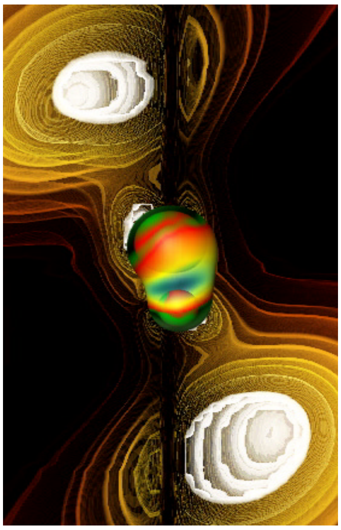
GridLab: Dynamic Grid Applications for Science and Engineering

A story from the difficult to the ridiculous…



Ed Seidel

Max-Planck-Institut für Gravitationsphysik (Albert Einstein Institute)

NCSA, U of Illinois

+

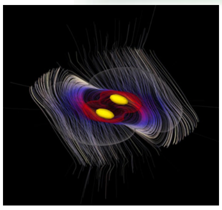
Lots of colleagues…

eseidel@ncsa.uiuc.edu

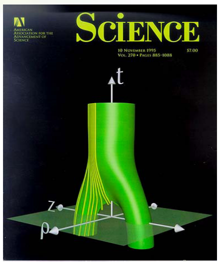
Co-Chair, GGF Applications Working Group

Grand Challenge Simulations

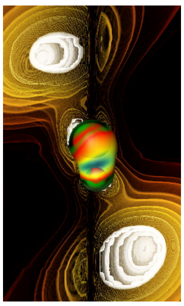
Science and Eng. Go Large Scale: Needs Dwarf Capabilities



- **NASA Neutron Star Grand Challenge**
  - 5 US Institutions
  - Solve problem of colliding neutron stars (try…)



- **NSF Black Hole Grand Challenge**
  - 8 US Institutions, 5 years
  - Solve problem of colliding black holes (try…)



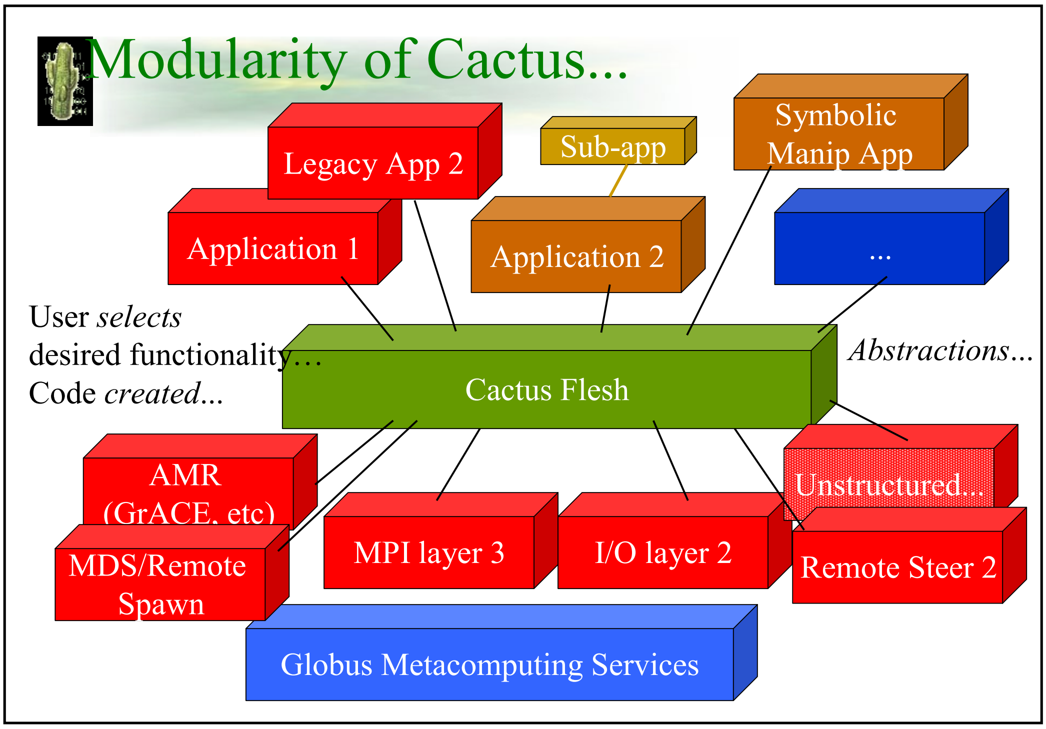
- **EU Network Astrophysics**
  - 10 EU Institutions, 3 years
  - Try to finish these problems …
  - *Entire Community becoming Grid enabled*
- **Examples of Future of Science & Engineering**
  - Require Large Scale Simulations, beyond reach of any machine
  - Require Large Geo-distributed Cross-Disciplinary Collaborations
  - Require Grid Technologies, but not yet using them!
  - Both Apps and Grids Dynamic…

Any Such Computation Requires Incredible Mix of Varied Technologies and Expertise!

