Grid Computing: NetSolve and the GrADS Project

Jack Dongarra
Innovative Computing Lab
University of Tennessee
http://www.cs.utk.edu/~dongarra/

1

Innovative Computing Laboratory

- Numerical Linear Algebra
- Heterogeneous Distributed Computing
- Software Repositories
- Performance Evaluation

Software and ideas have found there way into many areas of Computational Science

Around 40 people: At the moment...

16 Researchers: Research Assoc/Post-Doc/Research Prof

15 Students: Graduate and Undergraduate

8 Support staff: Secretary, Systems, Artist

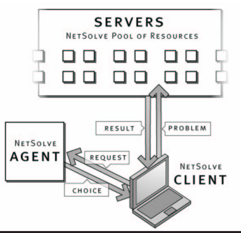
1 Long term visitors (Japan)

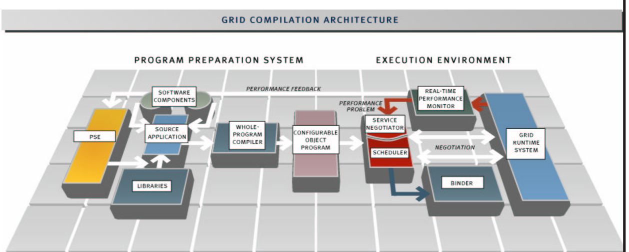
Responsible for about $4M/years in research funding from NSF, DOE, DOD, etc

3

Outline

- Grid computing in general
- Two approaches to Grid numerical libraries, some early experiments
  - NetSolve – Grid enabled portal - software servers
  - GrADS Project - Software Technology for Problem Solving on Computational Grids





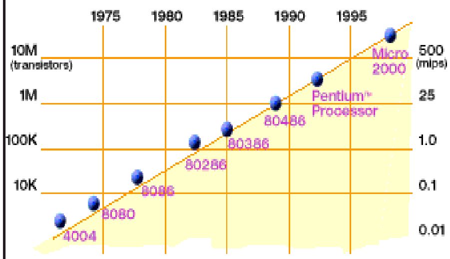
Grid Computing

- Enable communities ("virtual organizations") to share geographically distributed resources as they pursue common goals—in the absence of central control, omniscience, trust relationships.
- Resources (HPC systems, visualization systems & displays, storage systems, sensors, instruments, people) are integrated via 'middleware' to facilitate use of all resources.

Why Grids?

- Large problems require teamwork and computation
- Power of any single resource is small compared to aggregations of resources
- Network connectivity is increasing rapidly in bandwidth and availability

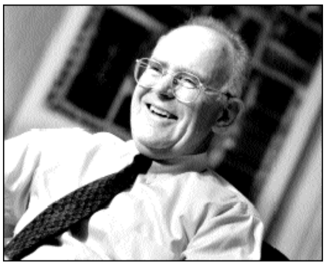
Technology Trends: Microprocessor Capacity



2X transistors/Chip Every 1.5 years

Called "Moore's Law"

Microprocessors have become smaller, denser, and more powerful.
Not just processors, bandwidth, storage, etc

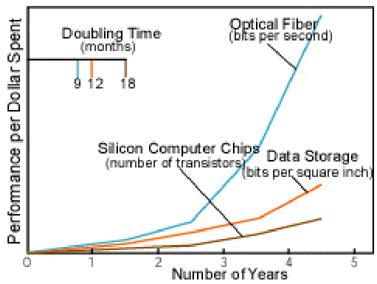


Gordon Moore (co-founder of Intel) predicted in 1965 that the transistor density of semiconductor chips would double roughly every 18 months.

7

Network Bandwidth Growth

- Network vs. computer performance
  - Computer speed doubles every 18 months
  - Network speed doubles every 9 months
  - Difference = order of magnitude per 5 years
- 1986 to 2000
  - Computers: x 500
  - Networks: x 340,000
- 2001 to 2010
  - Computers: x 60
  - Networks: x 4000



Moore's Law vs. storage improvements vs. optical improvements. Graph from **Scientific American** (Jan-2001) by Cleo Vilett, source Vined Khoslan, Kleiner, Caufield and Perkins.

Bandwidth Won't Be A Problem Soon -- Bisection Bandwidth (BB) Across the US

- 1971 - BB 112 Kb/s
- 1986 - BB 1 Mb/s
- 2001 - BB 200 Gb/s
- Today in the lab, 4000 channels on single fiber and each channel 10 Gb/s
- 12 strands of fiber can carry 4000*10 Gb/s or 40 Tb/s
- 5 backbone network across the US each w/ 2 sets of 12 strands can provide 2.4 Pb/s
- "When the Network is as fast as the computer's internal links, the machine disintegrates across the Net into a set of special purpose appliances"
  - Gilder Technology Report June 2000
- Internet doubling every 9 months
- Factor of 100 in 5 years
- BB will grow be a factor of 12000.

9

Grid Possibilities

- A biochemist exploits 10,000 computers to screen 100,000 compounds in an hour
- 1,000 physicists worldwide pool resources for petaflop analyses of petabytes of data
- Civil engineers collaborate to design, execute, & analyze shake table experiments
- Climate scientists visualize, annotate, & analyze terabyte simulation datasets
- An emergency response team couples real time data, weather model, population data

10

Some Grid Usage Models

- Distributed computing: job scheduling on Grid resources with secure, automated data transfer
- Workflow: synchronized scheduling and automated data transfer from one system to next in pipeline (e.g. compute-viz-storage)
- Coupled codes, with pieces running on different systems simultaneously
- Meta-applications: parallel apps spanning multiple systems

11

Grid Usage Models

- Some models are similar to models already being used, but are made much simpler by the Grid due to:
  - single sign-on
  - automatic process scheduling
  - automated data transfers
- But Grids can encompass new resources likes sensors and instruments, so new usage models will arise

12

Example Application Projects

- Earth Systems Grid: environment (US DOE)
- EU DataGrid: physics, environment, etc. (EU)
- EuroGrid: various (EU)
- Fusion Collaboratory (US DOE)
- GridLab: astrophysics, etc. (EU)
- Grid Physics Network (US NSF)
- MetaNEOS: numerical optimization (US NSF)
- NEESgrid: civil engineering (US NSF)
- Particle Physics Data Grid (US DOE)

13

Some Grid Requirements – Systems/Deployment Perspective

- Identity & authentication
- Authorization & policy
- Resource discovery
- Resource characterization
- Resource allocation
- (Co-)reservation, workflow
- Distributed algorithms
- Remote data access
- High-speed data transfer
- Performance guarantees
- Monitoring
- Adaptation
- Intrusion detection
- Resource management
- Accounting & payment
- Fault management
- System evolution
- Etc.

14

Some Grid Requirements – User Perspective

- Single sign-on: authentication to any Grid resources authenticates for all others
- Single compute space: one scheduler for all Grid resources
- Single data space: can address files and data from any Grid resources
- Single development environment: Grid tools and libraries that work on all grid resources

15

The Systems Challenges: Resource Sharing Mechanisms That…

- Address security and policy concerns of resource owners and users
- Are flexible enough to deal with many resource types and sharing modalities
- Scale to large number of resources, many participants, many program components
- Operate efficiently when dealing with large amounts of data & computation

16

The Security Problem

- Resources being used may be extremely valuable & the problems being solved extremely sensitive
- Resources are often located in distinct administrative domains
  - Each resource may have own policies & procedures
- The set of resources used by a single computation may be large, dynamic, and/or unpredictable
  - Not just client/server
- It must be broadly available & applicable
  - Standard, well-tested, well-understood protocols
  - Integration with wide variety of tools

17

The Resource Management Problem

- Enabling secure, controlled remote access to computational resources and management of remote computation
  - Authentication and authorization
  - Resource discovery & characterization
  - Reservation and allocation
  - Computation monitoring and control

18

Grid Systems Technologies

- Systems and security problems addressed by new protocols & services. E.g., Globus:
  - Grid Security Infrastructure (GSI) for security
  - Globus Metadata Directory Service (MDS) for discovery
  - Globus Resource Allocations Manager (GRAM) protocol as a basic building block
    - Resource brokering & co-allocation services
  - GridFTP, IBP for data movement

19

The Programming Problem

- How does a user develop robust, secure, long-lived applications for dynamic, heterogeneous, Grids?
- Presumably need:
  - Abstractions and models to add to speed/robustness/etc. of development
  - Tools to ease application development and diagnose common problems
  - Code/tool sharing to allow reuse of code components developed by others

20

Examples of Grid Programming Technologies

- MPICH-G2: Grid-enabled message passing
- CoG Kits, GridPort: Portal construction, based on N-tier architectures
- GDMP, Data Grid Tools, SRB: replica management, collection management
- Condor-G: simple workflow management
- Legion: object models for Grid computing
- NetSolve: Network enabled solver
- Cactus: Grid-aware numerical solver framework
  - Note tremendous variety, application focus

21

MPICH-G2: A Grid-Enabled MPI

- A complete implementation of the Message Passing Interface (MPI) for heterogeneous, wide area environments
  - Based on the Argonne MPICH implementation of MPI (Gropp and Lusk)
- Globus services for authentication, resource allocation, executable staging, output, etc.
- Programs run in wide area without change
- See also: MetaMPI, PACX, STAMPI, MAGPIE

22

www.globus.org/mpi

Grid Events

- Global Grid Forum: working meeting
  - Meets 3 times/year, alternates U.S.-Europe, with July meeting as major event
- HPDC: major academic conference
  - HPDC-11 in Scotland with GGF-8, July 2002
- Other meetings include
  - IPDPS, CCGrid, EuroGlobus, Globus Retreats

23

www.gridforum.org, www.hpdc.org

Useful References

- Book (Morgan Kaufman)
  - www.mkp.com/grids
- Perspective on Grids
  - "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", IJHPCA, 2001
  - www.globus.org/research/papers/anatomy.pdf
- All URLs in this section of the presentation, especially:
  - www.gridforum.org, www.grids-center.org, www.globus.org

24

Emergence of Grids

- But Grids enable much more than apps running on multiple computers (which can be achieved with MPI alone)
  - virtual operating system: provides global workspace/address space via a single login
  - automatically manages files, data, accounts, and security issues
  - connects other resources (archival data facilities, instruments, devices) and people (collaborative environments)

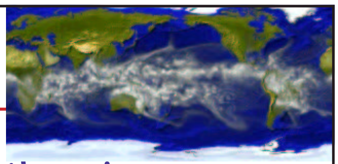
25

Grids Are Inevitable

- Inevitable (at least in HPC):
  - leverages computational power of all available systems
  - manages resources as a single system--easier for users
  - provides most flexible resource selection and management, load sharing
  - researchers' desire to solve bigger problems will always outpace performance increases of single systems; just as multiple processors are needed, 'multiple multiprocessors' will be deemed so

26

In the past: Isolation
Motivation for Grid Computing



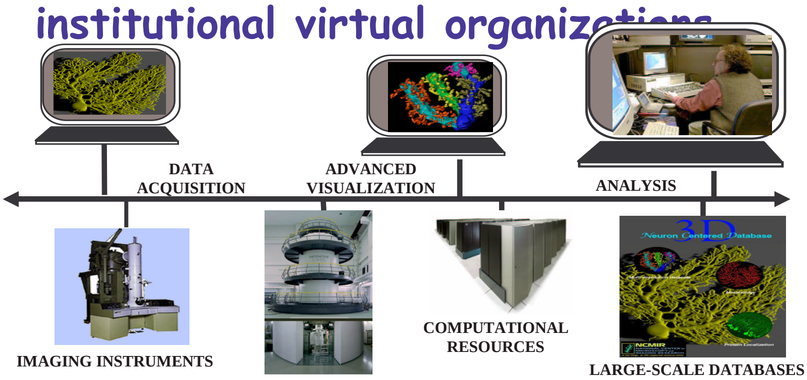
- There is a complex interplay and increasing interdependence among the sciences.
- What we do as collaborative infrastructure developers has profound influence on the future of science.
- This is especially true as theory, experiment, and computational models provide insights into the bedrock principles of nature.
- Networking, distributed computing, and parallel computation research have matured to make it possible for distributed systems to support high-performance applications, but...
  - Resources are dispersed
  - Connectivity is variable
  - Dedicated access may not be possible

Today: Collaboration

27

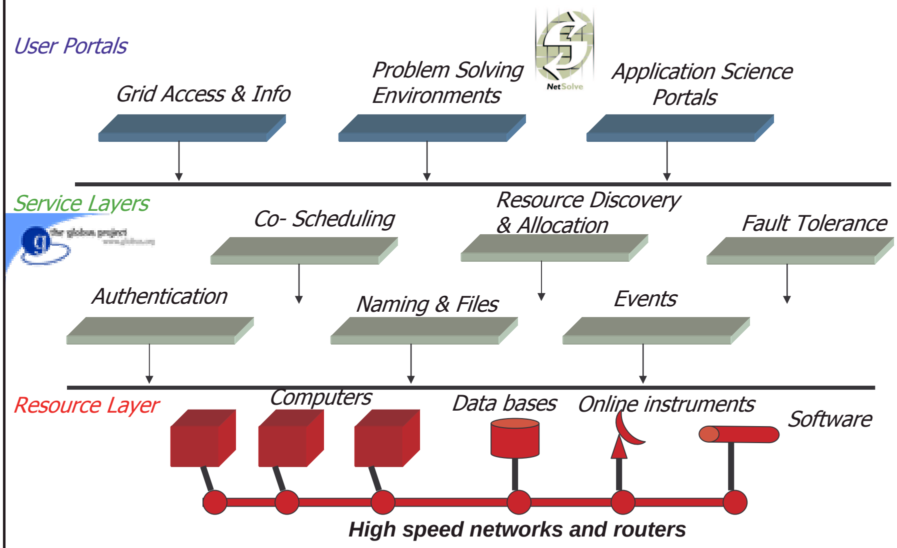
What is Grid Computing?

