CS514: Intermediate Course in Computer Systems

Lecture 39: April 25, 2003

"*Grid Computing Systems*"

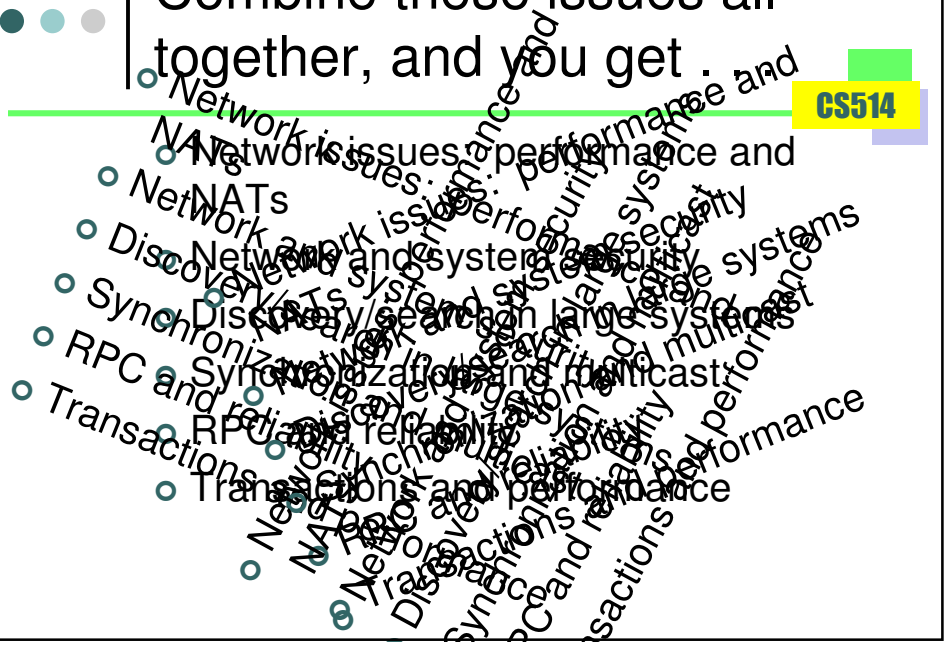
We've looked at some hard issues . . .

CS514

- Network issues: performance and NATs
- Network and system security
- Discovery/search in large systems
- Synchronization and multicast
- RPC and reliability
- Transactions and performance

Combine these issues all together, and you get . . .

CS514



Combine these issues all together, and you get . . .

CS514

Grid Computing!

What is grid computing?

CS514

- Means lots of things to different people
- *"Resource sharing and coordinated problem solving in dynamic, multi-institutional virtual organizations"*
- *"Coordinated resource sharing in a distributed dynamic heterogeneous environment to enable efficient large-scale problem solving"*

What is grid computing?

CS514

- *"Enable communities ("virtual organizations") to share geographically distributed resources as they pursue common goals—in the absence of central control, omniscience, trust relationships"*

Common elements in definitions:

CS514

- *Dynamic*
- *Resource sharing*
- *Across organizations*
- Where resources can be:
  - Raw materials of computing
    - CPU, storage
  - Execution of some specific function or service
    - A library routine, specific data, etc.

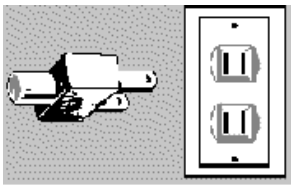
Sometimes compared to power grid

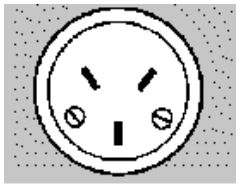
CS514

- This is where the “grid” in “grid computing” comes from
- Plug into the computational grid as easily as we plug into power grid today
- Power grid has many sources and consumers of power, gets traded, moved around, etc.
- Even so, I’m not sure it is a good analogy

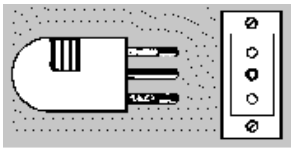
Even a standardization comparison

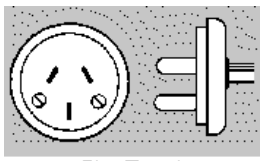
CS514


Plug Type A


Plug Type H


Plug Type G


Plug Type L


Plug Type I

Courtesy mark.baker@computer.org

There are some grand visions…

CS514

- Provider computers advertise their own resources to discovery systems
  - Even your desktop!
- Consumer computers search the discovery systems for resources
- Resources are dynamically allocated
- Resources are brokered (traded or paid for)
- Programming tools make this all easy
- Secure, efficient, fault tolerant, etc. etc.