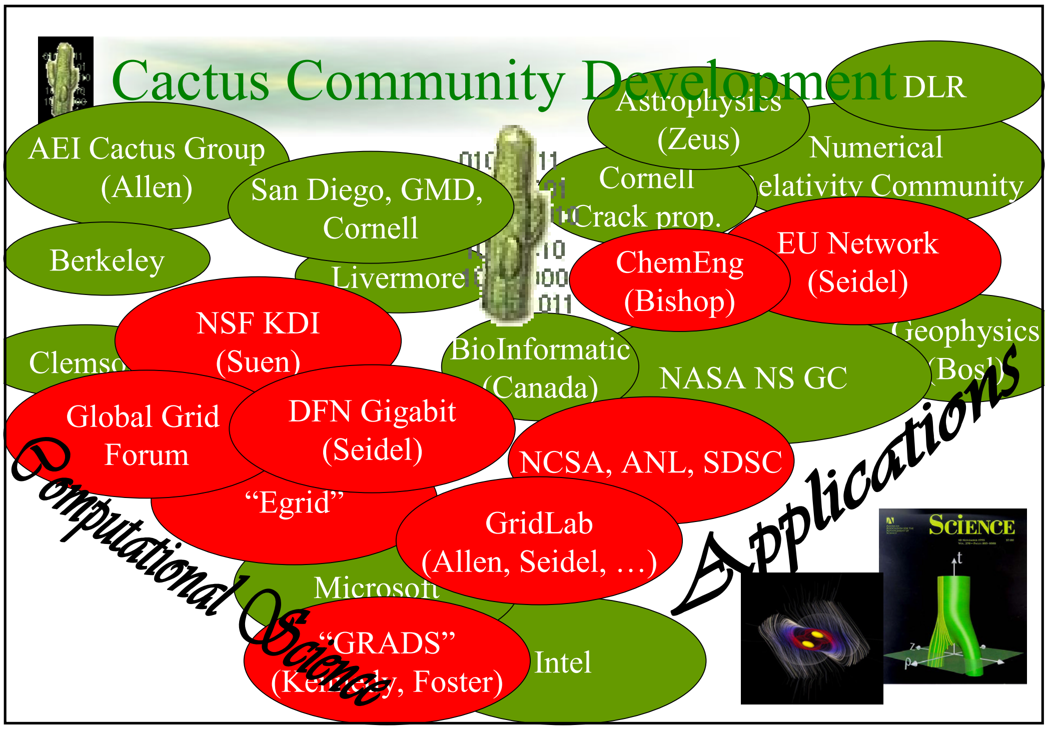
- Many Scientific/Engineering Components
  - Physics, astrophysics, CFD, engineering,...
- Many Numerical Algorithm Components
  - Finite difference methods?
  - Elliptic equations: multigrid, Krylov subspace, preconditioners,...
  - Mesh Refinement?
- Many Different Computational Components
  - Parallelism (HPF, MPI, PVM, ???)
  - Architecture Efficiency (MPP, DSM, Vector, PC Clusters, ???)
  - I/O Bottlenecks (generate gigabytes per simulation, checkpointing…)
  - Visualization of all that comes out!
- Scientist/eng. wants to focus on top, but *all* required for results...
- Such work cuts across many disciplines, areas of CS…

Cactus: ***community developed*** *simulation infrastructure*

- Developed as response to needs of large scale projects
- Numerical/computational infrastructure to solve PDE's
- Freely available, *Open Source* community framework: spirit of gnu/linux
  - Many communities contributing to Cactus
  - Cactus Divided in "Flesh" (core) and "Thorns" (modules or collections of subroutines)
  - Multilingual: User apps Fortran, C, C^{++}; automated interface between them
- Abstraction: Cactus Flesh provides API for virtually all CS type operations
  - Storage, parallelization, communication between processors, etc
  - Interpolation, Reduction
  - IO (traditional, socket based, remote viz and steering…)
  - Checkpointing, coordinates
- "Grid Computing": Cactus team and many collaborators worldwide, especially NCSA, Argonne/Chicago, LBL.



Modularity of Cactus...
Legacy App 2
Sub-app
Symbolic Manip App
Application 1
Application 2
...
User *selects* desired functionality…
Code *created*...
Cactus Flesh
Abstractions...
AMR (GrACE, etc)
MDS/Remote Spawn
MPI layer 3
I/O layer 2
Unstructured...
Remote Steer 2
Globus Metacomputing Services



Cactus Community Development
DLR
Astrophysics (Zeus)
AEI Cactus Group (Allen)
San Diego, GMD, Cornell
Cornell Crack prop.
Numerical Relativity Community
Berkeley
Livermore
EU Network (Seidel)
ChemEng (Bishop)
NSF KDI (Suen)
Clemson
BioInformatic (Canada)
NASA NS GC
Geophysics (Bosl)
Global Grid Forum
DFN Gigabit (Seidel)
NCSA, ANL, SDSC
“Egrid”
GridLab (Allen, Seidel, …)
Applications
Computational Science
Microsoft
“GRADS” (Kennedy, Foster)
Intel
SCIENCE

Future view: much of it here already...

- Scale of computations much larger
  - Complexity approaching that of Nature
  - Simulations of the Universe and its constituents
    - Black holes, neutron stars, supernovae
    - Human genome, human behavior
- Teams of computational scientists working together
  - Must support efficient, high level problem description
  - Must support collaborative computational science
  - Must support all different languages
- Ubiquitous Grid Computing
  - Very dynamic simulations, deciding their own future
  - Apps find the resources themselves: distributed, spawned, etc...
  - Must be tolerant of dynamic infrastructure (variable networks, processor availability, etc…)
  - Monitored, viz'ed, controlled from anywhere, with colleagues elsewhere

Grid Simulations: a new paradigm

- *Computational Resources Scattered Across the World*
  - Compute servers
  - Handhelds
  - File servers
  - Networks
  - Playstations, cell phones etc…

QuickTime™ and a
Photo - JPEG decompressor
are needed to see this picture.