Resource sharing & coordinated problem solving in dynamic, multi-institutional virtual organizations



28

The Grid Architecture Picture



29

Globus Grid Services

- The Globus toolkit provides a range of basic Grid services
  - Security, information, fault detection, communication, resource management, ...
- These services are simple and orthogonal
  - Can be used independently, mix and match
  - Programming model independent
- For each there are well-defined APIs
- Standards are used extensively
  - E.g., LDAP, GSS-API, X.509, ...
- You don't program in Globus, it's a set of tools like Unix

30

Evolution of a Community Grid Model

- Roll your own SW but agree on interfaces, service architecture, standards

Applications

User-focused grid middleware, tools, and services

Common Infrastructure layer (NMI, GGF standards, OGSA etc.)

Grid Resources

Maturation of Grid Computing

- Research focus moving from building of basic infrastructure and application demonstrations to
  - Middleware
  - Usable production environments
  - Application performance
  - Scalability → Globalization
- Development, research, and integration happening outside of the original infrastructure groups
- Grids becoming a first-class tool for scientific communities
  - GriPhyN (Physics), BIRN (Neuroscience), NVO (Astronomy), Cactus (Physics), …

The Computational Grid is…

- …a distributed control infrastructure that allows applications to treat compute cycles as commodities.
- Power Grid analogy
  - Power producers: machines, software, networks, storage systems
  - Power consumers: user applications
- Applications draw power from the Grid the way appliances draw electricity from the power utility.
  - Seamless
  - High-performance
  - Ubiquitous
  - Dependable

33

Computational Grids and Electric Power Grids

- Why the Computational Grid is like the Electric Power Grid
  - Electric power is ubiquitous
  - Don't need to know the source of the power (transformer, generator) or the power company that serves it

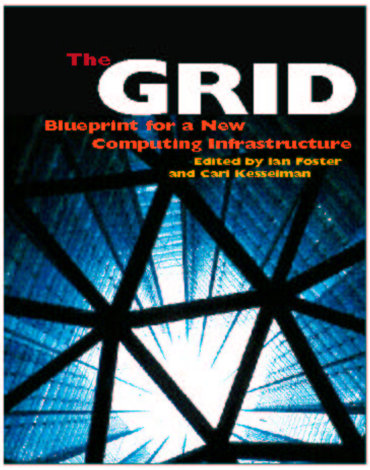
- Why the Computational Grid is different from the Electric Power Grid
  - Wider spectrum of performance
  - Wider spectrum of services
  - Access governed by more complicated issues
    - Security
    - Performance
    - Socio-political factors

34

An Emerging Grid Community

1995-2000

- "Grid book" gave a comprehensive view of the state of the art
- Important infrastructure and middleware efforts initiated
  - Globus
  - Legion
  - Condor
  - NetSolve, Ninf
  - Storage Resource Broker
  - Network Weather Service
  - AppLeS, …



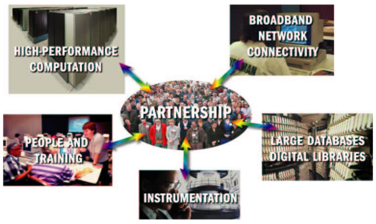
35

Grids are Hot

IPG NAS-NASA	http://nas.nasa.gov/~wej/home/IPG
Globus	http://www.globus.org/
Legion	http://www.cs.virgina.edu/~grimshaw/
AppLeS	http://www-cse.ucsd.edu/groups/hpcl/apples
NetSolve	http://www.cs.utk.edu/netsolve/
NINF	http://phase.etl.go.jp/ninf/
Condor	http://www.cs.wisc.edu/condor/
ACI Grid	http://www.recherche.gouv.fr/recherche/aci/grid.htm
WebFlow	http://www.npac.syr.edu/users/gcf/
LoCI	http://loci.cs.utk.edu/

Broad Acceptance of Grids as a Critical Platform for Computing

- Widespread interest from government in developing computational Grid platforms



NSF's Cyberinfrastructure

NASA's Information Power Grid

DOE's Science Grid

37

Broad Acceptance of Grids as a Critical Platform for Computing

- Widespread interest from industry in developing computational Grid platforms
- IBM, Sun, Entropia, Avaki, Platform, …

On August 2, 2001, IBM announced a new corporate initiative to support and exploit Grid computing.

AP reported that IBM was investing $4 billion into building 50 computer server farms around the world.

AVAKI

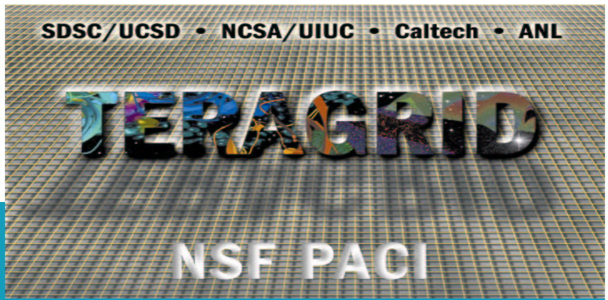
ENTROPIA

UNITED DEVICES

38

Grids Form the Basis of a National Information Infrastructure

August 9, 2001: NSF Awarded $53,000,000 to SDSC/NPACI and NCSA/Alliance for TeraGrid



TeraGrid will provide in aggregate

- *13.6 trillion calculations per second*
- *Over 600 trillion bytes of immediately accessible data*
- *40 gigabit per second network speed*
- *Provide a new paradigm for data-oriented computing*
  - *Critical for disaster response, genomics, environmental modeling, etc.*

39

Bookmarks Location: http://www.gridforum.org/ What's Related

Peer-to-Peer | Security | Scheduling | Performance and Information Services | Architecture | Data | Applications & Programming Models

GGF

About GGF
- Overview
- Who's Involved?
- People
- Structure & Process
- History & Background
- Origin
- Document Process

Get Involved
- How to Get Involved
- Join
- Working Groups
- Membership
- Sponsor Membership
- Sponsors
- Scholarships
- Job Postings
- Grid Initiatives

News & Events
- Announcements
- What's New?
- In the Press
- Related Events
- Newsletters

Contact the GGF
- General Inquiries
- GGF People
- GGF Offices

NEXT GGF Event
GGF5 21-24 July 2002 Edinburgh, Scotland, UK

Subscribe
GGF eAnnounce

Global Grid Forum

JOIN US for GGF5 in Edinburgh, Scotland,
(21-24 July 2002)

Click HERE for information on hotels, GGF5 program, transportation, registration and more...

GGF5 Tutorials **Registration** is now **OPEN** - Space is limited for each Tutorial, so please make your plans to participate as soon as possible. It will be posted when a Tutorial is **CLOSED**.

GGF5 Registration Fee Information - **ADVANCE Registration CLOSES on 10 JULY** -discounted fees will no longer apply.

Over 850 participants (and almost 3 for H C-11) are already registered to join the activities in Edinburgh, Scotland for oth GGF5 (GGF5 Welcome Reception on Sunday evening 21-July) nd H DC11 (24-26 July).
A joint GGF5/HPDC11 Dinner Reception celebration will be held on Wednesday, 24 July - please plan to attend!

!?

Global Grid Forum

Related Events

The Internet Society's 12th Annual INET Conference
18-21 June, 2002 in Arlington, Virginia

2nd Grid Forum Korea workshop

8th workshop on Job Scheduling Strategies for Parallel Processing is scheduled to be held in conjunction with HPDC11/GGF5 in Edinburgh on July 24

SuperComputing 2002 -- SC2002
From Terabytes to Insights High Performance Networking and Computing
Baltimore Nov 16-22

NeSC Workshop on Applications and Testbeds on the Grid

CALL FOR PAPERS - The March 2003 issue of **IEEE Computer Graphics and Applications** will be a special issue on Graphics Applications for Grid Computing. The Call for Papers can be found at:

New Documents

Final: GGF Document Series and Process. This document has been approved as GFD-C.1 (.doc, .pdf).
GGF work is being captured as a persistent document series modeled after the Request for Comments (RFC) documents used in the Internet Standards Process. This document outlines processes for the GGF document series, including the development of "standards" in the form of recommendations track documents.

Participate in public comment on GGF draft documents.

Drafts as of May 2002:

Draft: GGF Structure (.doc, .pdf).
This document describes the restructuring of GGF (in July 2001) into Areas as well as distringuishing some groups as "research" groups and others as "working" groups. It is provided primarily to document that process for historical purposes.

Draft: GGF Management (.doc, .pdf).
GGF is managed by a Chair in cooperation with the GGF Steering Group (GFSG). An external GGF advisory committee (GFAC) provides input to the management process. This document outlines GGF management roles and related processes.
This document contains guidelines and processes for creating working groups

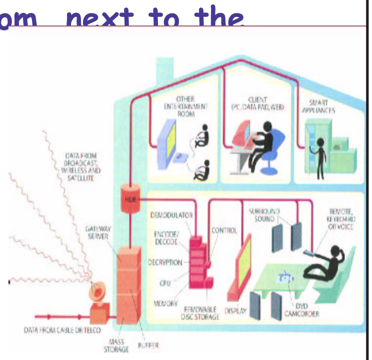
"Grids Meet Peer-to-Peer"

- Currently Grids and P2P have distinct foci
  - Grids: small scale, general purpose, static, managed
  - P2P: large scale, specialized, dynamic, unmanaged
- Future systems will combine aspects of both
  - Large scale, general-purpose, dynamic, self-managed
- Keys to progress: <u>exploiting heterogeneity</u> and <u>self organization</u>

41

Peer to Peer Computing

- Peer-to-peer is a style of networking in w a group of computers communicate directly each other.
- Wireless communication
- Home computer in the utility room, next to the water heater and furnace.
- Web tablets
- Imbedded computers in things all tied together.
  - Books, furniture, milk cartons, etc
- Smart Appliances
  - Refrigerator, scale, etc



Internet On Everything

Distributed Computing

- Concept has been around for two decades
- Basic idea: run scheduler across systems to runs processes on least-used systems first
  - Maximize utilization
  - Minimize turnaround time
- Have to load executables and input files to selected resource
  - Shared file system
  - File transfers upon resource selection

44

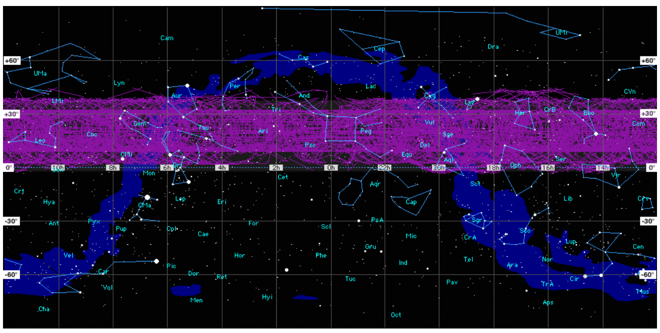
Examples of Distributed Computing

- Workstation farms, Condor flocks, etc.
  - Generally share file system
- SETI@home project, Entropia, etc.
  - Only one source code; copies correct binary code and input data to each system
- Napster, Gnutella: file/data sharing
- NetSolve
  - Runs numerical kernel on any of multiple independent systems, much like a Grid solution

45

SETI@home: Global Distributed Computing

- Running on 500,000 PCs, ~1000 CPU Years per Day
  - 485,821 CPU Years so far
- Sophisticated Data & Signal Processing Analysis
- Distributes Datasets from Arecibo Radio Telescope



46

SETI@home

- Use thousands of Internet-connected PCs to help in the search for extraterrestrial intelligence.
- Uses data collected with the Arecibo Radio Telescope, in Puerto Rico
- When their computer is idle or being wasted this software will download a 300 kilobyte chunk of data for analysis.
- The results of this analysis are sent back to the SETI team, combined with thousands of other participants.

