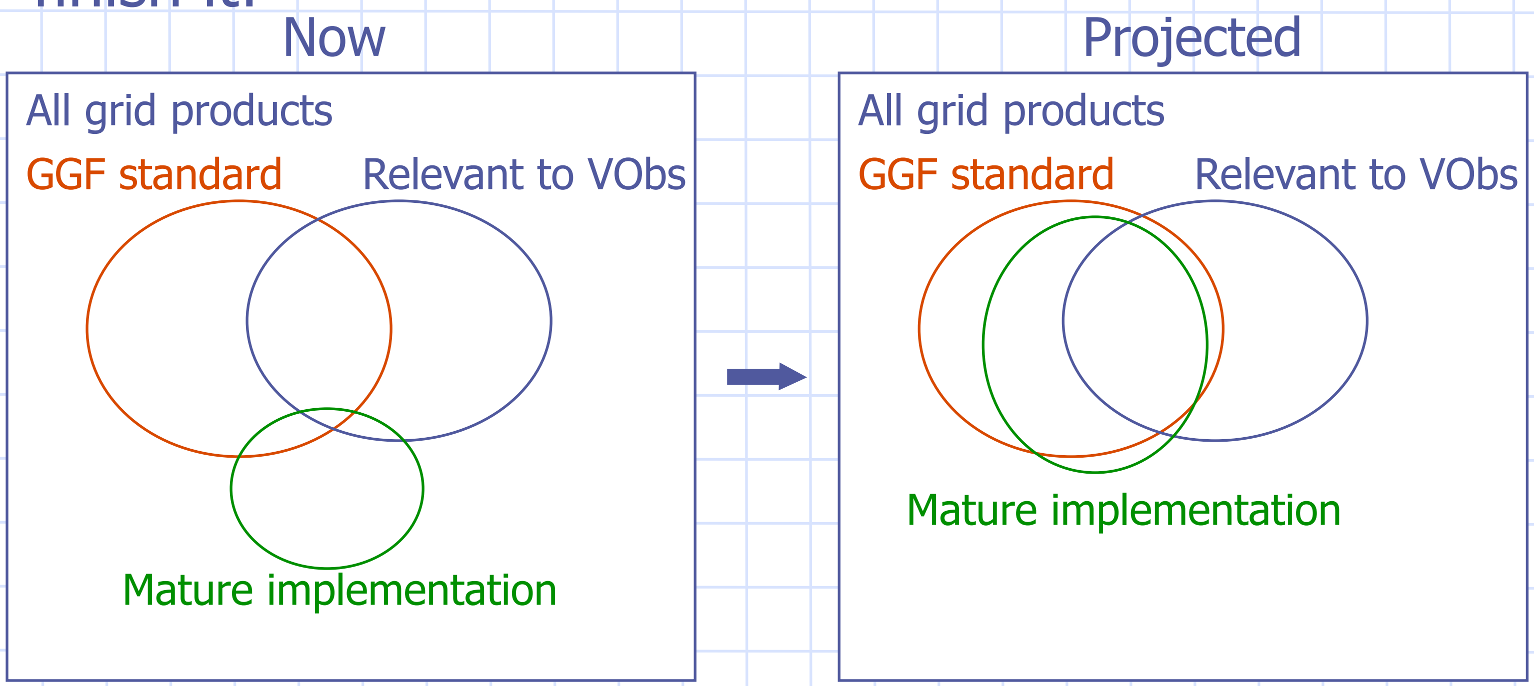
Pre- and post surgery
Courtesy Dr. Dr. Th. Hierl,
Clinic for Facial Surgery,
University Clinic Leipzig.



Modelling the distraction procedure

Confusion: GGF products, home grown products, other products…

- Few complete specifications
- Few mature, compliant implementations
- Need to wait for more finished product: you can help finish it!



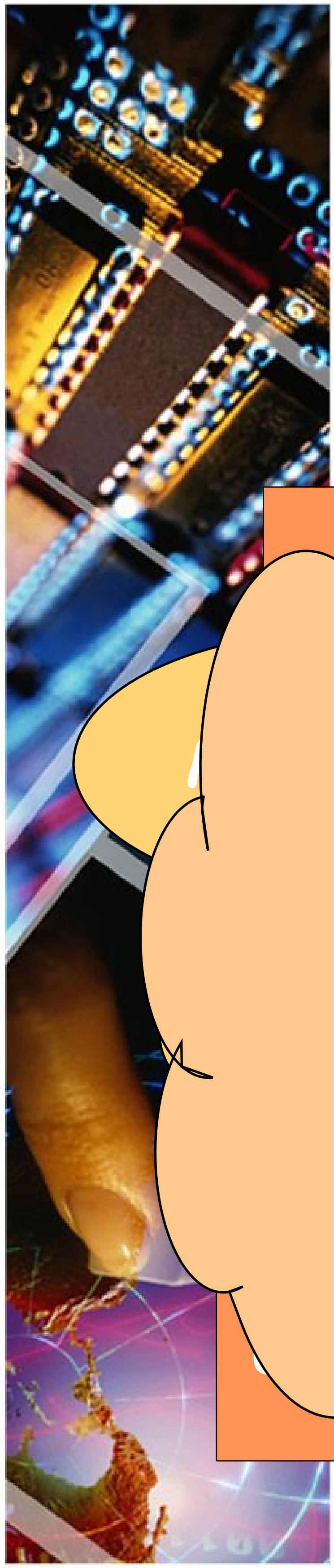
From Guy Rixon

TWG: "Grid Applications and Application Interfaces"

Peter Kacsuk, SZTAKI, Budapest
Jochen Fingberg, NEC Europe
Guy Lonsdale, NEC Europe
Ed Seidel, AEI, Potsdam and LSU, USA

- Goals
  - Define both common and specific requirements of applications for Grid environments in terms of
    - application classes and domains
    - particular Grid capabilities
- Issues
  - Grid Application Programming Interface (A P I)
  - Tools that would facilitate new application development,
  - Legacy application integration within Grids
  - More uniform and efficient development and deployment

How to build Grid Apps?