- *How to take advantage of this for scientific/engineering simulations*?
  - Harness multiple sites and devices
  - Simulations at new level of complexity and scale

QuickTime™ and a
Photo decompressor
are needed to see this picture.

Many Components for Grid Computing:
all have to work for real applications

1. Resources: Egrid (www.egrid.org)
   - A "Virtual Organization" in Europe for Grid Computing
   - Over a dozen sites across Europe
   - Many different machines

QuickTime™ and a
Photo - JPEG decompressor
are needed to see this picture.

2. Infrastructure: Globus Metacomputing Toolkit (Example)
   - Develops fundamental technologies needed to build computational grids.
   - Security: logins, data transfer
   - Communication
   - Information (GRIS, GIIS)

Components for Grid Computing, cont.

3. Grid Aware Applications (Cactus example):
   - Grid Enabled Modular Toolkits for Parallel Computation: Provide to Scientist/Engineer…
   - Plug your Science/Eng. Applications in!
   - Must Provide Many Grid Services
     - Ease of Use: automatically find resources, given need!
     - Distributed simulations: use as many machines as needed!
     - Remote Viz and Steering, tracking: watch what happens!
   - Collaborations of groups with different expertise: no single group can do it! Grid is natural for this…

Egrid Testbed

- Many sites, heterogeneous
  - MPI-Gravitationsphysik,
  - Konrad-Zuse-Zentrum,
  - Poznan,
  - Lecce, Vrije Universiteit-Amsterdam,
  - Paderborn,
  - Brno,
  - MTA-Sztaki-Budapest,
  - DLR-Köln,
  - GMD-St. Augustin
  - ANL, ISI, & friends
- In 12 weeks, all sites had formed a Virtual Organization with
  - Globus 1.1.4
  - MPICH-G2
  - GSI-SSH
  - GSI-FTP
- Central GIIS's at Poznan, Lecce
- Key Application: Cactus
- Egrid merged with Grid Forum to form GGF, but maintains Egrid testbed, identity

Cactus & the Grid

Cactus Application Thorns
Distribution information hidden from programmer
Initial data, Evolution, Analysis, etc

Grid Aware Application Thorns
Drivers for parallelism, IO, communication, data mapping
PUGH: parallelism via MPI
(MPICH-G2, grid enabled message passing library)

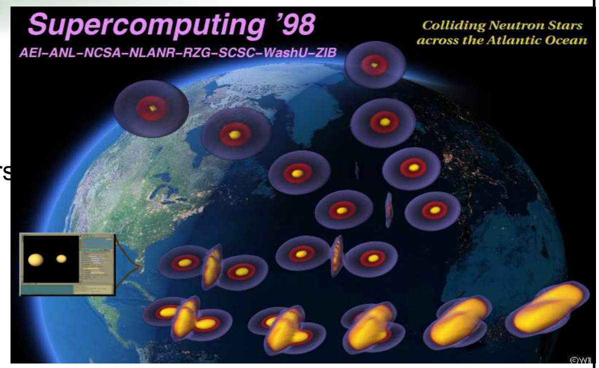
Single Proc

Standard MPI

Grid Enabled Communication Library
MPICH-G2 implementation of MPI, can run MPI programs across heterogenous computing resources

Grid Applications so far...

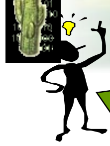
- SC93 - SC2000
- Typical scenario
  - Find remote resource
    (often using multiple computers)
  - Launch job
    (usually static, tightly coupled)
  - Visualize results
    (usually in-line, fixed)



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

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

The Astrophysical Simulation Collaboratory (ASC)



1. User has *science* idea...

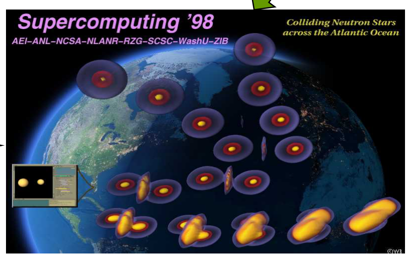
2. *Composes/Builds* Code Components w/Interface...

Cactus Computational Toolkit

Science, Autopilot, AMR, Petsc, HDF, MPI, GrACE, Globus, Remote Steering...

3. *Selects* Appropriate Resources...

4. *Steers* simulation, monitors performance...



5. *Collaborators log in* to monitor...

Want to integrate and migrate this technology to the generic user…

Supercomputing super difficult

Consider simplest case: sit here, compute there

Accounts for one AEI user (real case):

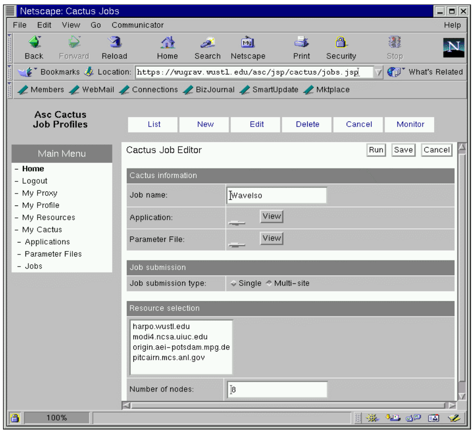
- berte.zib.de
- denali.mcs.anl.gov
- golden.sdsc.edu
- gseaborg.nersc.gov
- harpo.wustl.edu
- horizon.npaci.edu
- loslobos.alliance.unm.edu
- mcurie.nersc.gov
- modi4.ncsa.uiuc.edu
- ntsc1.ncsa.uiuc.edu
- origin.aei-potsdam.mpg.de
- pc.rzg.mpg.de
- pitcairn.mcs.anl.gov
- quad.mcs.anl.gov
- rr.alliance.unm.edu
- sr8000.lrz-muenchen.de

16 machines, 6 different usernames, 16 passwords, ...