- Largest distributed computation project in existence
  - ~ 400,000 machines
  - Averaging 27 Tflop/s
- Today many companies trying this for profit.

47

CNN.com / SCI-TECH

PCs tapped to help fight anthrax

January 22, 2002 Posted: 12:16 PM EST (1716 GMT)

SAN JOSE, California (AP) -- A coalition of scientists and technology companies is asking people around the world to use their computers' extra processing power to help search for a cure for anthrax.

The project follows similar efforts to use "distributed computing" to hunt for extraterrestrial life and a cure for cancer. It is being launched Tuesday to help Oxford University researchers find ways to treat anthrax that can no longer be treated by antibiotics.

The project is based on the premise that the average personal computer uses between 13 percent and 18 percent of its processing power at any given time. It employs "peer-to-peer" technology, in which millions of computers can share files over the Internet.

Participants download a screen-saver that runs whenever their computers have resources to spare, and uses that power to perform computations for the project. When the user connects to the Internet, the computer sends data back to a central hub and gets another assignment.

The company that designed the program, United Devices Inc. of Austin, Texas, promises that no personal information on participants' PCs can be compromised while they take part.

If the project attracts more than 160,000 participants, it can give researchers more computational power than the world's 10 best supercomputers combined, said United Devices spokesman Andy Prince.

With enough participants, the project would provide researchers 10 times more power than the world's best supercomputer, said Graham Richards, the Oxford professor leading the study.

"The screen-saver doesn't cost you anything, and at least you're taking part in something, adding your bit," he said.

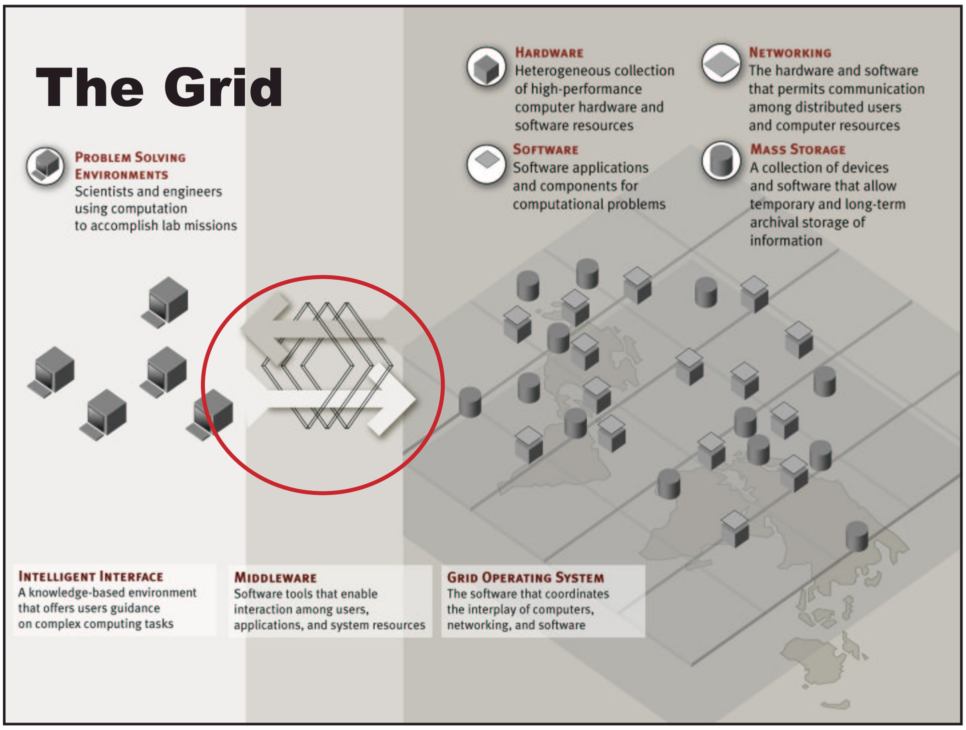
Intel, Microsoft involved

Scientists have discovered that the anthrax toxin is made up of three proteins that are not toxic on their own but become toxic after binding together.

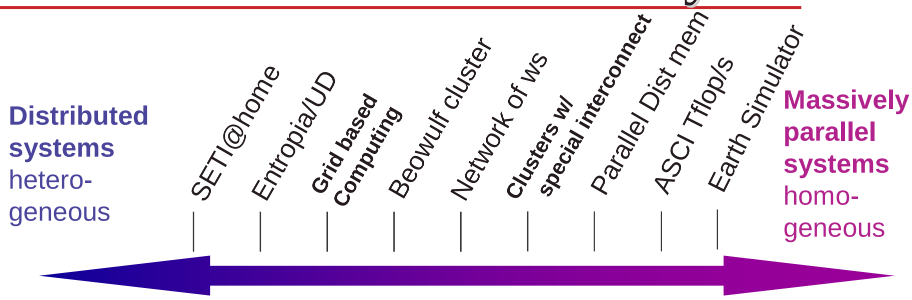
Grid Computing - from ET toAnthrax

UNITED DEVICES™

48



Distributed and Parallel Systems



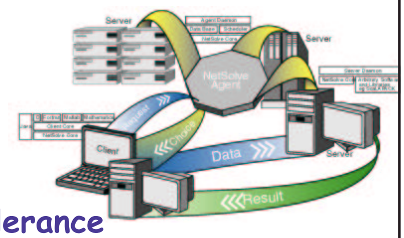
- Gather (unused) resources
- Steal cycles
- System SW manages resources
- System SW adds value
- 10% - 20% overhead is OK
- Resources drive applications
- Time to completion is not critical
- Time-shared
- SETI@home
  - ~ 400,000 machines
  - Averaging 27 Tflop/s

- Bounded set of resources
- Apps grow to consume all cycles
- Application manages resources
- System SW gets in the way
- 5% overhead is maximum
- Apps drive purchase of equipment
- Real-time constraints
- Space-shared
- Earth Simulator
  - 5000 processors
  - Averaging 35 Tflop/s

50

Motivation for NetSolve

Design an easy-to-use tool to provide efficient and uniform access to a variety of scientific packages on UNIX and Window's platforms



Basics

- Client-Server Design
- Non-hierarchical system
- Load Balancing and Fault Tolerance
- Heterogeneous Environment Supported
- Multiple and simple client interfaces
- Built on standard components

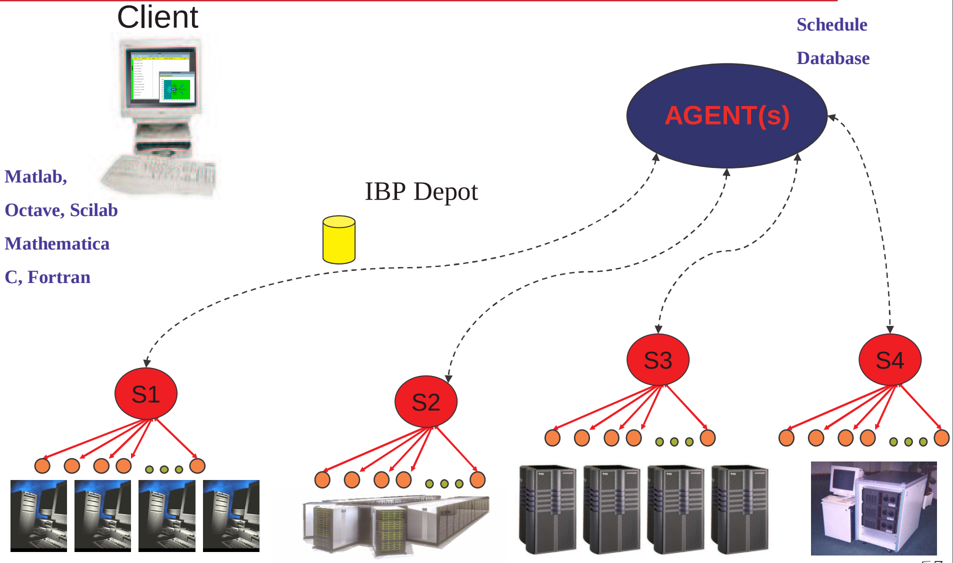
51

NetSolve Network Enabled Server

- NetSolve is an example of a Grid based hardware/software/data server.
- Based on a Remote Procedure Call model but with …
  - resource discovery, dynamic problem solving capabilities, load balancing, fault tolerance asynchronicity, security, …
- Easy-of-use paramount
- Its about providing transparent access to resources.

52

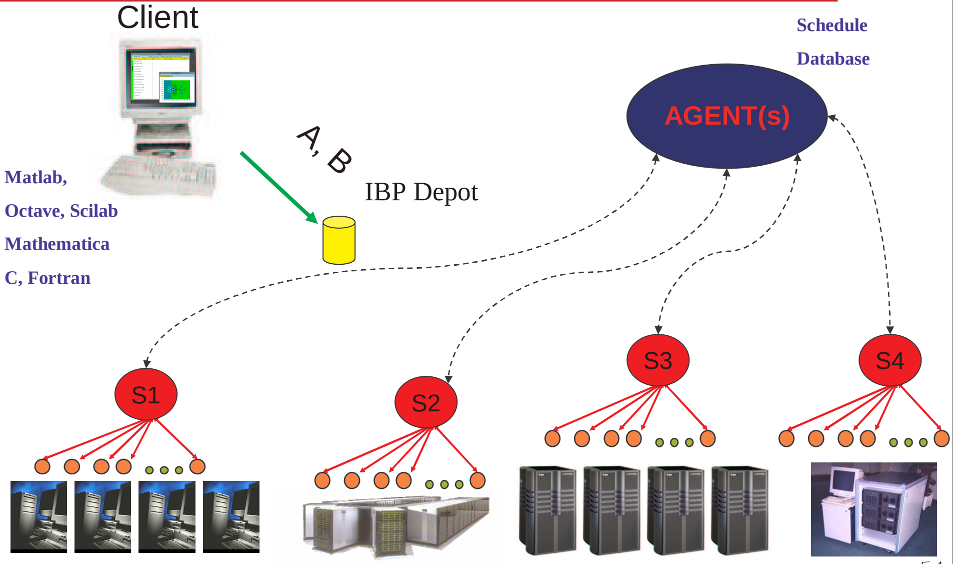
NetSolve: The Big Picture



53

No knowledge of the grid required, RPC like.