Kind-of like web services on steroids!

CS514

- And in fact, some web services standards are becoming the basis for grid computing standards . . .
  - WSDL has grid service description tag
    - Web Services Definition Language
  - SOAP for transport
  - UDDI registry and WSIL for service location
    - IBM, Microsoft, HP, SAP registries
    - Web Services Inspection Language
    - Universal Description, Discovery, and Integration

Some grid computing history

CS514

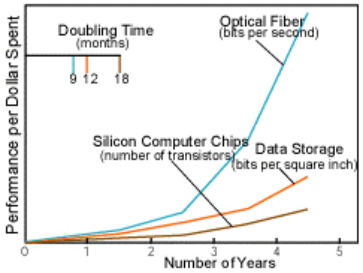
- Early 90s
  - Gigabit testbeds, metacomputing
- Mid to late 90s
  - Early experiments (e.g., I-WAY), academic software projects, application experiments
- 2002
  - Dozens of application communities & projects in scientific and technical computing
  - Major infrastructure deployments
  - De facto standard technology: Globus Toolkit™
  - Growing industrial interest
  - Global Grid Forum: ~1000 people, 30+ countries

from Globus/IBM tutorial

Why now? Network Exponentials

CS514

- 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.

from Globus/IBM tutorial

Or: Hype Exponentials!?!?

RED HERRING

TOP TEN TRENDS 2001

DISTRIBUTE THE WEALTH

Distributed computing initiatives.

GLOBUS PROJECT

Globus is a research and development project focused on enabling the application of grid computing concepts to scientific and engineering com-

Computing power on tap

Jun 21st 2001

From The Economist print edition

Economist.com

In the first of two articles, we look at the most ambitious attempt yet to combine millions of computers seamlessly around the world—to make processing power available on demand anywhere, rather like electrical power.

I.B.M. Making a Commitment to Next Phase of the Internet

By STEVE LOHR

The New York Times

I.B.M. is announcing today a new initiative to support and exploit a technology known as grid computing, which the company and much of the computer research community say is the

Merrill Lynch

Globus Grid Computing—the Next Internet

by John Roy/Steve Milunovich

The Internet was first a network and is now a communications platform. The next evolutionary step could be to a platform for distributed computing. This ability to manage applications and share data over the network is called "grid computing."

Grid Computing

By M. Mitchell Waldrop

May 2002

AN MIT ENTERPRISE TECHNOLOGY REVIEW

Emerging Technologies and Their Impact

Hook enough computers together and what do you get? A new kind of utility that offers supercomputer processing on tap.

Is Internet history about to repeat itself?

from Globus/IBM tutorial

What we'll look at

CS514

- Two very different current successes
  - seti@home (and similar efforts)
  - Cornell etc. Adaptive Software Project
- And a number of different grid computing middleware projects
  - NetSolve
  - Globus/IBM

seti@home goals

CS514

- Look for signs of extra-terrestrial life
- By harnessing the spare power of hundreds of thousands of Internet-connected computers
- To do signal processing on space signals
  - Piggybacked on data collected from Arecibo radio telescope, Puerto Rico

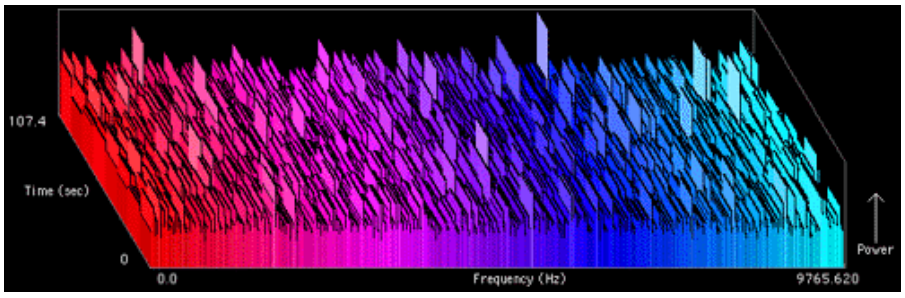
Brilliant concept for user incentive

CS514

- "Each participant will have the slight but captivating possibility that his or her computer will detect the faint murmur of a civilization beyond Earth."
- Cool screen saver!
- User rankings on web site
  - Most hours logged