This is hard, but it gets much worse from here…

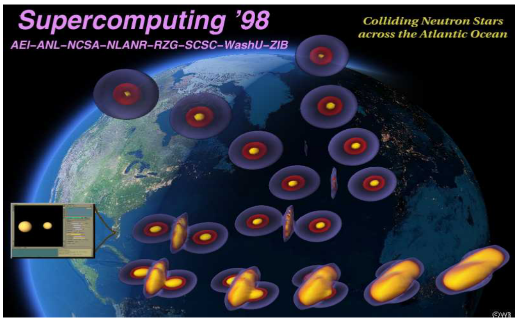
ASC Portal (Russell, Daues, Wind^2, Bondarescu, Shalf, et al)

- ASC Project
  - Code management
  - Resource selection (including distributed runs
  - Code Staging, Sharing
  - Data Archiving, Monitoring, etc…
- Technology: Globus, GSI, Java, DHTML, MyProxy, GPDK, TomCat, Stronghold
- Used for the ASC Grid Testbed (SDSC, NCSA, Argonne, ZIB, LRZ, AEI)
- Driven by the need for easy access to machines
- Useful tool to test Alliance VMR!!

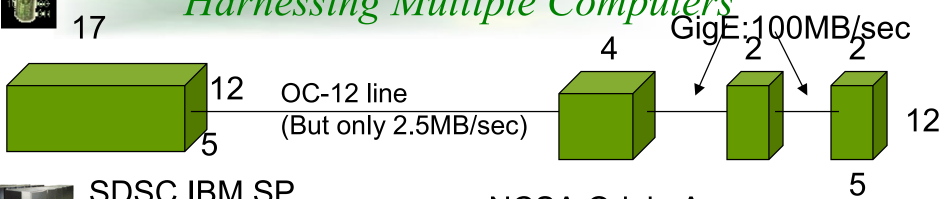


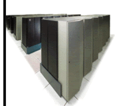
Distributed Computation:
Harnessing Multiple Computers

- Why would anyone want to do this?
  - Capacity
  - Throughput
- Issues
  - Bandwidth
  - Latency
  - Communication needs
  - Topology
  - Communication/computation
- Techniques to be developed
  - Overlapping comm/comp
  - Extra ghost zones
  - Compression
  - Algorithms to do this for the scientist…
- Experiments
  - 3 T3Es on 2 continents
  - Last month: joint NCSA, SDSC test with 1500 processors (Dramlitsch talk)



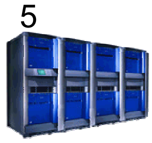
Distributed Computation:
Harnessing Multiple Computers





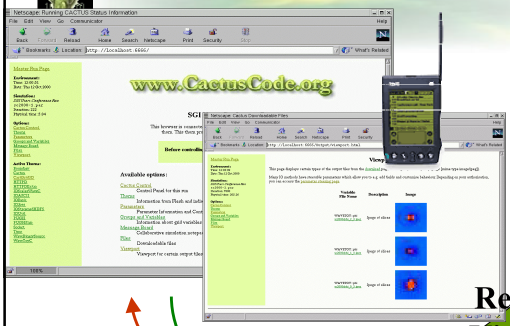
SDSC IBM SP
1024 procs
5x12x17 =1020

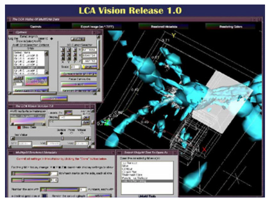
NCSA Origin Array
256+128+128
5x12x(4+2+2) =480



- Why would anyone want to do this?
  - Capacity, Throughput
- Solving Einstein Equations, but could be any application
  - 70-85% scaling, ~250GF (only 15% scaling without tricks)
- Techniques to be developed
  - Overlapping comm/comp, Extra ghost zones
  - Compression
  - ***Adaption!!***
  - Algorithms to do this for the scientist…

Remote Viz/Steering *watch/control simulation live…*





Any Viz Client:
LCA Vision, OpenDX

Changing any steerable parameter
- *Parameters*
- *Physics, algorithms*
- *Performance*