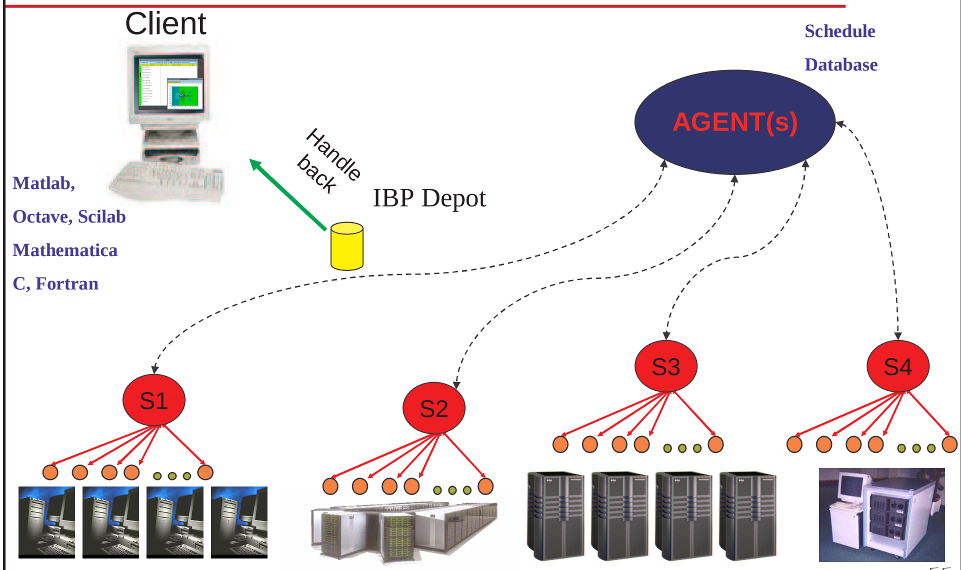
NetSolve: The Big Picture



54

No knowledge of the grid required, RPC like.

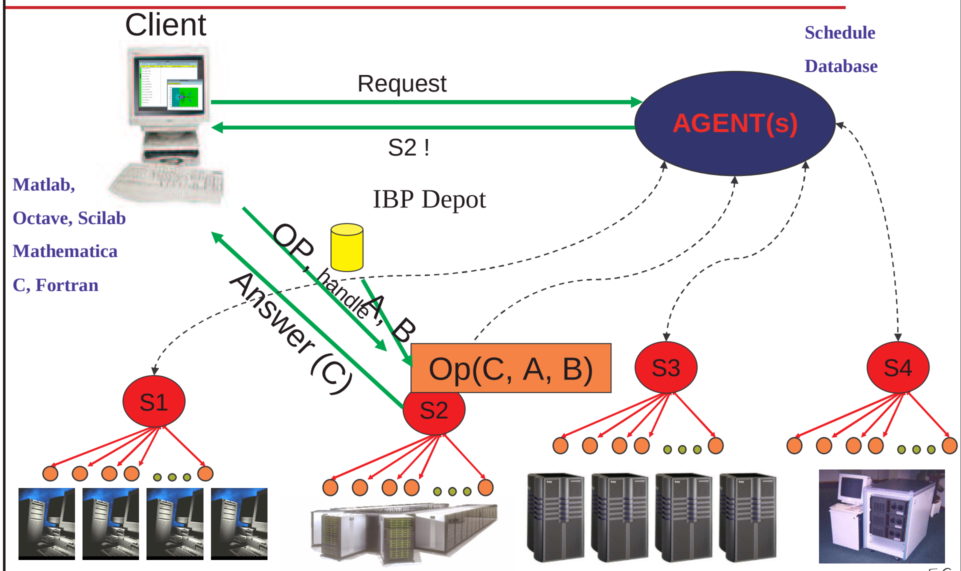
NetSolve: The Big Picture



No knowledge of the grid required, RPC like.

55

NetSolve: The Big Picture

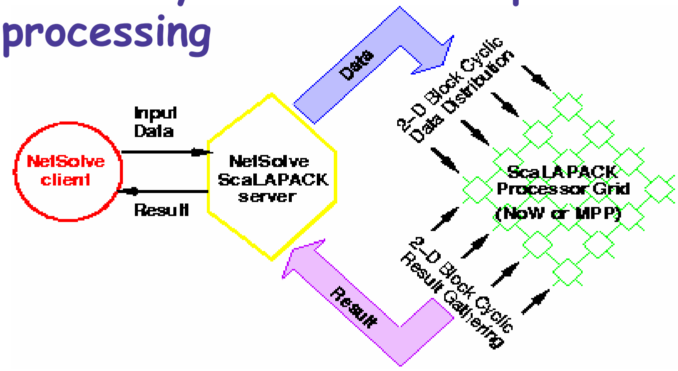


No knowledge of the grid required, RPC like.

56

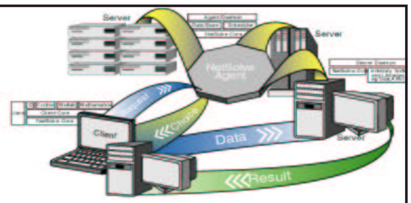
Hiding the Parallel Processing

- User maybe unaware of parallel processing



- NetSolve takes care of the starting the message passing system, data distribution, and returning the results.

Basic Usage Scenarios



- Grid based numerical library routines
  - User doesn't have to have software library on their machine, LAPACK, SuperLU, ScaLAPACK, PETSc, AZTEC, ARPACK
- Task farming applications
  - "Pleasantly parallel" execution eg Parameter studies
- Remote application execution
  - Complete applications with user specifying input parameters and receiving output
- "Blue Collar" Grid Based Computing
  - Does not require deep knowledge of network programming
  - Level of expressiveness right for many users
  - User can set things up, no "su" required
  - In use today, up to 200 servers in 9 countries
- Can plug into Globus, Condor, NINF, …

NetSolve Agent

- Name server for the NetSolve system.
- Information Service
  - client users and administrators can query the hardware and software services available.
- Resource scheduler
  - maintains both static and dynamic information regarding the NetSolve server components to use for the allocation of resources

59

NetSolve Agent

- Resource Scheduling (cont'd):
  - CPU Performance (LINPACK).
  - Network bandwidth, latency.
  - Server workload.
  - Problem size/algorithm complexity.
  - Calculates a "Time to Compute." for each appropriate server.
  - Notifies client of most appropriate server.

60

NetSolve Client

- Function Based Interface.
- Client program embeds call from NetSolve's API to access additional resources.
- Interface available to C, Fortran, Matlab, Octave, Scilab, and Mathematica.
- Opaque networking interactions.
- NetSolve can be invoked using a variety of methods: blocking, non-blocking, task farms, …

61

NetSolve Client

- Intuitive and easy to use.
- Matlab Matrix multiply e.g.:
  - A = matmul(B, C);

    A = netsolve('matmul', B, C);

- Possible parallelisms hidden.

- In Matlab:

```
[x,its]=netsolve('sparse_iterative_solve','PETSC',A,rhs,1.e-6,500);
[x]=netsolve('sparse_direct_solve','MA28',A,rhs,0.3,1);
```

62

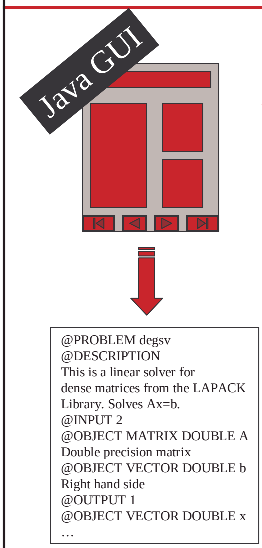
NetSolve Client

i. Client makes request to agent.

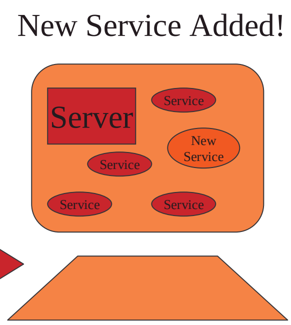
ii. Agent returns list of servers.

iii. Client tries first one to solve problem.

Generating New Services in NetSolve



- Add additional functionality
  - Describe the interface
  - Generate wrapper
  - Install into server



Task Farming - Multiple Requests To Single Problem

- **A Solution:**
  - Many calls to netslnb( ); /* non-blocking */
- **Farming Solution:**
  - Single call to netsl_farm( );
- **Request iterates over an "array of input parameters."**
- **Adaptive scheduling algorithm.**
- **Useful for parameter sweeping, and independently parallel applications.**

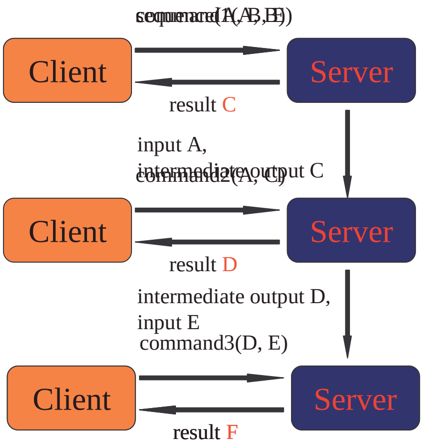
65

Data Persistence

- **Chain together a sequence of NetSolve requests.**
- **Analyze parameters to determine data dependencies. Essentially a DAG is created where nodes represent computational modules and arcs represent data flow.**
- **Transmit superset of all input/output parameters and make persistent near server(s) for duration of sequence execution.**
- **Schedule individual request modules for execution.**

66

Data Persistence (cont'd)

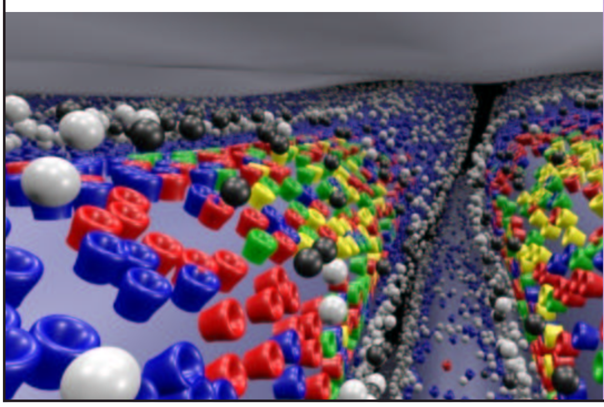


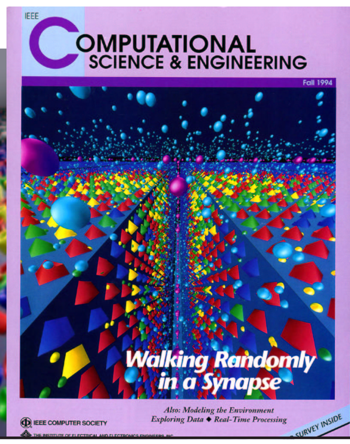

```
netsl_begin_sequence( );
netsl("command1", A, B, C);
netsl("command2", A, C, D);
netsl("command3", D, E, F);
netsl_end_sequence(C, D);
```

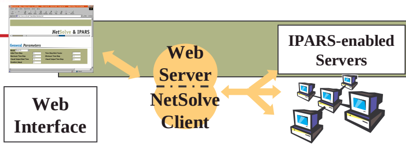
67

NPACI Alpha Project - MCell: 3-D Monte-Carlo Simulation of Neuro-Transmitter Release in Between Cells

- UCSD (F. Berman, H. Casanova, M. Ellisman), Salk Institute (T. Bartol), CMU (J. Stiles), UTK (Dongarra, M. Miller, R. Wolski)
- Study how neurotransmitters diffuse and activate receptors in synapses
- blue unbounded, red singly bounded, green doubly bounded closed, yellow doubly bounded open







*Net*Solve & IPARS

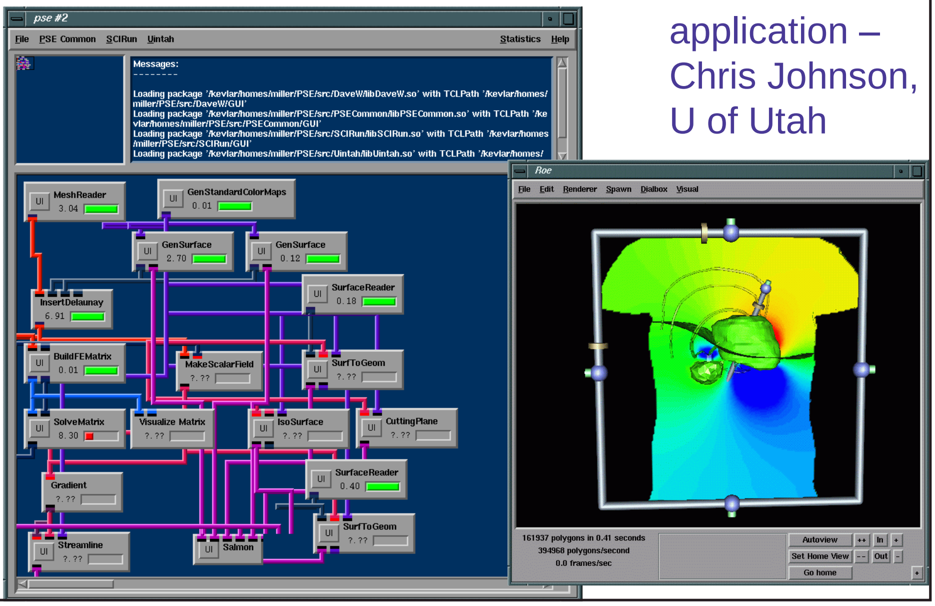
- Integrated Parallel Accurate Reservoir Simulator.
  - Mary Wheeler's group, UT-Austin
- Reservoir and Environmental Simulation.
  - models black oil, waterflood, compositions
  - 3D transient flow of multiple phase
- Integrates Existing Simulators.
- Framework simplified development
  - Provides solvers, handling for wells, table lookup.
  - Provides pre/postprocessor, visualization.
- Full IPARS access without Installation.
- IPARS Interfaces:
  - C, FORTRAN, Matlab, Mathematica, and Web.

