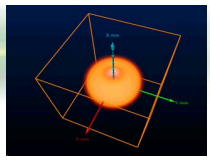
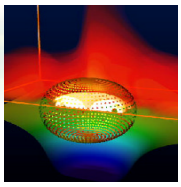
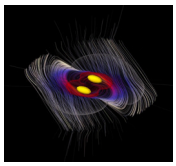
Applications for the Grid

Here at GGF1:
Gabrielle Allen, Thomas, Dramlitsch,
Gerd Lanfermann, Thomas Radke,
Ed Seidel

Max Planck Institute for Gravitational Physics,
(Albert Einstein Institute)

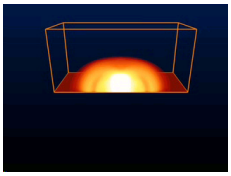


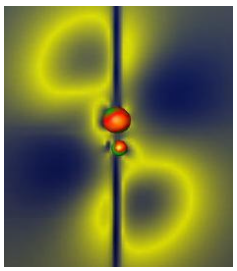


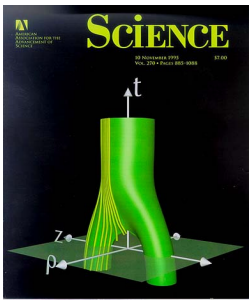


CACTUS is a freely available, modular, portable and manageable environment for collaboratively developing parallel, high-performance multi-dimensional simulations

THE GRID: Dependable, consistent, pervasive access to high-end resources

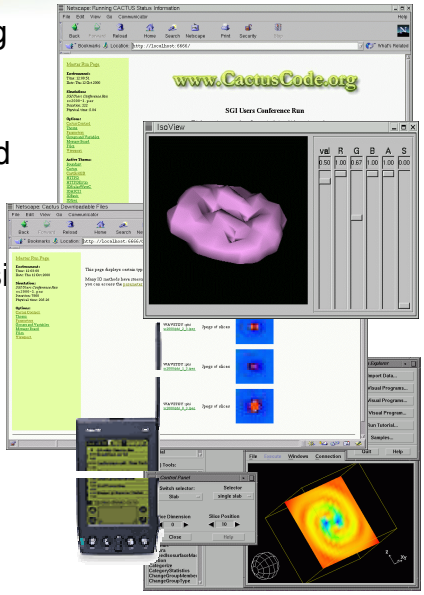






Working Cactus Grid Tools

- Remote monitoring and steering of simulations (thorn HTTPD).
- Just adding point-and-click visualization to Web Server, and embedded visualization.
- Remote visualization ... data streamed to local site for analysi with local clients "Window into simulation"
- Notification by Email at start of simulation, including details of where to connect (URLs) for other tools.
- Checkpointing and restart between machines.



Grand Picture


Remote Viz in St Louis

Remote Viz and steering from Berlin


Remote steering and monitoring from airport


Viz of data from previous simulations in SF café


IsoSurfaces

HDF5

http

DataGrid/DPSS Downsampling


T3E: Garching

Origin: NCSA

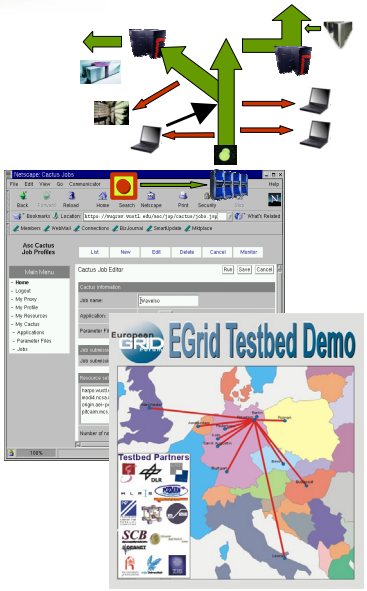

Globus

Simulations launched from Cactus Portal

Grid enabled Cactus runs on distributed machines

Developing Cactus Grid Tools

- Cactus Simulation Portal (*Astrophysics Simulation Collaboratory*, ASC) - Version 2 next month. Lots of ideas, users are designing it.
- Testbeds/VMRs - CVMR (big production machines), Egrid (test and development), EU Astrophysics.
- Dynamic scenarios ... Production Level Cactus Worm (Migration Tool) and more.
- Non-demo distributed simulations ... cheaper, faster, bigger
- Notification, Info Services, Data management ... requested by users



Grid Applications

- How can applications really exploit Grids?
- The Grid is much more than Portals, distributed computing, and remote visualization ... we should be thinking of new application scenarios
- Cactus Worm (Simulation Migrator) was a prototype of such classes of applications, this is a thorn (module) which can be added to any application for automatic migration between machines (using checkpoint files)
  - Dynamically locate new resources
  - Move appropriate data files between old and new machine
  - Get hold of appropriate executable for central repository
  - Restart application on new machine