Remote
Viz data

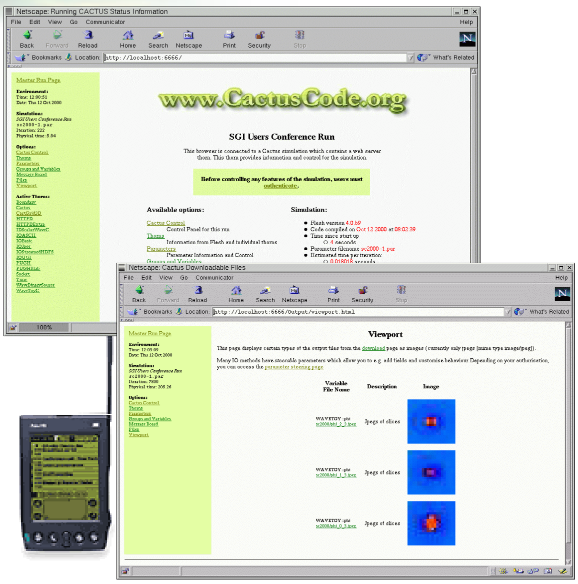
Streaming HDF5
Autodownsample

Remote
Viz data

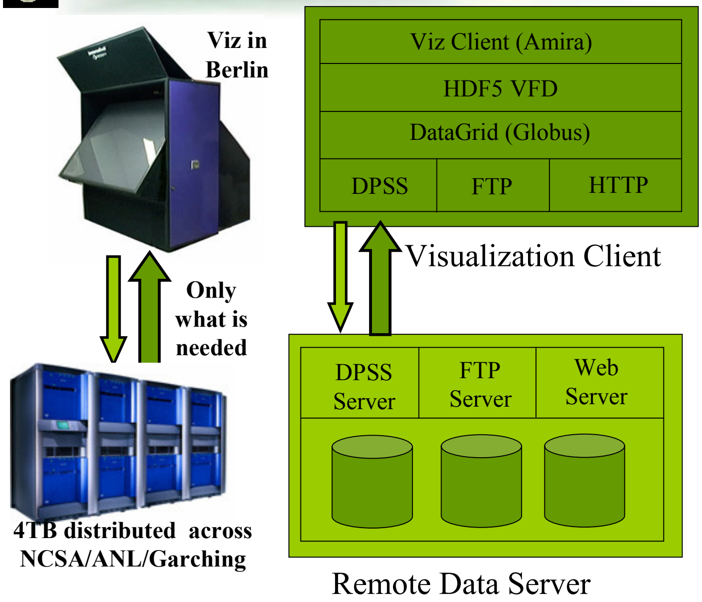
Amira

Thorn HTTPD

- Thorn which allows simulation any to act as its own web server
- Connect to simulation from any browser anywhere
- Monitor run: parameters, basic visualization, ...
- Change *steerable* parameters
- See running example at www.CactusCode.org
- Wireless remote viz, monitoring and steering



Remote Offline Visualization



- Accessing remote data for local visualization
- Should allow downsampling, hyperslabbing, etc.
- Grid World: file pieces left all over the world, but logically one file…

Dynamic Distributed Computing

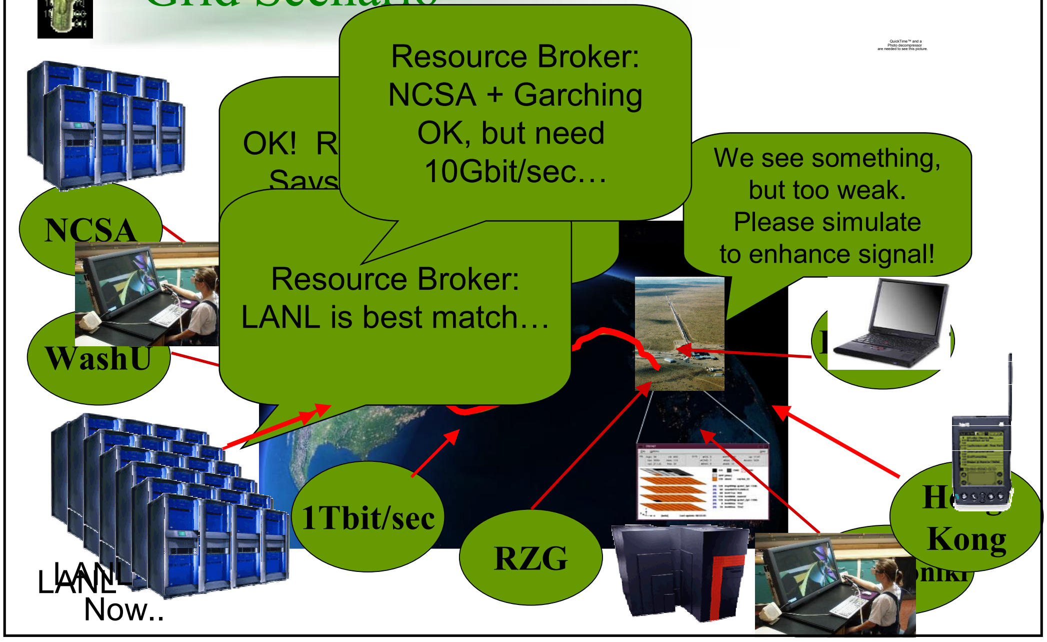
Static grid model works only in special cases; must make apps able to respond to changing Grid environment...

- Many new ideas
  - Consider: the Grid IS your computer:
    - Networks, machines, devices come and go
    - Dynamic codes, aware of their environment, seeking out resources
    - Rethink algorithms of all types
    - Distributed and Grid-based thread parallelism
  - Scientists and engineers will change the way they think about their 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: New Paradigms for Dynamic Grids

- Code should be aware of its environment
  - What resources are out there NOW, and what is their current state?
  - What is my allocation?
  - What is the bandwidth/latency between sites?
- Code should be able to make decisions on its own
  - A slow part of my simulation can run asynchronously…spawn it off!
  - New, more powerful resources just became available…migrate there!
  - Machine went down…reconfigure and recover!
  - Need more memory…get it by adding more machines!
- Code should be able to publish this information to central server for tracking, monitoring, steering…
  - Unexpected event…notify users!
  - Collaborators from around the world all connect, examine simulation.