69

- Show IPARS Demo

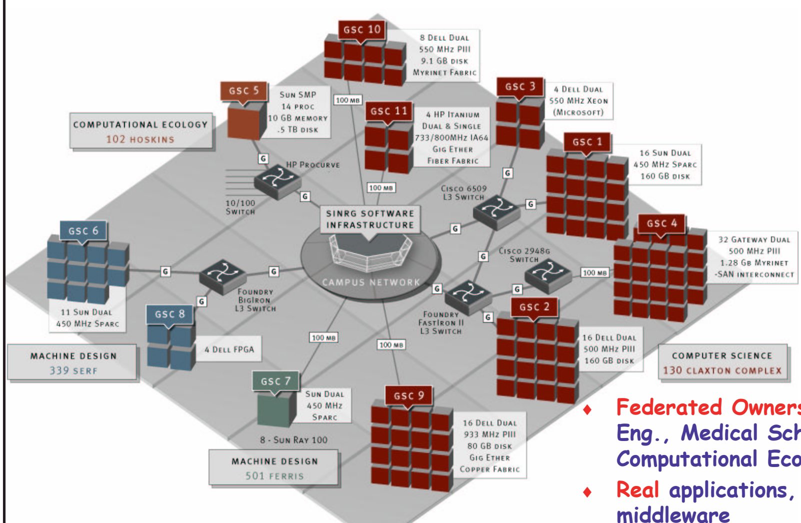
70

Netsolve and SCIRun



SCIRun torso defibrillator application – Chris Johnson, U of Utah

University of Tennessee Deployment: Scalable Intracampus Research Grid: SInRG



The Knoxville Campus has two DS-3 commodity Internet connections and one DS-3 Internet2/Abilene connection. An OC-3 ATM link routes IP traffic between the Knoxville campus, National Transportation Research Center, and Oak Ridge National Laboratory. UT participates in several national networking initiatives including Internet2 (I2), Abilene, the federal Next Generation Internet (NGI) initiative, Southern Universities Research Association (SURA) Regional Information Infrastructure (RII), and Southern Crossroads (SoX).

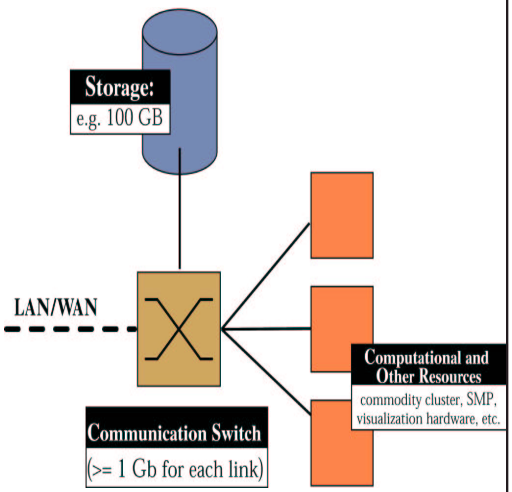
The UT campus consists of a meshed ATM OC-12 being migrated over to switched Gigabit by early 2002.

- Federated Ownership: CS, Chem Eng., Medical School, Computational Ecology, El. Eng.
- Real applications, middleware development, logistical networking

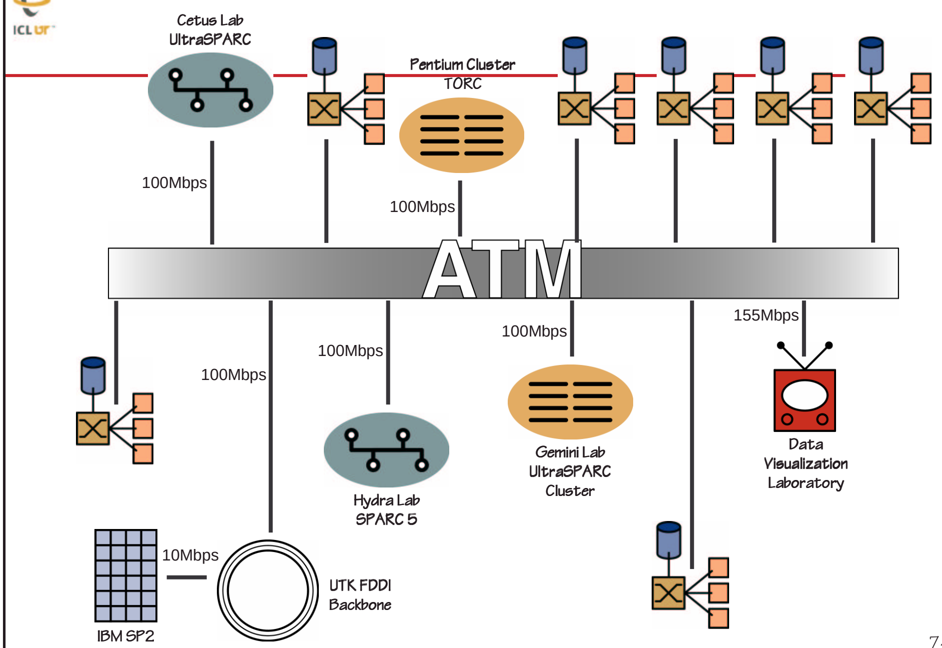
72

Resources: Grid Service Cluster

- **Computation**
  - used to run Grid controlware
  - Committed dynamically to augment other CPUs on Grid
- **Storage**
  - State management
    - data caching
    - migration and fault-tolerance
- **Network**
  - allows dynamic reconfigutation of resources



73



74

SInRG

- SInRG provides a testbed
  - CS grid middleware
  - Computational Science applications
- Many hosts, co-existing in a loose confederation tied together with high-speed links.
- Users have the illusion of a very powerful computer on the desk.
- Spectrum of users

UTK - SInRG

- SInRG constitutes a novel Grid research approach
  - Empirical
  - Vertically integrated and collaborative
  - Both technology and applications driven
  - A *real* research project
- UTK Grid research efforts are drawing national international attention
  - Burgeoning user communities for software artifacts
  - Research and infrastructure funding
  - Persistent installations

The Internet Backplane Protocol (IBP)

- Network middleware which makes distributed network storage available as a flexibly allocated resource.
- Storage buffers exposed to the network.
- A simple mechanism for experimenting with allocation and scheduling

77

IBP's Unit of Storage

- You can think of it as a buffer.
- You can think of it as a "file".
- Append-only semantics.
- Can be used by *anyone* who can talk to the server.

- Seven procedure calls in three categories:
  - Allocation (1)
  - Data transfer (5)
  - Management (1)

Sharing more than the wires.

78

IBP Servers

- Daemons that serve local disk or memory
- Root access not required to set or use.
- Servers can be dynamically added to collection
- Can specify sliding time limits or revocability.
- Encourages resource sharing.
- Data encrypted on the servers

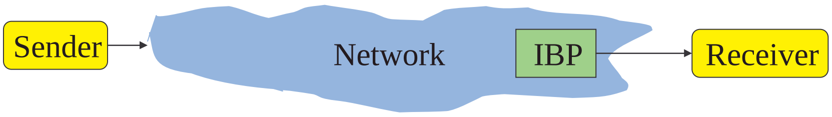
79

Strategy #1: Keep data close to the sender (lazy transmission)

- Network Weather Service sensor data collection.
- Checkpoint servers.
- Tony's example of jet engine.

80

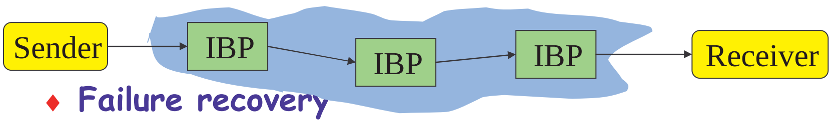
Strategy #2:
Place data close to the receiver



- IBP Mail (MIME encoded attachment, eg video movies)
- Speculative HTTP Transfer.
- NWS Client Data Acquisition.

81

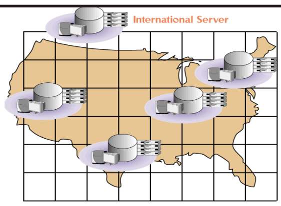
Strategy #3:
Utilize transient storage throughout



- Failure recovery
- Optimum routing
- IBP uses Network Weather Service (NWS) Rich Wolski - UCSB
  - Monitors and extrapolates network metrics
    - Network bandwidth and latency
    - Storage availability
    - CPU load

82

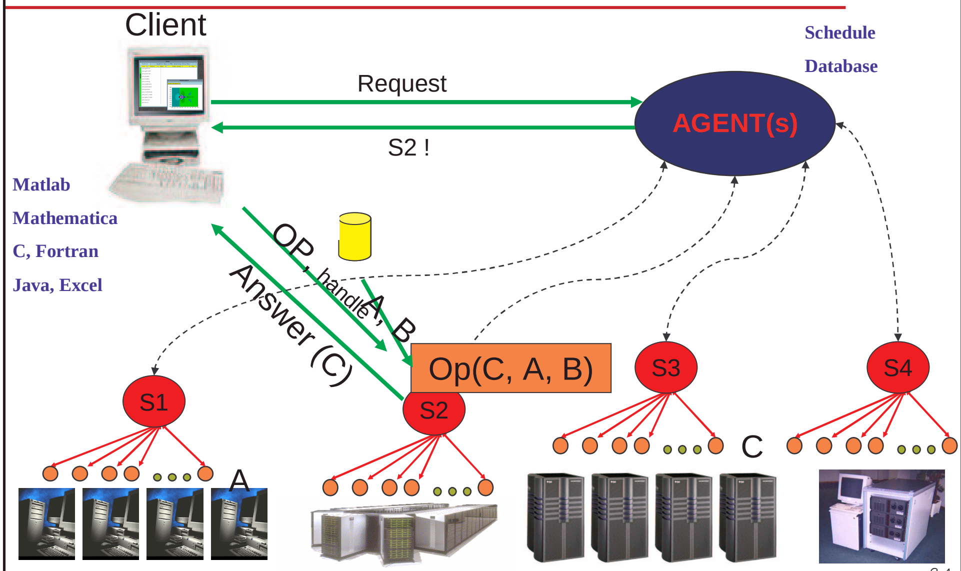
Replicated Services



- Clients access nearby server
- Everyone gets performance
- Local resources implement a global service
- 7-10 Storage Servers
  - 7 IBM Web Cache Mgrs. (72GB disk/900GB tape)
  - ~3 StorageTek systems (700GB disk, tape backup)
- GigaPOP/Campus located
  - Tennessee, MCNC, Indiana, NASA EROS, Hawaii
  - IBM: MREN/iCAIR, Amsterdam
  - Discussions with Canarie (Canada), NTT (Japan)

83

NetSolve: The Big Picture



84

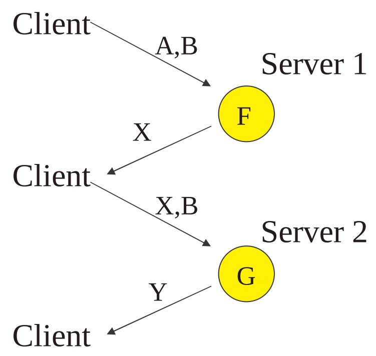
No knowledge of the grid required, RPC like.