Grid Applications

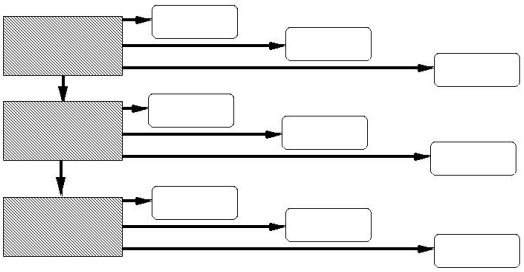
- Execution Staging
  - Use most appropriate (virtual) machine (fastest, cheapest, biggest)
    - Need it by Wednesday, only use 200 SUs, need 1GB.
  - Set parameters on the basis of available machines ...
    - eg anywhere between 70<nx<150 but I want to run it right now
- Simulation Redistribution
  - Put new grids on different machines (AMR, Multigrid)
  - Adapt automatically to varying processor speeds and application loads
  - Slow Startup ... start application now on a slow resource, and move to faster resources as they become available

Grid Applications

- Simulation Migration
  - Move to more appropriate machines (faster, cheaper)
  - Move to new resource at end of queue time
  - Convergence testing
    - Send coarser grid to different resources, either at start of simulation or dynamically at user request, or following some event
  - Look-Ahead
    - Spawn a downsampled resolution to a different resource to predict the future
  - Cloning/Multiple Universe
    - Dynamically initiate a cloned version with changed parameters

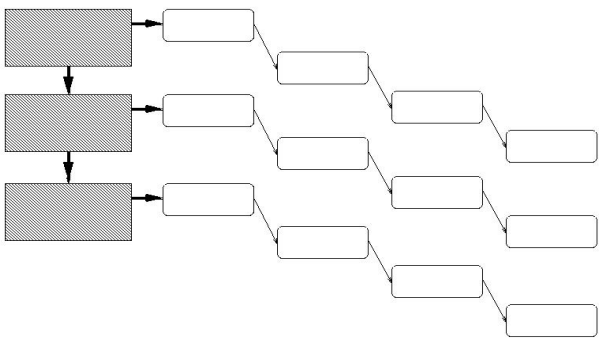
Grid Applications

- Spawn Independent Tasks
  - Analysis tasks
    - Don't need to be performed in line with simulation, send to different available (free) machines
  - Vector
    - Send to a vector pool of resources ... e.g. Calculate FFT of each grid variable, send each one to different machine



Grid Applications

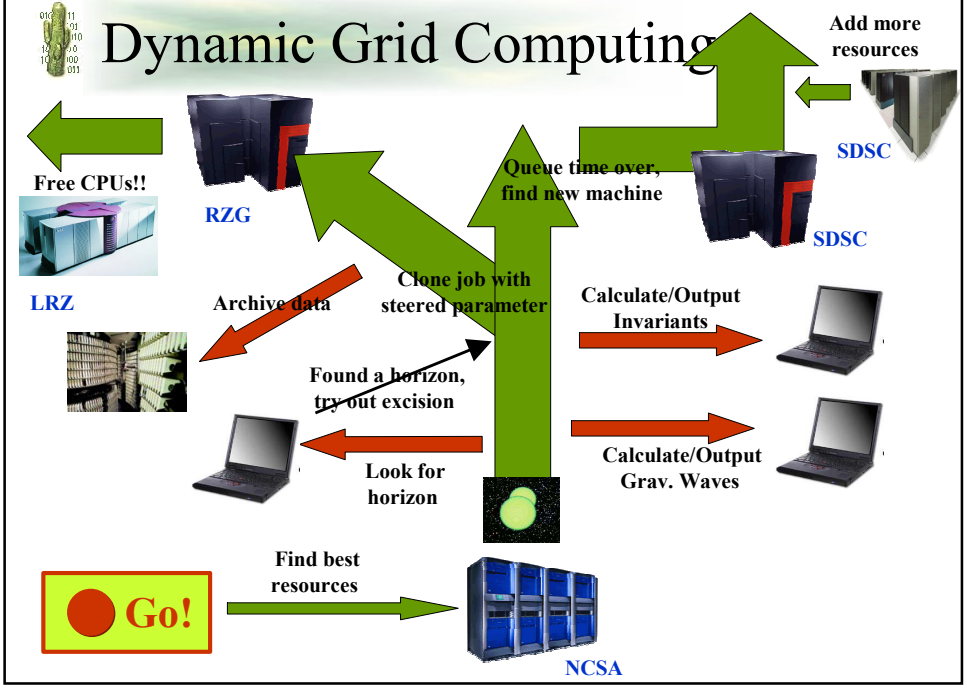
- Spawn Independent Tasks
  - Pipeline
    - Series of tasks performed on same data