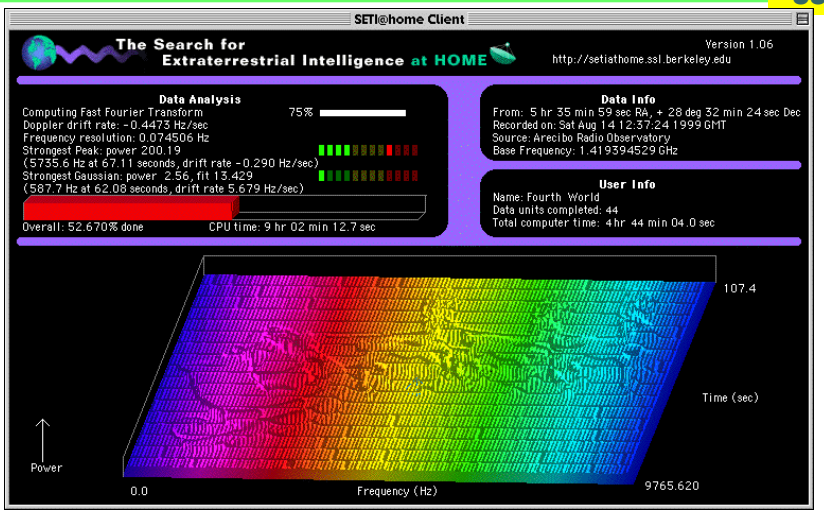
Screen saver shows colorful freq-time-power graph

CS514

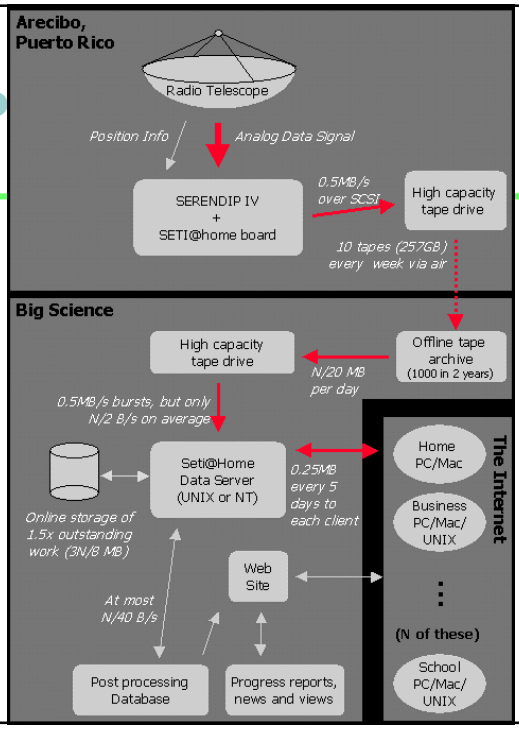


Including this intriguing one around Christmas 1999

CS514



CS514



Impressive results

CS514

- Over 4 million users logging around 1.5 million CPU hours
- Why does this work?
- seti@home is “embarrassingly parallel”
- Data processing can be broken out into individual tasks
  - no inter-processor communications
  - no fault-tolerance issues

Common class of scientific computing problem

CS514

- Globus folks call this a “Data Grid”
- Examples include
  - High-energy physics
  - Biotech: screen molecules for new drugs
    - “patriotgrid” in the case of anthrax treatment (United Devices)
  - Math problems, code breaking

Adaptive Software Project

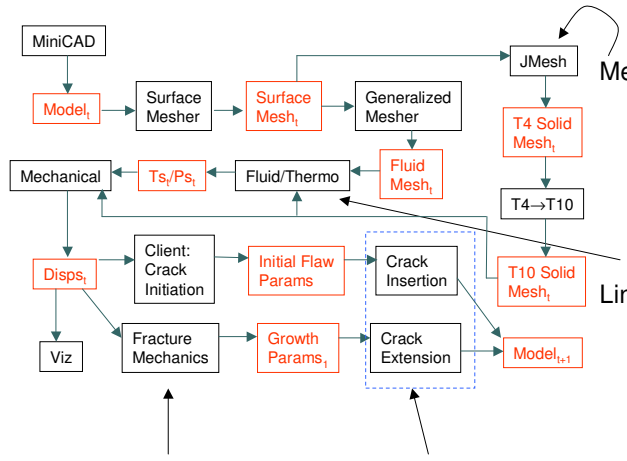
CS514

- Cornell, Mississippi State, Ohio State, William and Mary, Clark Atlanta
- Inter-disciplinary team to simulate fracture mechanics in an internal combustion engine