State Management in NetSolve

- **The Problem: NetSolve calls are functional**
- **Excessive data transfers**

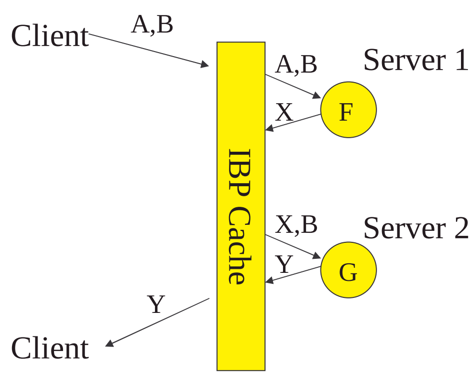
For example:

```
X = F(A, B);
Y = G(X, B);
```

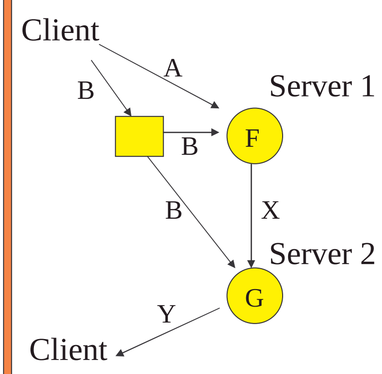


Two Logistical Scheduling Strategies

Caching

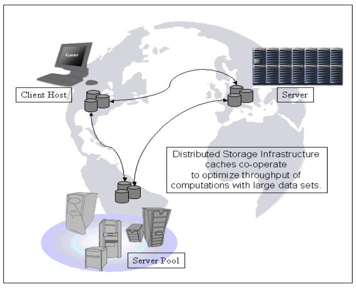


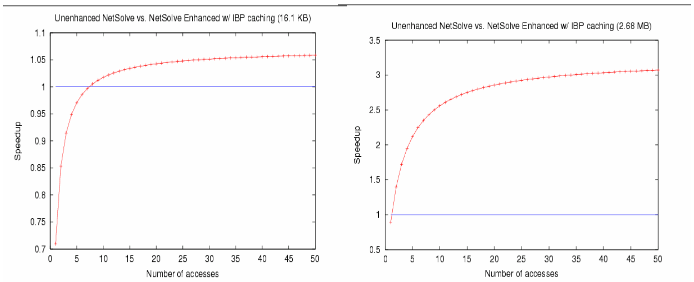
Dependence Flow



Stage Data Close to Server

- Experiments with Unenhanced NetSolve vs. NetSolve w/IBP caching
  Stage datasets close to the point of computation
- In this case client @ UCSD stages datafiles, sparse matrices from Harwell-Boeing collection, to servers at UTK.



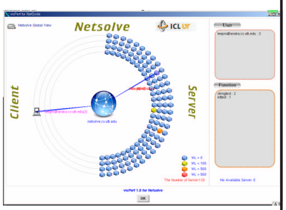


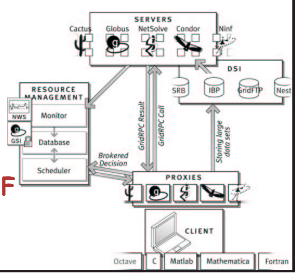
87

NetSolve- Things Not Touched On

- Integration with other NMI tools
  - Globus, Condor, Network Weather Service
- Security
  - Using Kerberos V5 for authentication.
- Separate Server Characteristics
  - Implementing Hardware and Software servers
- Hierarchy of Agents
  - More scalable configuration
- Monitor NetSolve Network
  - Track and monitor usage
- Fault Tolerance
- Local / Global Configurations
- Dynamic Nature of Servers
- Automated Adaptive Algorithm Selection
  - Dynamic determine the best algorithm based on system status and nature of user problem
- NetSolve evolving into GridRPC
  - Being worked on under GGF with joint with NINF

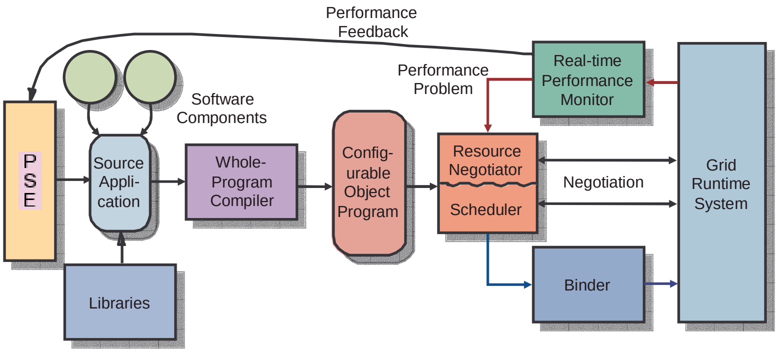
NETWORK WEATHER SERVICE





NSF/NGS
GrADS - GrADSoft Architecture

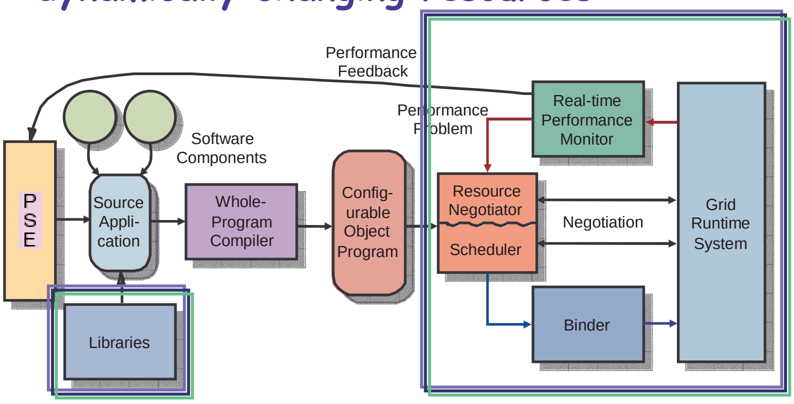
- Goal: reliable performance on dynamically changing resources



PIs: Ken Kennedy, Fran Berman, Andrew Chein, Keith Cooper, JD, Ian Foster, Lennart Johnsson, Dan Reed, Carl Kesselman, John Mellor-Crummey, Linda Torczon & Rich Wolski

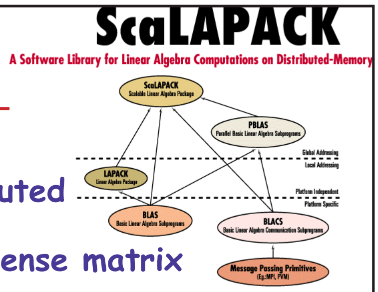
NSF/NGS
GrADS - GrADSoft Architecture

- Goal: reliable performance on dynamically changing resources



PIs: Ken Kennedy, Fran Berman, Andrew Chein, Keith Cooper, JD, Ian Foster, Lennart Johnsson, Dan Reed, Carl Kesselman, John Mellor-Crummey, Linda Torczon & Rich Wolski

ScaLAPACK



- ScaLAPACK is a portable distributed memory numerical library
- Complete numerical library for dense matrix computations
- Designed for distributed parallel computing (MPP & Clusters) using MPI
- One of the first math software packages to do this
- Numerical software that will work on a heterogeneous platform
- Funding from DOE, NSF, and DARPA
- In use today by IBM, HP-Convex, Fujitsu, NEC, Sun, SGI, Cray, NAG, IMSL, …
  - Tailor performance & provide support

91

To Use ScaLAPACK a User Must:

- Download the package and auxiliary packages (like PBLAS, BLAS, BLACS, & MPI) to the machines.
- Write a SPMD program which
  - Sets up the logical 2-D process grid
  - Places the data on the logical process grid
  - Calls the numerical library routine in a SPMD fashion
  - Collects the solution after the library routine finishes
- The user must allocate the processors and decide the number of processes the application will run on
- The user must start the application
  - "mpirun –np *N* user_app"
    - Note: the number of processors is fixed by the user before the run, if problem size changes dynamically …
- Upon completion, return the processors to the pool of resources

92

ScaLAPACK Grid Enabled

- Implement a version of a ScaLAPACK library routine that runs on the Grid.
  - Make use of resources at the user's disposal
  - Provide the best time to solution
  - Proceed without the user's involvement
- Make as few changes as possible to the numerical software.
- Assumption is that the user is already "Grid enabled" and runs a program that contacts the execution environment to determine where the execution should take place.

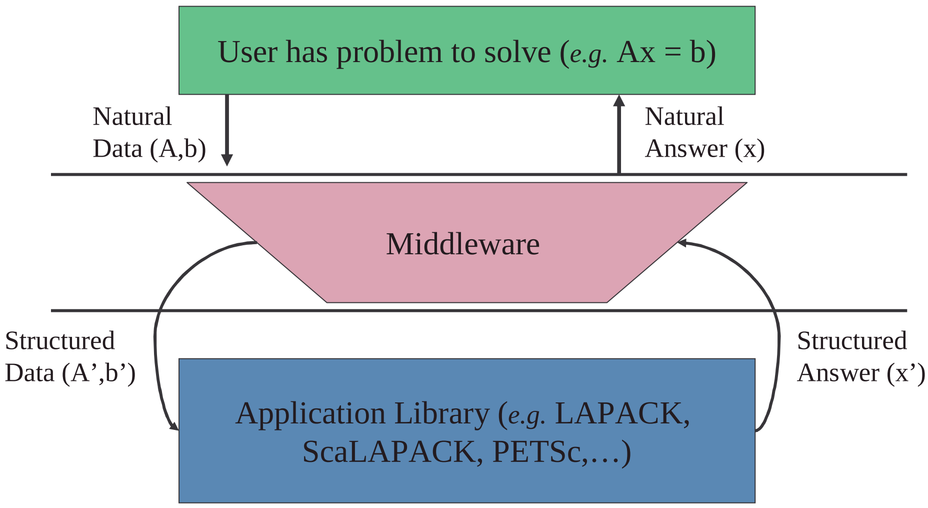
93

GrADS Numerical Library

- Want to relieve the user of some of the tasks
- Make decisions on which machines to use based on the user's problem and the state of the system
  - Determinate machines that can be used
  - Optimize for the best time to solution
  - Distribute the data on the processors and collections of results
  - Start the SPMD library routine on all the platforms
  - Check to see if the computation is proceeding as planned
    - If not perhaps migrate application

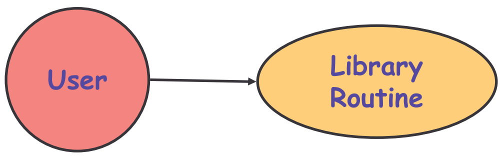
94

Big Picture…



95

GrADS Library Sequence

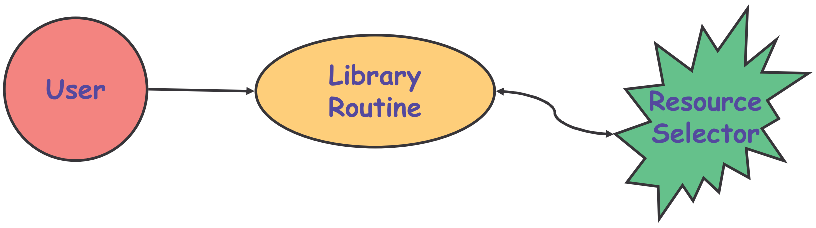


- Has "crafted code" to make things work correctly and together.

Assumptions:
Autopilot Manager has been started
and
Globus is there.

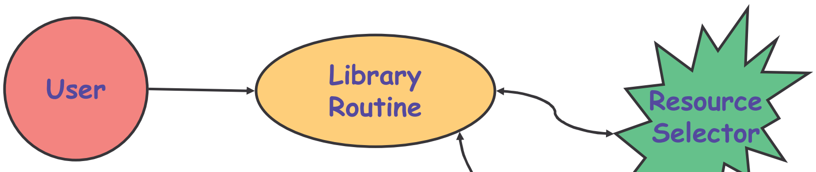
96

Resource Selector

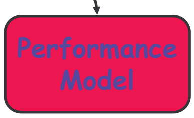


- Uses MDS and NWS to build an array of values for the machines that are available for the user.
  - 2 matrices (bw,lat) 2 arrays (cpu, memory available)
  - Matrix information is clique based
- On return from RS, Crafted Code filters information to use only machines that have the necessary software and are really eligible to be used.