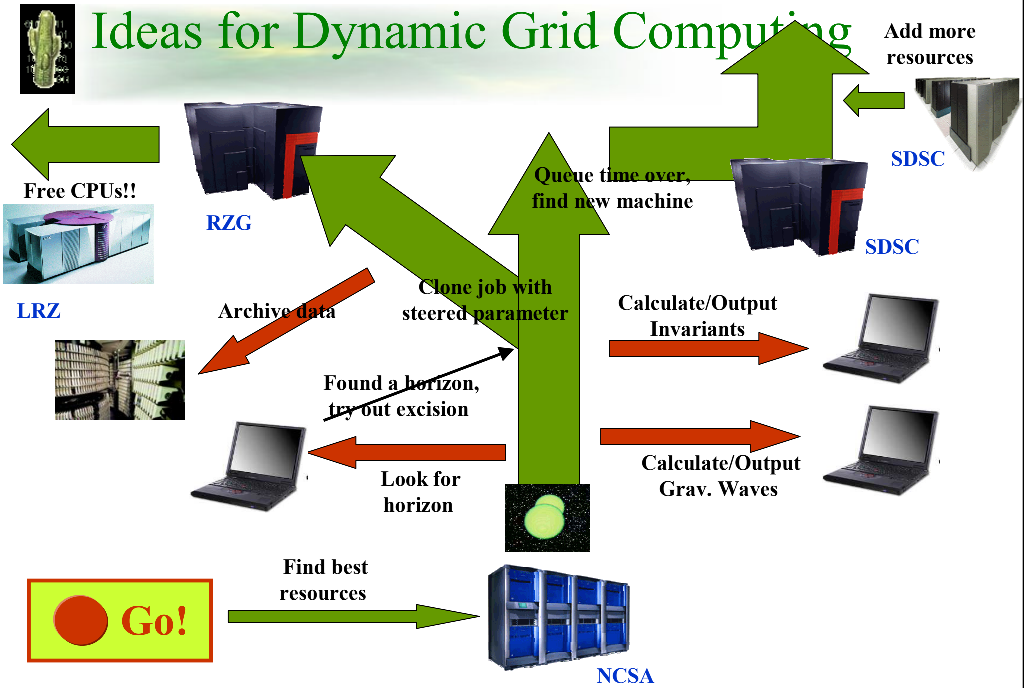
Grid Scenario



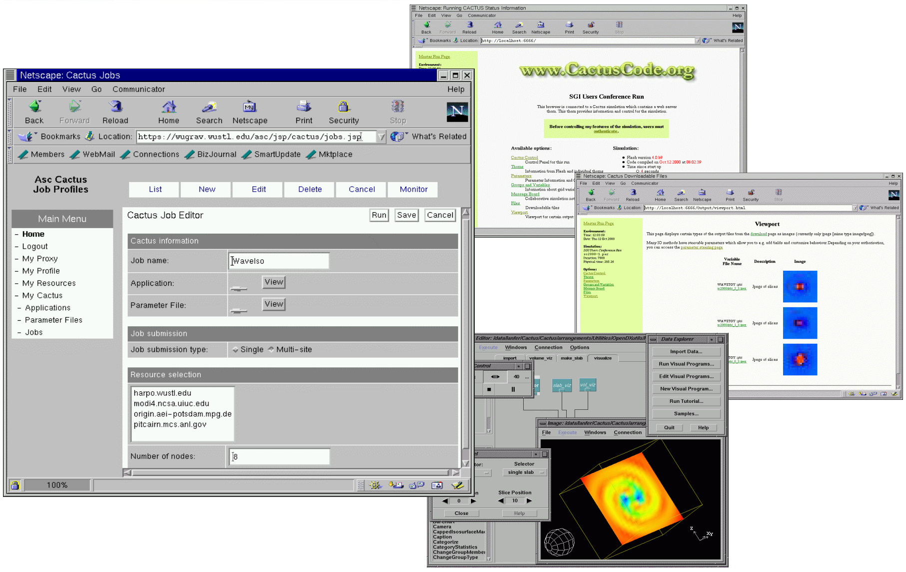
New Grid Applications: some examples

- Dynamic Staging: move to faster/cheaper/bigger machine
  - "Cactus Worm"
- Multiple Universe
  - create clone to investigate steered parameter ("Cactus Virus")
- Automatic Convergence Testing
  - from intitial data or initiated during simulation
- Look Ahead
  - spawn off and run coarser resolution to predict likely future
- Spawn Independent/Asynchronous Tasks
  - send to cheaper machine, main simulation carries on
- Thorn Profiling
  - best machine/queue, choose resolution parameters based on queue
- Dynamic Load Balancing
  - inhomogeneous loads, multiple grids
- Intelligent Parameter Surveys
  - farm out to different machines
- …Must get application community to rethink algorithms…

Ideas for Dynamic Grid Computing



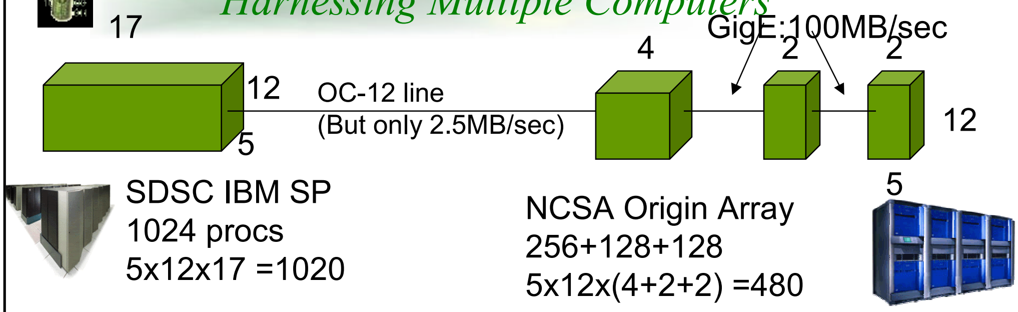
User's View … *simple!*



Issues Raised by Grid Scenarios

- Infrastructure:
  - Is it ubiquitous? Is it reliable? Does it work?
- Security:
  - How does user pass proxy from site to site?
  - Firewalls? Ports?
- How does user/application get information about Grid?
  - Need reliable, ubiquitous Grid information services
  - Portal, Cell phone, PDA
- What is a file? Where does it live?
  - Crazy Grid apps will leave pieces of files all over the world
- Tracking
  - How does user track the Grid simulation hierarchies?
- Two Current Examples that work Now: Building blocks for the future
  - Dynamic, Adaptive Distributed Computing
  - Migration: Cactus Worm