Slides courtesy of Paul Stodghill

ASP Grid Simulation Example

CS514



Mesh generation: William&Mary

- Problems require meshes with O(10^6) elements
- Time: 2-3 hours on // mc.s

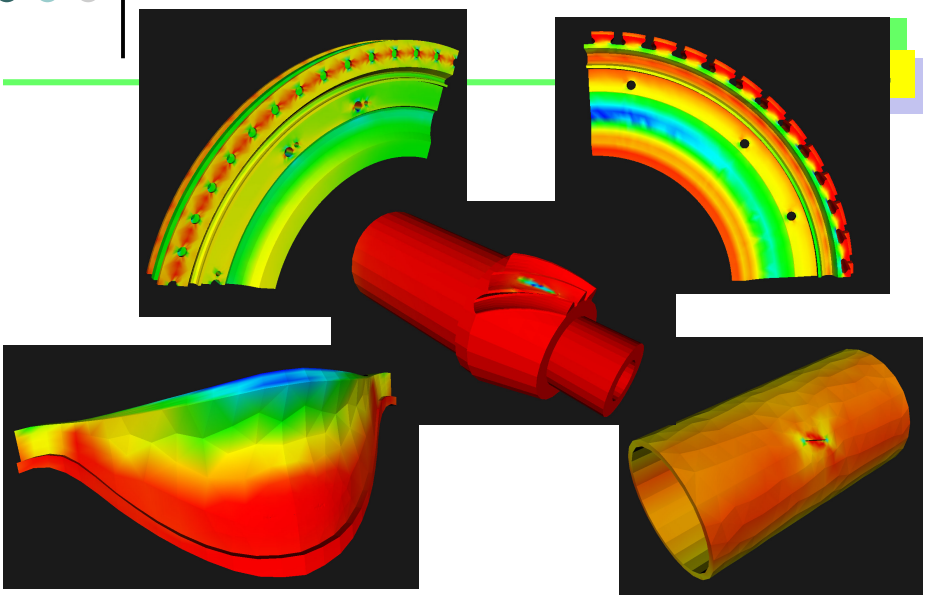
Linear system solvers: MSU

- Large sparse systems Ax =b where A has O(10^7) elements
- Time: ~1 hour on // mc.s

Fracture specialist: Cornell

Computational Geometry: UA@Birm.

Sample solutions from test-bed



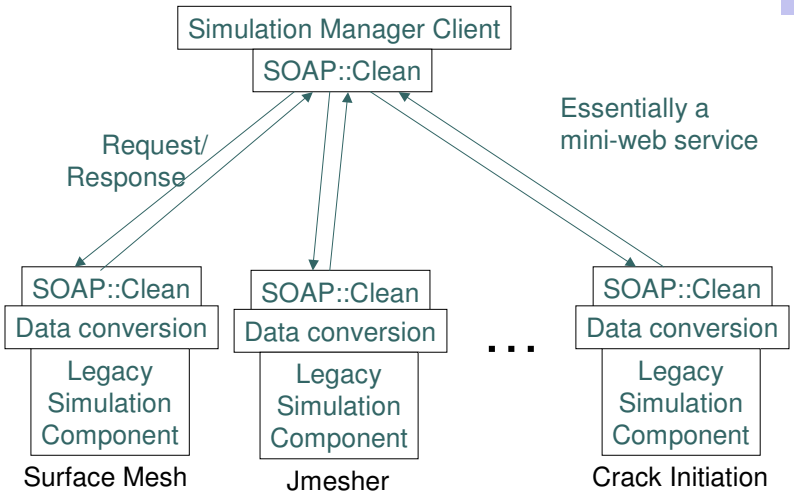
Motivation for “grid” based solution

CS514

- Primarily software engineering
  - ***Not*** need for more computing resources
- Each component written by different groups for different platforms
- Software engineering nightmare to combine into a single monolithic program