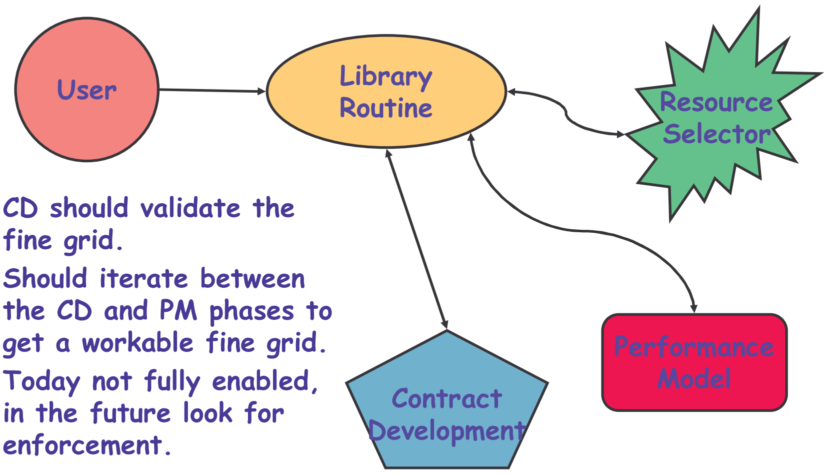
Performance Model



- Performance Model uses the information generated in the RS to decide on the fine grid.
  - Pick a machine that is closest to every other machine in the collection.
  - If not enough memory, adds machines until it can solve problem.
  - Cost model is run on this set.
  - Process adds a machine to group and reruns cost model.
  - If "better", iterate last step, if not stop.



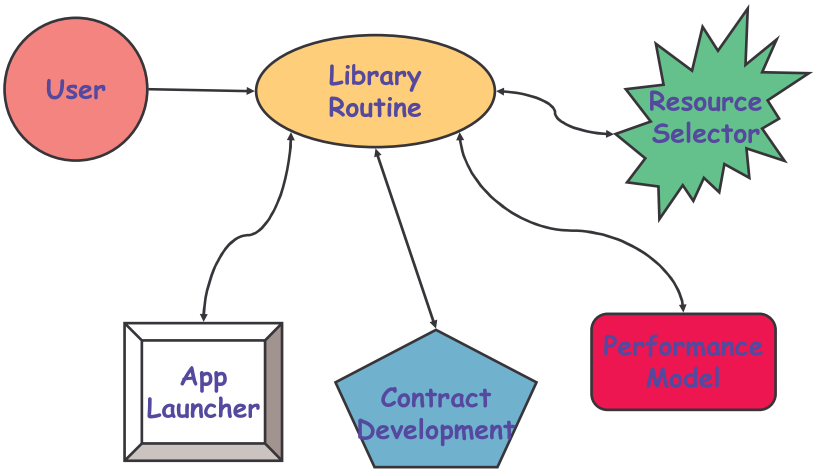
Contract Development



- CD should validate the fine grid.
- Should iterate between the CD and PM phases to get a workable fine grid.
- Today not fully enabled, in the future look for enforcement.

99

Application Launcher

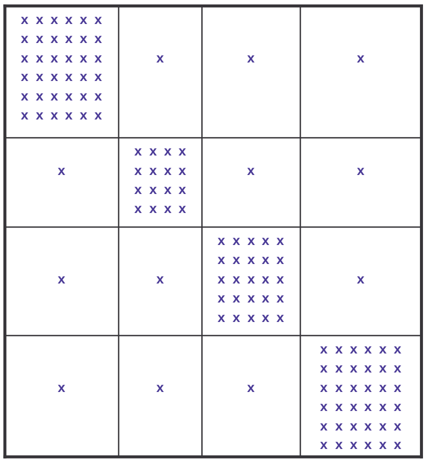


"mpirun -machinefile -globusrsl fine_grid grid_linear_solve"

100

Resource Selector Input

- Clique based
  - 2 @ UT, UCSD, UIUC
    - Part of the MacroGrid
  - Full at the cluster level and the connections (clique leaders)
  - Bandwidth and Latency information looks like this.
  - Linear arrays for CPU and Memory
- Matrix of values are filled out to generate a complete, dense, matrix of values.
- At this point have a workable coarse grid.
  - Know what is available, the connections, and the power of the machines



101

ScaLAPACK Performance Model

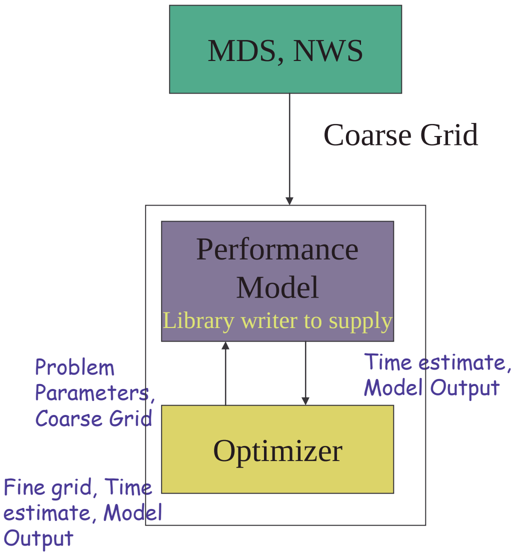
$$T(n,p) = C_f t_f + C_v t_v + C_m t_m$$

- $C_f = \frac{2n^3}{3p}$ Total number of floating-point operations per processor
- $C_v = (3 + \frac{1}{4}\log_2 p)\frac{n^2}{\sqrt{p}}$ Total number of data items communicated per processor
- $C_m = n(6 + \log_2 p)$ Total number of messages
- t_f Time per floating point operation
- t_v Time per data item communicated
- t_m Time per message

102

Resource Selector/Performance Modeler

- Refines the course grid by determining the process set that will provide the best time to solution.
- This is based on dynamic information from the grid and the routines performance model.
- The PM does a simulation of the actual application using the information from the RS.
  - It literally runs the program without doing the computation or data movement.
- There is no backtracking in the Optimizer.
  - This is an area for enhancement and experimentation.
- Simulated annealing used as well



103

Performance Model Validation

	Opus14	Opus13	Opus16	Opus15	Torc4	Torc6	Torc7
mem(MB)	215	214	227	215	233	479	479
speed	270	270	270	270	330	330	330
load	1	0.99	1	0.99	1	1.04	0.87

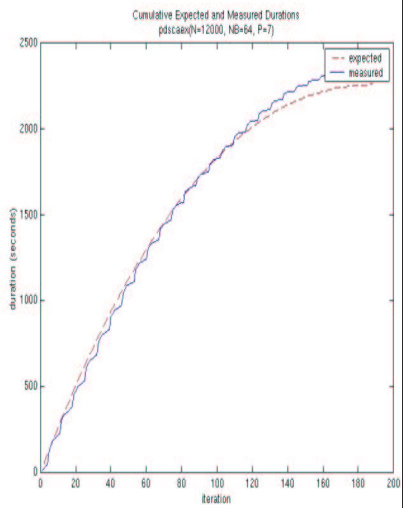
Speed = 60% of the peak

Latency	Opus14	Opus13	Opus16	Opus15	Torc4	Torc6	Torc7
Opus14	-1	0.24	0.29	0.26	83.78	83.78	83.78
Opus13	0.24	-1	0.24	0.23	83.78	83.78	83.78
Opus16	0.29	0.24	-1	0.23	83.78	83.78	83.78
Opus15	0.26	0.23	0.23	-1	83.78	83.78	83.78
Torc4	83.78	83.78	83.78	83.78	-1	0.31	0.31
Torc6	83.78	83.78	83.78	83.78	0.31	-1	0.31
Torc7	83.78	83.78	83.78	83.78	0.31	0.31	-1

Latency in msec

Bandwidth	Opus14	Opus13	Opus16	Opus15	Torc4	Torc6	Torc7
Opus14	-1	248.83	247.31	246.38	2.83	2.83	2.83
Opus13	248.83	-1	244.54	240.94	2.83	2.83	2.83
Opus16	247.31	244.54	-1	247.54	2.83	2.83	2.83
Opus15	246.38	240.94	247.54	-1	2.83	2.83	2.83
Torc4	2.83	2.83	2.83	2.83	-1	81.96	56.47
Torc6	2.83	2.83	2.83	2.83	81.96	-1	50.9
Torc7	2.83	2.83	2.83	2.83	56.47	50.9	-1

Bandwidth in Mb/s



This is for a refined grid

104

Experimental Hardware / Software Grid

MacroGrid Testbed	TORC	CYPHER	OPUS
Type	Cluster 8 Dual Pentium III	Cluster 16 Dual Pentium III	Cluster 8 Pentium II
OS	Red Hat Linux 2.2.15 SMP	Debian Linux 2.2.17 SMP	Red Hat Linux 2.2.16
Memory	512 MB	512 MB	128 or 256 MB
CPU speed	550 MHz	500 MHz	265 – 448 MHz
Network	Fast Ethernet (100 Mbit/s) (3Com 3C905B) and switch (BayStack 350T) with 16 ports	Gigabit Ethernet (SK-9843) and switch (Foundry FastIron II) with 24 ports	Myrinet (LANai 4.3) with 16 ports each

- Globus version 1.1.3
- Autopilot version 2.3
- NWS version 2.0.pre2
- MPICH-G version 1.1.2
- ScaLAPACK version 1.6
- ATLAS/BLAS version 3.0.2
- BLACS version 1.1
- PAPI version 1.1.5
- GrADS' "Crafted code"

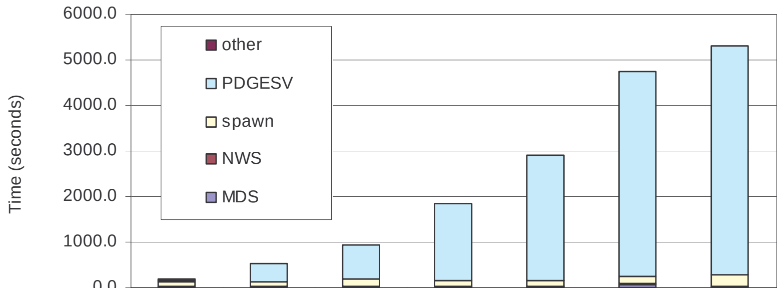
Independent components being put together and interacting

105

PDGESV Time Breakdown

ScaLAPACK - PDGESV - Using collapsed NWS query from UCSB

42 machine available, using mainly torc, cypher, msc clusters at UTK [Jan 2002]

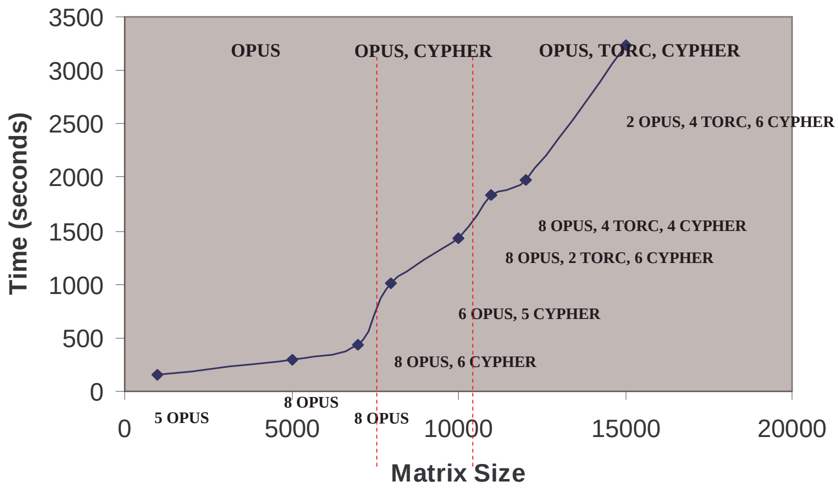


	5000-10	10000-12	15000-14	20000-14	25000-18	30000-18	35000-27
other	4.7	5.3	1.0	2.3	7.6	1.1	4.7
PDGESV	58.7	394.7	749.4	1686.7	2747.1	4472.7	5020.4
spawn	92.2	105.9	154.1	124.7	135.6	181.0	264.5
NWS	5.5	7.4	12.3	12.0	4.2	10.2	8.5
MDS	13.0	11.0	10.0	11.0	14.0	73.0	12.0