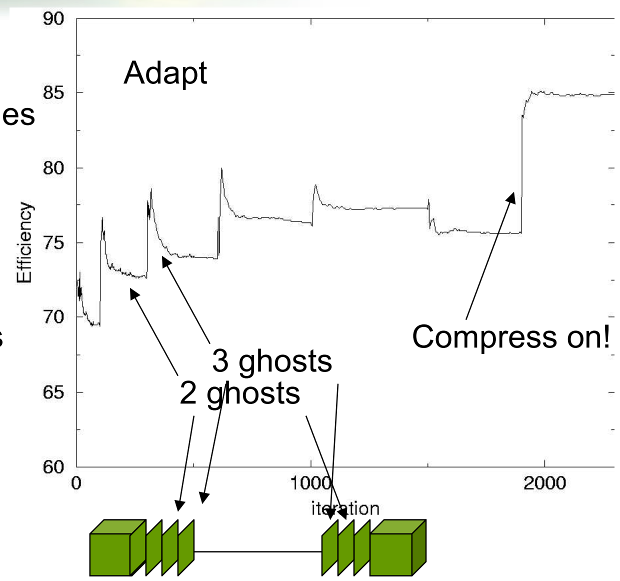
Distributed Computation:
Harnessing Multiple Computers



- Solving Einstein Equations, but could be any application
  - 70-85% scaling, ~250GF (only 15% scaling without tricks)
- Techniques to be developed
  - Overlapping comm/comp, Extra ghost zones
  - Compression
  - ***Adaption!!***
  - Algorithms to do this for the scientist…

Dynamic Adaptation in Distributed Computing

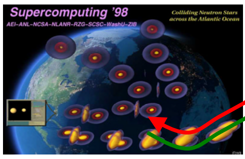
- Automatically adapt to bandwidth latency issues
- Application has NO KNOWLEDGE of machines(s) it is on, networks, etc
- Adaptive techniques make NO assumptions about network
- Issues: if network conditions change faster than adaption…

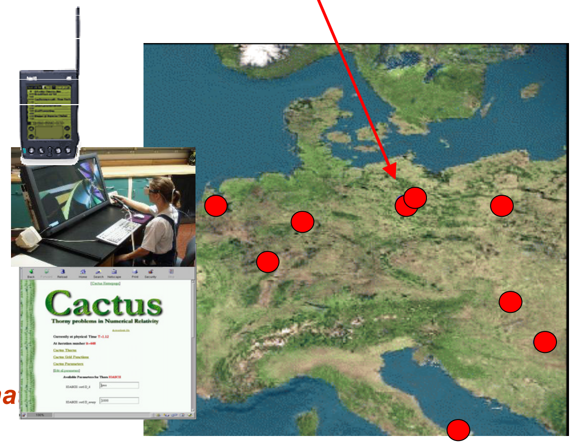


Cactus Worm: Illustration of basic scenario

Live demo at http://www.CactusCode.org (usually)

- Cactus simulation (could be anything) starts, launched from a portal
- Queries a Grid Information Server, finds available resources
- Migrates itself to next site, according to some criterion
- Registers new location to GIS, terminates old simulation
- User tracks/steers, using http, streaming data, etc……





- Continues around Europe…
- ***If we can do this, much of wha we want can be done!***

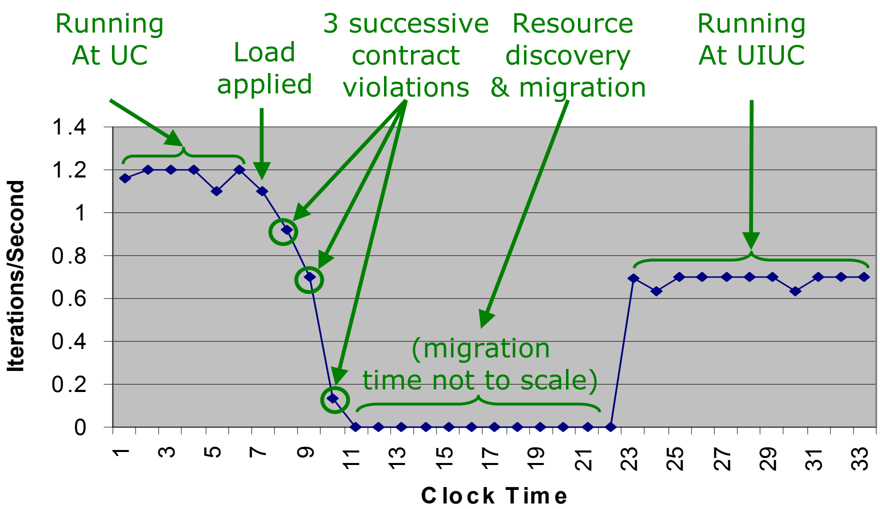
Worm as a building block for dynamic Grid applications: many uses

- Tool to test operation of Grid: Alliance VMR, Egrid, other testbeds
- Will be outfitted with diagnostics, performance tools
  - What went wrong where?
  - How long did a given Worm “payload” take to migrate
  - Are grid map files in order?
  - Certificates, etc…
- Basic technology for migrating
  - Entire simulations
  - Parts of simulations
- Example: contract violation…
  - Code going too slow, too fast, using too much memory, etc…

How to determine when to migrate: Contract Monitor

- GrADS project activity: Foster, Angulo, Cactus team
- Establish a "Contract"
  - Driven by user-controllable parameters
    - Time quantum for "time per iteration"
    - % degradation in time per iteration (relative to prior average) before noting violation
    - Number of violations before migration
- Potential causes of violation
  - Competing load on CPU
  - Computation requires more processing power: e.g., mesh refinement, new subcomputation
  - Hardware problems
  - Going too fast! Using too little memory? Why waste a resource??