ASP Grid architecture

CS514



SOAP::Clean

CS514

A framework for distributed computing using Web Services

- Web Services = XML, SOAP, UDDI, …
- SOAP::Clean,
  - Allows existing applications to be quickly deployed as Web Service servers
  - Provides a client program that makes Web Services appear as command line programs

- Legacy application + dozen lines of Perl + SOAP::Clean::Server = Web Service
- Distributed Computations
  - Makefile + wsdl-client.pl, or
  - Perl code + SOAP::Clean::Client
  - C# + Visual Studio.NET
  - …

http://www.asp.cornell.edu/SOAP-Clean/

Performance using SOAP::Clean

CS514

Component	Wght	Request size (bytes)	Response size (bytes)	Base Running (seconds)	Remote Running (seconds)	Overhead (seconds)	(%)
Surface Mesh	2	135,074	377,992	63.69	74.85	11.16	17.52%
Jmesher	2	512,386	1,334,431	107.26	116.51	9.25	8.62%
T4→T10	2	1,468,764	4,454,170	70.01	81.32	11.31	16.15%
Generalized Mesh	2	513,015	2,824,138	39.17	30.63	-8.54	-21.80%
Fluid/Thermal	2	7,416,514	1,038,255	1584.36	1603.39	19.03	1.20%
Crack Growth	1	135,017	141,467	0.42	7.72	7.30	1738.10%
Fracture Mechani	1	6,621,903	3,201	36.43	56.59	20.16	55.34%
Crack Initiation	1	141,299	180,465	0.31	8.65	8.34	2690.32%
Weighted Total		26,989,725	20,383,105	3766.14	3886.36	120.22	3.19%

Preliminary results

- Small problem size
- Room for additional optimizations

Base Running = time to directly run components, without the use of Web Services

Remote Running = client at UA@Birm., server at CUCS

Conclusions

- 3.19% overhead is fantastic for an interstate computation.
- Overhead will decrease as the problem size increases.
  - Data is *O(n)*.
  - Running times are > *O(n)*.

Advantages of grid-based simulation

CS514

- Simplifies project management
  - no need for everyone to agree on a common implementation language or hardware platform
  - need agree only on data exchange format (XML/SOAP)
- Avoids software maintenance problem
  - each project site maintains its own code but makes it available to other partners as a web service
- In future
  - computations scheduled for execution wherever there are free resources
  - computations may even migrate during execution where more resources become available

Implications for software

CS514

- Software needs to be adaptive
  - adaptation for efficiency
    - application must be optimized dynamically when computation starts on or migrates to a new platform
  - adaptation for survival
    - adapt gracefully to processor and link failures: self-healing software

➔ Software must become more intelligent

Two examples of working grid computation

CS514

- One works because “embarrassingly” easy to parallelize
- One works because component teams work closely together to define interactions
- Neither really satisfies definition of “dynamic resource sharing across organizations”