Matrix Size - Nproc

106

ScaLAPACK across 3 Clusters



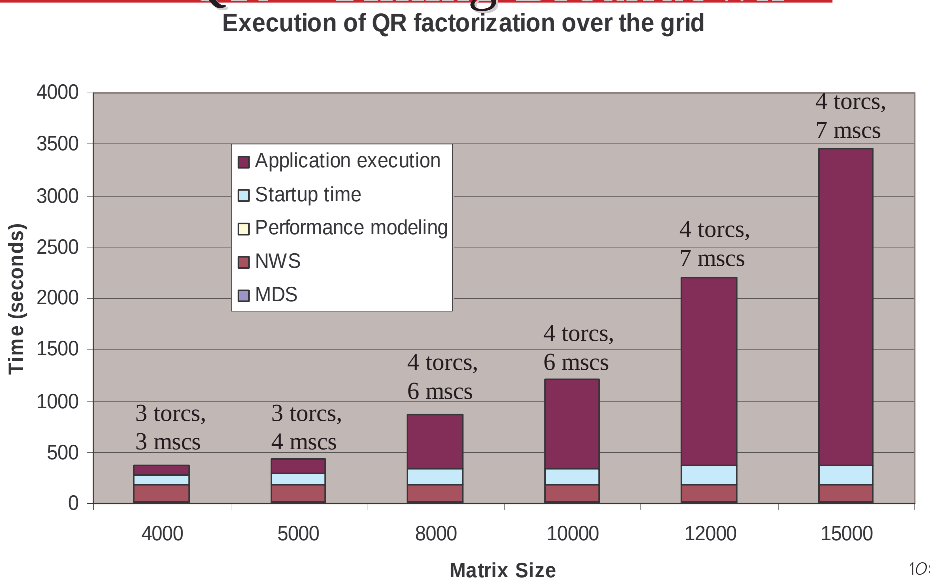
107

Largest Problem Solved

- Matrix of size 30,000
  - 7.2 GB for the data
  - 32 processors to choose from UIUC and UT
    - Not all machines have 512 MBs, some little as 128 MBs
  - PM chose 17 machines in 2 clusters from UT
  - Computation took 84 minutes
    - 3.6 Gflop/s total
    - 210 Mflop/s per processor
    - ScaLAPACK on a cluster of 17 processors would get about 50% of peak
    - Processors are 500 MHz or 500 Mflop/s peak
    - For this grid computation 20% less than ScaLAPACK

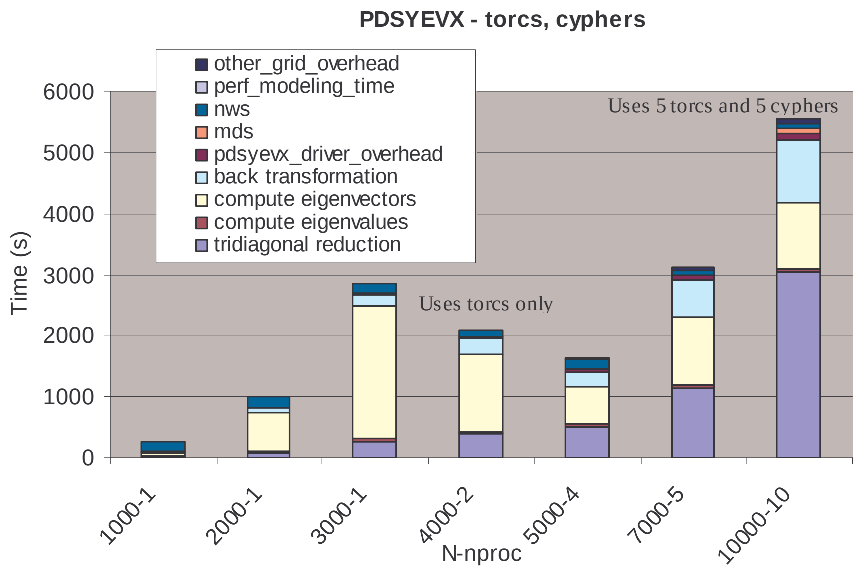
108

QR – Timing Breakdown



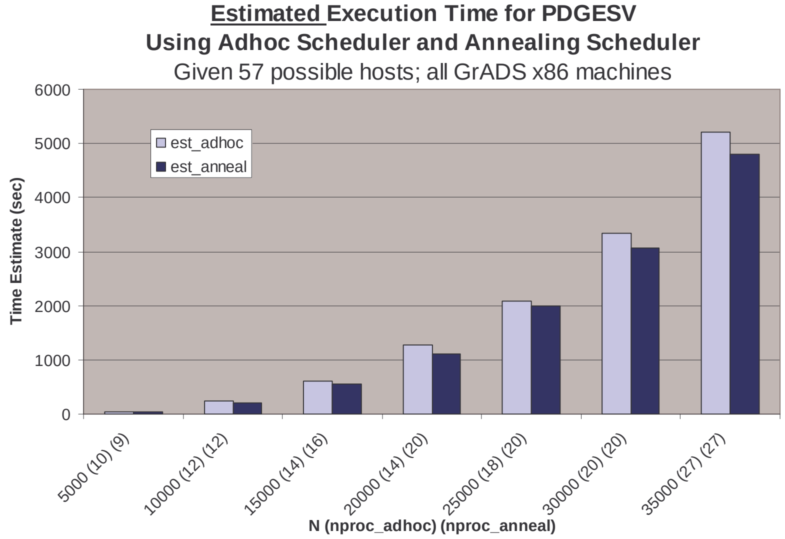
109

PDSYEVX – Timing Breakdown



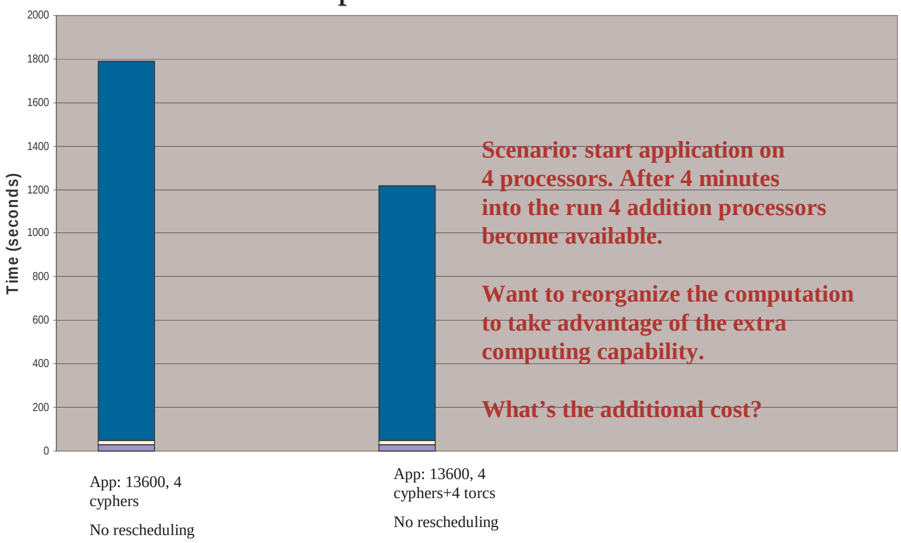
110

Adhoc vs Annealing Scheduling



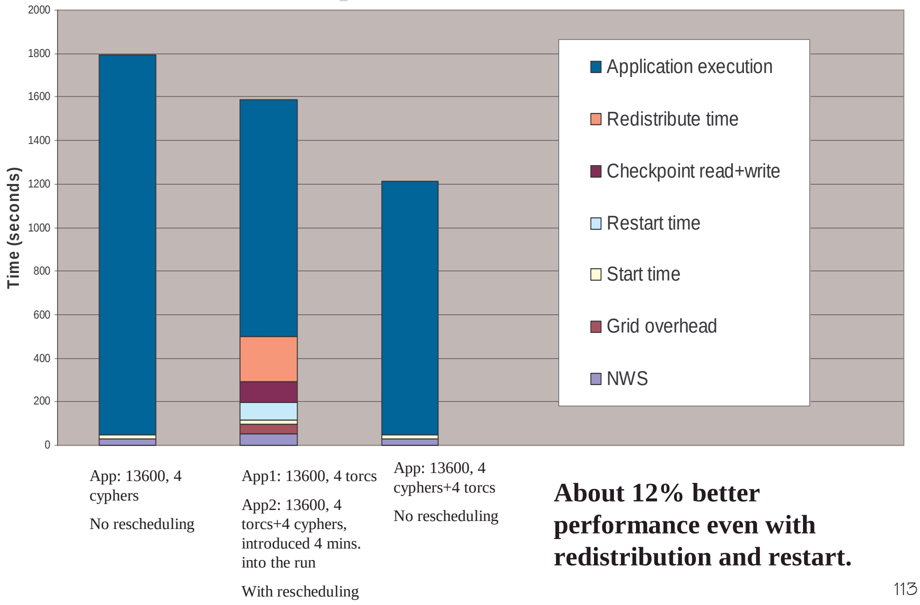
111

Rescheduling/Redistribution Experimental Results



112

Rescheduling/Redistribution Experimental Results



About 12% better performance even with redistribution and restart.

113

Major Challenge - Adaptivity

- These characteristics have major implications for applications that require performance guarantees.
- **Adaptivity** is a key so applications can function appropriately...
  - as resource utilization and availability change,
  - as processors and networks fail,
  - as old components are retired,
  - as new systems are added, and
  - as both software and hardware on existing systems are updated and modified.

114

Conclusion

- Exciting time to be in scientific computing
- Grid computing is here
- The Grid offers tremendous opportunities for collaboration
- Important to develop algorithms and software that will work effectively in this environment

115

Collaborators

- GrADS
  - Sathish Vadhiyar, UTK
  - Asim YarKhan, UTK
  - Ken Kennedy, Fran Berman, Andrew Chein, Keith Cooper, Ian Foster, Carl Kesselman, Lennart Johnsson, Dan Reed, Linda Torczon, & Rich Wolski
- IBP
  - Micah Beck, UTK
  - Jim Plank, UTK
  - Rich Wolski, UCSB
  - Fran Berman,UCSD
  - Henri Casanova, UCSD
- NetSolve
  - Sudesh Agrawal, UTK
  - Henri Casanova, UCSD
  - Keith Seymour, UTK
  - Sathish Vadhiyar, UTK

- Software Availability
  - NetSolve
    - icl.cs.utk.edu/netsolve/
  - LFC
    - 5 drivers from ScaLAPACK around the end of summer
    - Next look at iterative solvers

Many opportunities within the group at Tennessee

116

Major Challenge - Adaptivity

- These characteristics have major implications for applications that require performance guarantees.
- **Adaptivity** is a key so applications can function appropriately...
  - as resource utilization and availability change,
  - as processors and networks fail,
  - as old components are retired,
  - as new systems are added, and
  - as both software and hardware on existing systems are updated and modified.

117

Futures for Numerical Algorithms and Software on Clusters and Grids

- Retargetable Libraries - Numerical software will be adaptive, exploratory, and intelligent
- Determinism in numerical computing will be gone.
  - After all, its not reasonable to ask for exactness in numerical computations.
  - Auditability of the computation, reproducibility at a cost
- Importance of floating point arithmetic will be undiminished.
  - 16, 32, 64, 128 bits and beyond.
- Reproducibility, fault tolerance, and auditability
- Adaptivity is a key so applications can effectively use the resources.

118

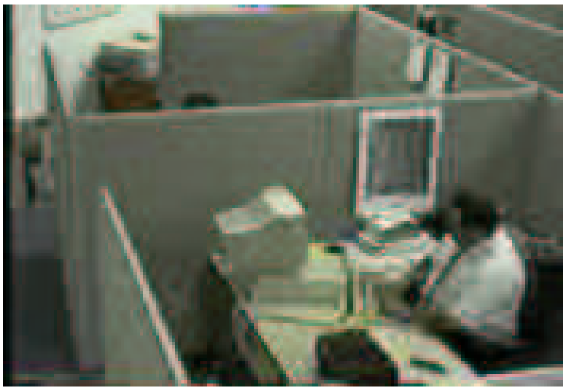
Conclusion

- Exciting time to be in scientific computing
- Network computing is here
- The Grid offers tremendous opportunities for collaboration
- Important to develop algorithms and software that will work effectively in this environment

119

Vinny's Bad Day

- Hopefully the Grid will simplify computer use not make it more difficult.



120