Migration due to Contract Violation (Foster, Angulo, Cactus Team…)

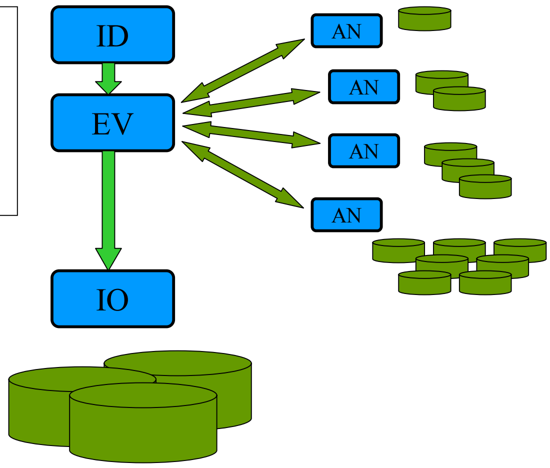


Grid Application Development Toolkit

- Application developer should be able to build simulations with tools that easily enable dynamic grid capabilities
- Want to build programming API to easily allow:
  - Query information server (e.g. GIIS)
    - What's available for me? What software? How many processors?
  - Network Monitoring
  - Decision Routines (Thorns)
    - How to decide? Cost? Reliability? Size?
  - Spawning Routines (Thorns)
    - Now start this up over here, and that up over there
  - Authentication Server
    - Issues commands, moves files on your behalf (can't pass-on Globus proxy)
  - Data Transfer
    - Use whatever method is desired (Gsi-ssh, Gsi-ftp, Streamed HDF5, scp…)
  - Etc…

Example Toolkit Call: Routine Spawning

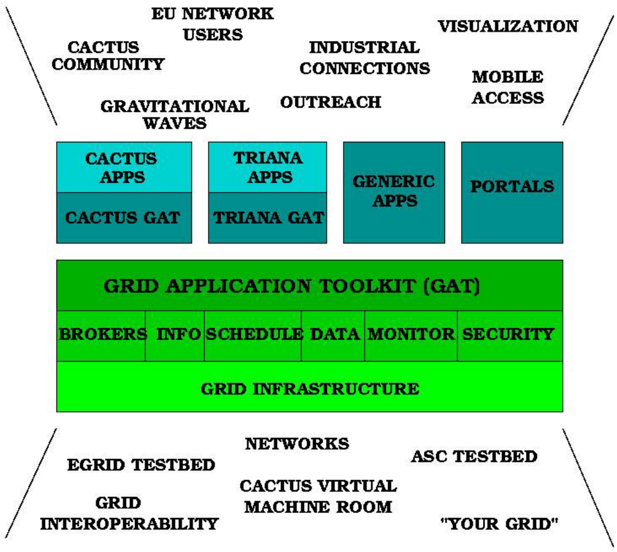
```
Schedule AHFinder at
Analysis
{
  EXTERNAL=yes
  LANG=C
 } "Finding Horizons"
```



GridLab

Egrid + US Friends: working to make this happen…

- Large EU Project Under Negotiation with EC…
- AEI, Lecce, Poznan, Brno, Amsterdam, ZIB-Berlin, Cardiff, Paderborn, Compaq, Sun, Chicago, ISI, Wisconsin
- 20 positions open!



Grid Related Projects

- GridLab *www.gridlab.org*
  - Enabling these scenarios
- ASC: Astrophysics Simulation Collaboratory *www.ascportal.org*
  - NSF Funded (WashU, Rutgers, Argonne, U. Chicago, NCSA)
  - Collaboratory tools, Cactus Portal
- Global Grid Forum (GGF) *www.gridforum.org*
  - Applications Working Group
- GrADs: Grid Application Development Software *www.isi.edu/grads*
  - NSF Funded (Rice, NCSA, U. Illinois, UCSD, U. Chicago, U. Indiana...)
- TIKSL/GriKSL *www.zib.de/Visual/projects/TIKSL/*
  - German DFN funded: AEI, ZIB, Garching
  - Remote online and offline visualization, remote steering/monitoring
- Cactus Team *www.CactusCode.org*
  - Dynamic distributed computing …

Summary

- Science/Engineering Drive/Demand Grid Development
  - Problems very large, need new capabilities
- Grids will fundamentally change research
  - *Enable problem scales far beyond present capabilities*
  - *Enable larger communities to work together (they'll need to)*
  - *Change the way researchers/engineers think about their work*
- Dynamic Nature of Grid makes problem much more interesting
  - *Harder*
  - *Matches dynamic nature of problems being studied*
- Need to get applications communities to rethink their problems
  - *The Grid is the computer…*
- Join the Applications Working Group of GGF
- Join our project: www.gridlab.org
  - *Work with us from here, or come to Europe!*

Credits: *this work resulted from a great team*

QuickTime™ and a
Sorenson Video decompressor
are needed to see this picture.