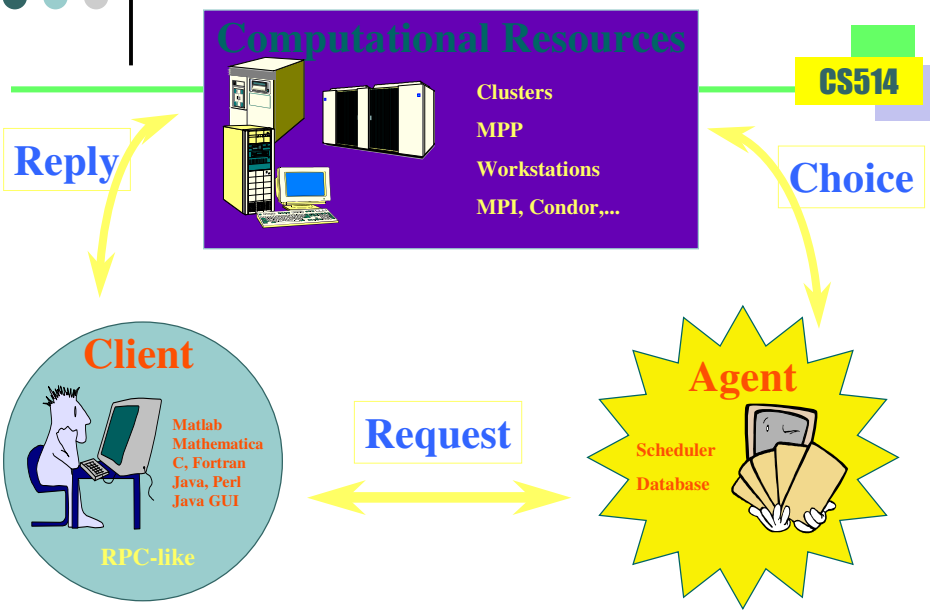
Netsolve

CS514

- Jack Dongarra et.al. (many et.al.)
  - Univ of Tennessee
- Motivated by Problem Solving Environments (PSE)
  - MatLab, Mathematica, etc.
- Goal more to find resources specific to a problem than to find massive computing resources
  - Certain computation libraries, computations that require large intermediate storage

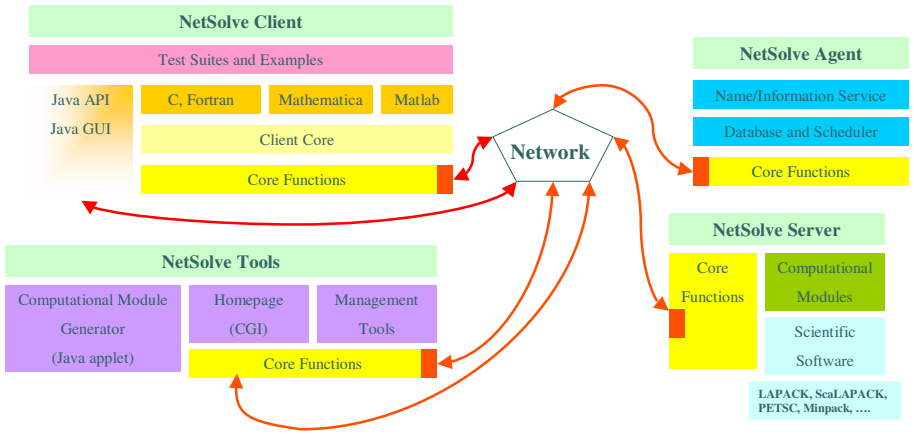
Some slides courtesy Casanova and Dongarra

NetSolve - The big picture



NetSolve components

CS514



Netsolve Agent

CS514

- Resource Scheduling
- Static and dynamic server information
  - 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.

Netsolve Client

CS514

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

Netsolve programming

CS514

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

```
netsl_farm("i=0,10","monte_carlo",
            ns_int_array(seeds,"$i"),
            ns_ptr_array(input,"$i"),
            ns_ptr_array(output,"$i"));
```

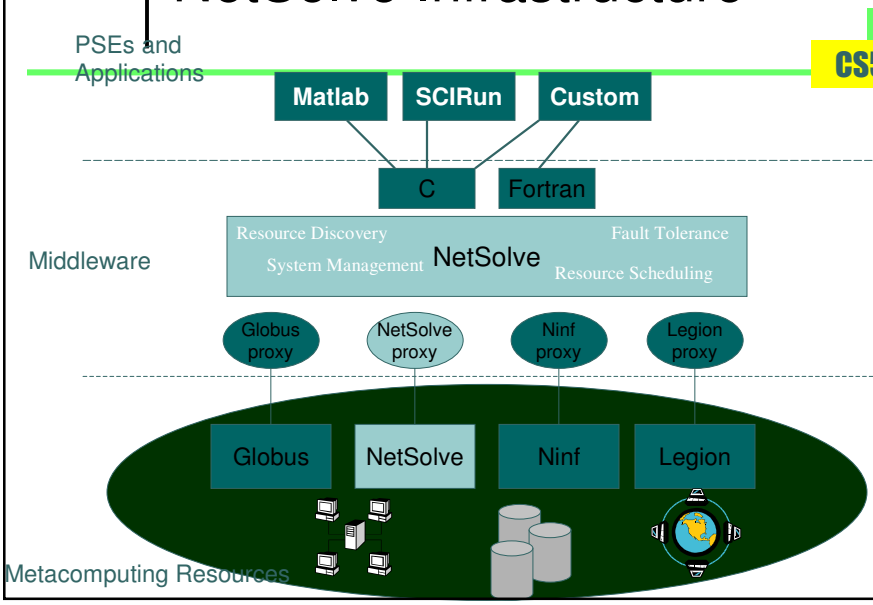
Netsolve

CS514

- Not meant to scale to massive infrastructures
- Uses Kerberos for security
- Plugs in to other grid middleware

NetSolve Infrastructure

CS514



Why didn't Adaptive Software Project use Netsolve???