Application

"Is there a better resource I could be using?"

"Where can I get my data?"

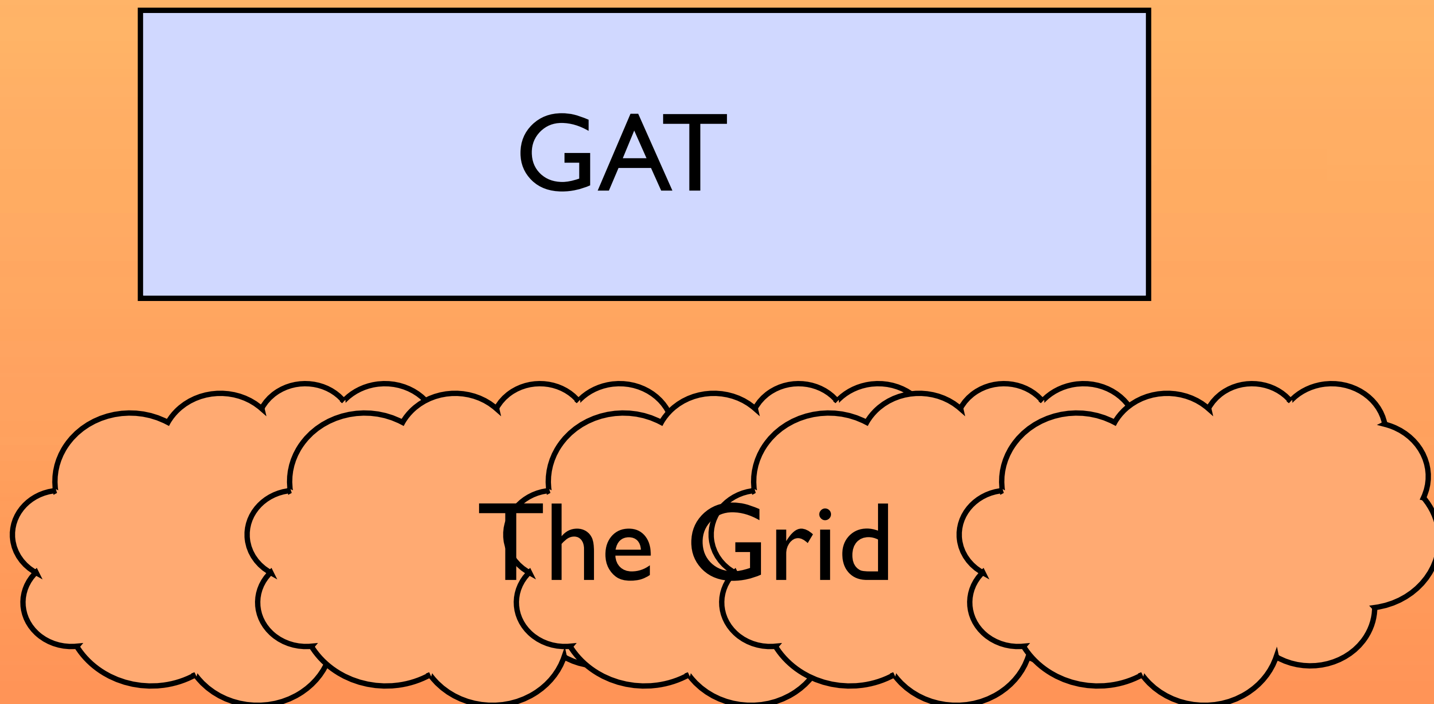
The Grid

Grid Application Toolkit

Application

“Is there a better resource I could be using?”

GAT_FindResource( )



Philosophy

- Application makes GAT API calls for operations which may be Grid-related.
- Application links against the GAT Engine
- Application runs irrespective of actual underlying infrastructure deployment
  - Engine loads adaptors which are valid in the environment extant when the application starts
  - Adaptors try to do Grid operations on request, on failure another adaptor provided function may be called.
- Application can thus be compiled, linked and tested without any Grid services
- Same application executable can run in a full Grid enviroment.

SAGA GGF Working Group

- Simple API for Grid Applications
- Aims to define standard APIs for commonly needed Grid operations
- Various implementations to be created
  - GridLab GAT with GAT Engine and adaptors
  - Others
- Leaders
  - Keith Jackson, LBNL
  - Tom Goodale, LSU
  - Stephen Pickles, Manchester

Summary

- Grids are coming
  - A lot of hype, promise, but definitely coming
- Will be critical for exploiting resources, integrating communities
- New applications must be created to take advantage
  - New paradigms, New Tools
  - Think creatively!
- We are in the early phases
  - Similar to parallel computing decade ago
  - MPI, HPF, etc ---> GAT, other Grid App development tools
- Need to move towards standards: SAGA, GAT: you can help!