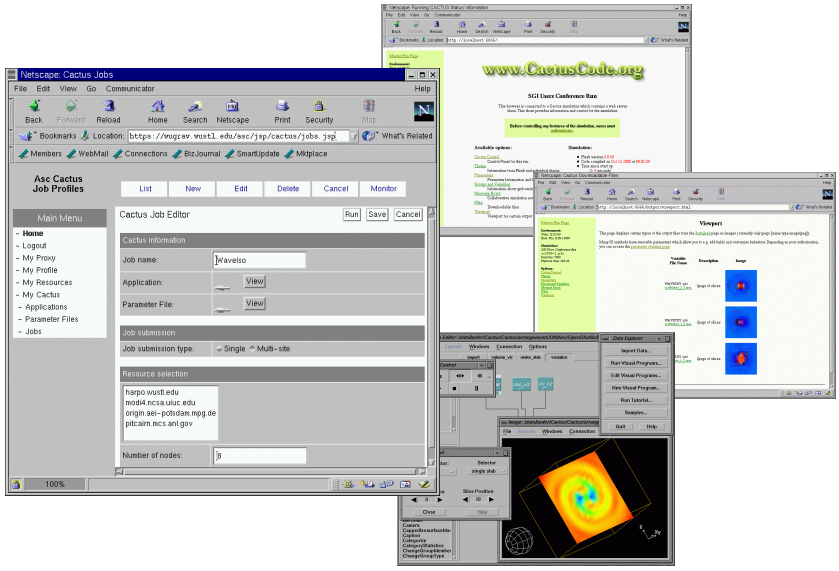
Grid Applications

- Task Farming
  - Automated parameter studies
  - Genetic parmater studies
    - Use heuristic algorithms to cover large parameter spaces
  - Manage embarrassing parallel applications
- Make use of
  - Condor, Entropia
  - Grid scripting languages ... everything available from the command line.
- Many other possibilities ...

Dynamic Grid Computing



Users View ☺ Has To Be Easy!



Cactus Grid Application Toolkit (GAT)

Cactus Application

[Black hole collisions, chemical engineering, cosmology, hydrodynamics]

Grid Tools

[Worm, Spawner, Taskfarmer]

Cactus Programming APIs

[Grid variables, parameters, scheduling, IO, interpolation, reduction, driver]

Grid Application Toolkit

[Information services, resource broker, data management, monitoring]

Grid Service

Grid Service

Grid Service

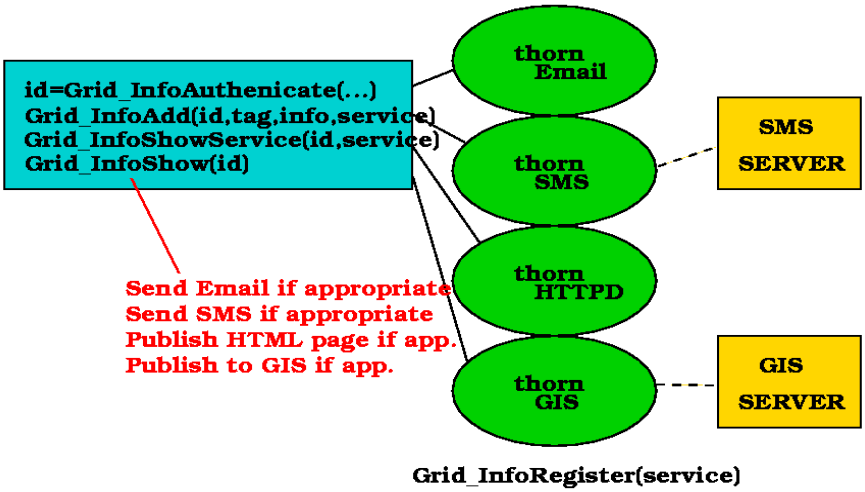
YOUR GRID

Distributed resources, different OS, software

Grid Application Toolkit (GAT)

- Application developer should be able to build simulations with tools that easily enable dynamic grid capabilities
- Want to build programming API to:
  - Query/Publish to Information Services
    - Application, Network, Machine and Queue status
  - Resource Brokers
    - Where to go, how to decide?, how to get there?
  - Move and manage data, compose executables
    - Gsiscp, gsiftp, streamed HDF5, scp, GASS, ...
  - Higher level grid tools:
    - Migrate, Spawn, Taskfarm, Vector, Pipeline, Clone, ...
  - Notification
    - Send me an Email, SMS, fax, page, when something happens.
- Much more.

Cactus GAT: Example



Cactus GAT: Example