CS514

- Anybody wanna guess?

Why didn't Adaptive Software Project use Netsolve???

CS514

- Anybody wanna guess?
- Doesn't run on windows

The Globus Project (www.globus.org)

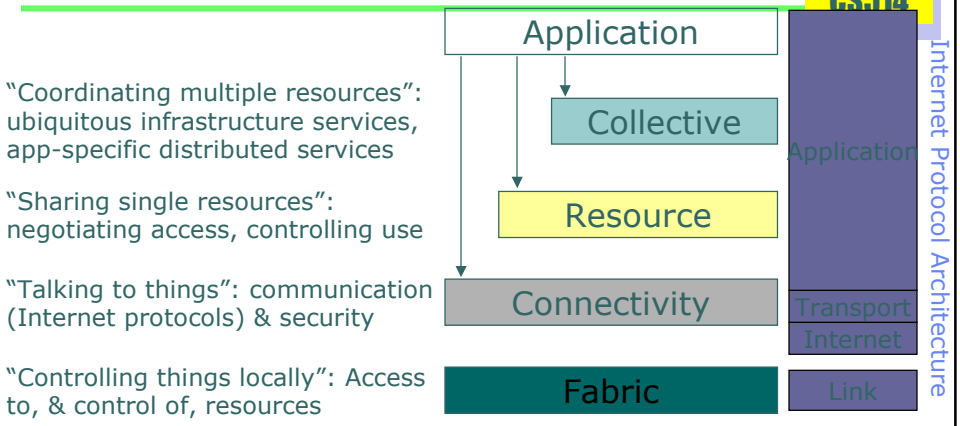
CS514

- Ian Foster, Carl Kesselman
  - and about 60 people!
- Argonne National Lab, Univ of Chicago, USC-ISI
- Globus Toolkit v2 (GT2)
- Many projects (a few dozen)
- Picked up by IBM
  - Which is cramming Globus into a web services framework
- Has won mindshare, now it just needs to work!

Slides courtesy "Introduction to Grid Computing and Globus Toolkit", only 220 pages long…

Layered Grid Architecture

CS514



"The Anatomy of the Grid: Enabling Scalable Virtual Organizations", Foster, Kesselman, Tuecke, Intl Journal of High Performance Computing Applications, 15(3), 2001.

Key Protocols

CS514

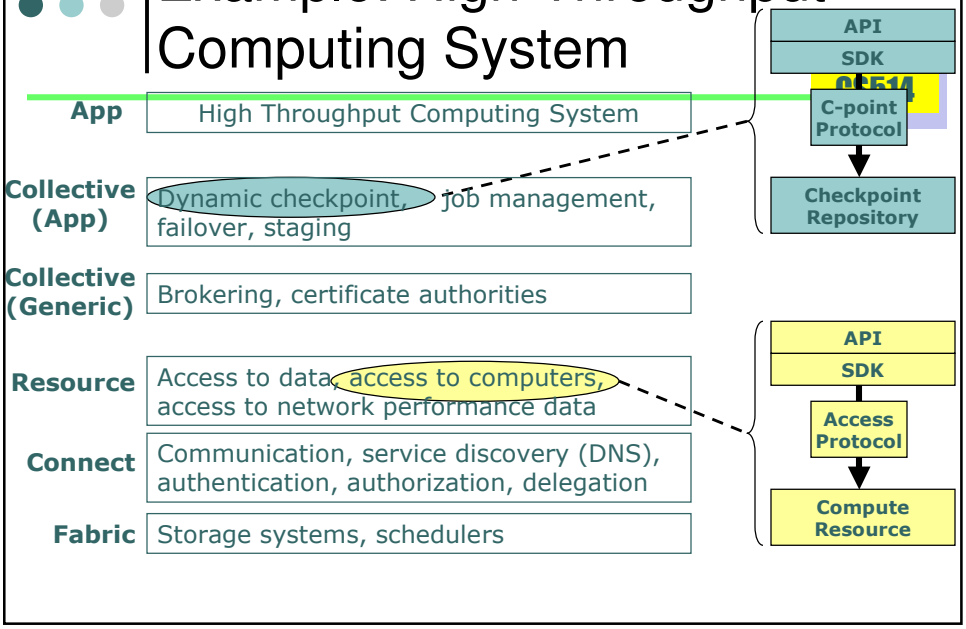
- The Globus Toolkit™ v2 (GT2) centers around four key protocols
  - Connectivity layer:
    - *Security*: Grid Security Infrastructure (GSI)
  - Resource layer:
    - *Resource Management*: Grid Resource Allocation Management (GRAM)
    - *Information*: Grid Resource Information Protocol (GRIP/LDAP)
    - *Data Transfer*: Grid File Transfer Protocol (GridFTP)
- Also key collective layer protocols
  - Monitoring & Discovery, Replication, etc.

GT-Based Grid Tools & Applications

CS514

- Data Grids
  - Distributed management of large quantities of data—physics, astronomy, engineering
- High-throughput computing
  - Coordinated use of many computers
- Collaborative environments
  - Authentication, resource discovery, and resource access
- Portals
  - Thin client access to remote resources & services
- And combinations of the above

Example: High-Throughput Computing System



Example: Data Grid Architecture

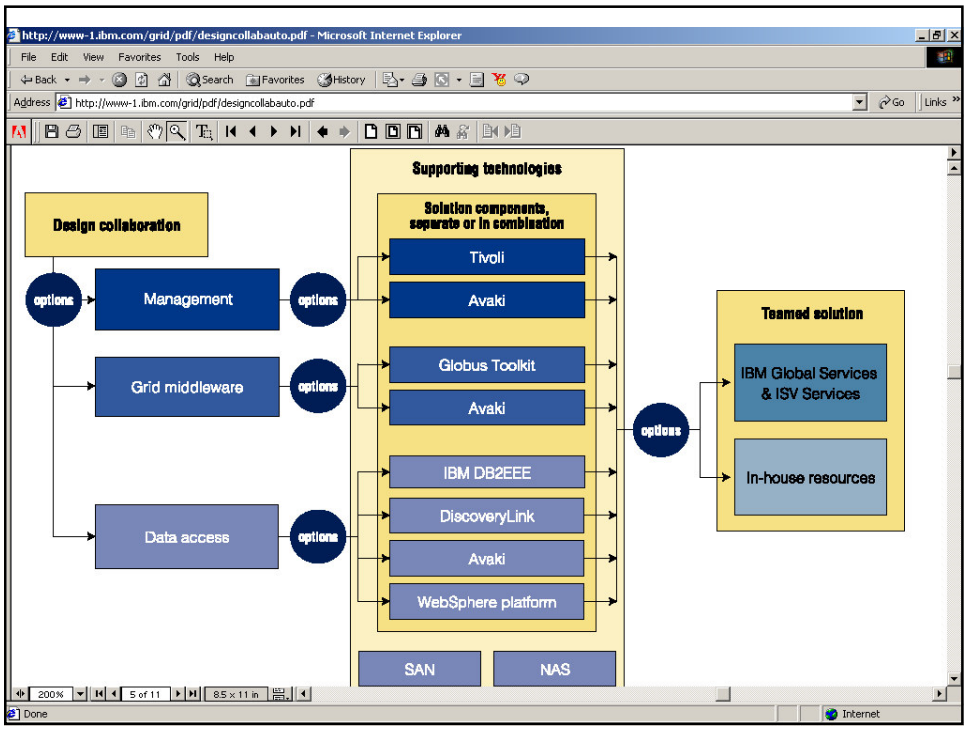
CS514

App	Discipline-Specific Data Grid Application
Collective (App)	Coherency control, replica selection, task management, virtual data catalog, virtual data code catalog, …
Collective (Generic)	Replica catalog, replica management, co-allocation, certificate authorities, metadata catalogs,
Resource	Access to data, access to computers, access to network performance data, …
Connect	Communication, service discovery (DNS), authentication, authorization, delegation
Fabric	Storage systems, clusters, networks, network caches, …

IBM strategy

CS514

- Target specific industries with compute intensive tasks
  - Aerospace
  - Automotive
  - Life sciences
  - Financial
- Sell them on idea of utilizing wasted computing resources in the organization
- In coordination with other web services, data sharing, and workflow products…



A hard sell

CS514

- Requires buying, customizing, and managing new software
- Are the problems associated with that outweighed by increased utilization of resources?