The Grid & Related Computing Technologies

CMP0915

David W. Walker
Some Questions

• Do you have access to a computer on the Internet that you can use for this module?
• Have you heard of the Grid outside of this course?
• Do you have experience with Jini, CORBA, XML, or other distributed computing technology mentioned in the course?
Overview of Course

• An introduction to the Grid.
• Examples of Grids.
• The computing technologies upon which the Grid is based.
Assessment

• Exam 50%.
• Independent study topic 25%. This will be on Globus, the Open Grid Service Architecture, and the Web Services resource framework
• Laboratory class on XML and XML Schema 25%. This will be arranged for the second contact weekend, 21 – 22 May 2004.
Introduction to the Grid

• What is the Grid?
• Analogies for viewing the Grid.
• Examples of Grid use:
  – Utility computing
  – Enterprise integration
  – Virtual Organisations
  – Grid-Enabled Computational Electromagnetics
• Problem-solving environments (PSEs)
• Cardiff University’s involvement in the Grid.
• Looking to the future.
The Grid Vision

Imagine a world.....

• In which computational power is as readily available as electrical power.
• In which computational services make this power available to users with differing levels of expertise in diverse areas.
• In which these services can interact to perform specified tasks efficiently and securely with a minimum of human intervention.
What is the Grid?

• The Grid is an emerging communication and computational infrastructure for the transparent sharing of distributed computing resources.

• Resources include computers, data, instruments, sensors, visualisation platforms, and sometimes even people.

The Grid is much more than just a faster Internet
The Grid = Resources + middleware + interfaces + users

Middleware = intermediate software infrastructure
Analogies for the Grid

• Electrical power / computing power analogy.
• Autonomic nervous system / adaptive response to demand analogy.
• Insect colony / community made up of many components.
Electrical Power Analogy

- Computing power is produced much like utilities such as power and water are produced for consumers.
- Users will have access to “power” on demand.
- Treat CPU cycles and software like commodities.
- Enable the coordinated use of geographically distributed resources – in the absence of central control.
- Transparency and pervasiveness are key aspects of this analogy.
The Transparent Grid

- **Transparency**: The user is not aware (and doesn’t care) what computing resources are used to solve their problem.
- Similarly, in an electrical grid we ignore the source of the power (oil, coal, etc..).

Distributed computing issues

- Heterogeneity
- Resource discovery
- Scheduling
The Pervasive Grid

- *Pervasive*: The Grid can be accessed from any networked device, eg, laptop, mobile phone, PDA, etc.
- In electrical analogy, any appliance can access power through a standard interface, eg, a wall socket.
  - Standard interfaces
  - Protocols
  - Legacy software
Autonomic Nervous System
Analogy

• Usually we are unaware of the workings of the ANS because it functions in an involuntary, reflexive manner.

• For example, we do not always notice when blood vessels change size or when our heart beats faster.
ANS Analogy

- When we run our body responds by supplying more power (oxygen) to our muscles.
- Breathing speeds up as well as other changes.
- The Grid should be self-regulating, and adaptive in meeting the demands made upon it.
Another Human Body Analogy: Self Healing

• When we are injured (but not too seriously!) our body responds to repair the damage.

• The Grid should be self-healing, resilient, and fault-tolerant in dealing with accidental and malicious damage to its infrastructure.
Insect Colony Analogy

- Large numbers of interacting autonomous components may display emergent or societal behaviours.
- The Grid is seen as “virtual community” of interacting people and services.
- Ability of complex systems to self-organise – perhaps in ways we cannot predict at present.
- How do you ensure the “correct” type of collective behaviour?
Other Grid Issues

- **Security**: When resources are shared across organisation boundaries security is an important issue.
- **Efficiency**: Resources should not be wasted, good load balancing needed.
- **Cost**: For broad impact The Grid should be inexpensive.
- **Portability**: Grid applications should be able to run on a wide range of hardware.
Service Providers and Brokers

• Trend is towards network-based computing paradigm.
• Nodes offer different sets of computing services with known advertised interfaces.
• Software is increasingly seen as a “pay-as-you-go” service rather than a product that you buy once computational economies.
• Web services important in architecture of the Grid Web Services Resource Framework (WSRF).
Wouldn’t it be great if…

It’s like a power grid, the ANS, an insect colony…

Use cases…

Desirable features…

Working Grid

Increasing difficulty, complexity, cost, and value

VISION

FUNCTION

IMPLEMENTATION
Use Case 1: Utility Computing

- Access the power of a remote computer from user’s desktop.
- Used to run “canned” applications provided on remote computer.
- End users may be provided with an API or job submission/management tools.
- Usually a fee is charged.
- Quite a simple use of the Grid.

Do this!

Here are your results!
Utility Computing: Advantages

• End user doesn’t have to install, maintain, or update application.
• Service provider has low user support costs, and there is less likelihood of the code being pirated.
• Simple business model.
• Also works for data and storage.
Utility Computing: Issues

• Can the end user trust the results returned by the service? (trust, reputation)
• Should the end user be allowed to execute their own applications on the remote machine? (trust, reputation)
• If there are several remote resources all providing the same service, which one should an end-user use (discovery, negotiation, and markets).
NetSolve: a Scientific Utility
Computing

- NetSolve is a client-server system for remote solution of complex scientific problems.
- NetSolve searches for computational resources on a network, chooses the best one available, and returns the answers to the user.
- Developed by Professor Jack Dongarra and colleagues at University of Tennessee, Knoxville.
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

This slide is courtesy of Professor Jack Dongarra
Network Enabled Server

• NetSolve is an example of a Grid based hardware/software server.
• Based on a RPC model but with …
  – resource discovery, dynamic problem solving capabilities, load balancing, fault tolerance asynchronicity, security, …
• Easy-of-use paramount
• Other examples are NEOS from Argonne and NINF Japan.

This slide is courtesy of Professor Jack Dongarra
Goals of the NetSolve System

• Ease-of-use for the user
• Efficient use of the resources,
• The ability to integrate any arbitrary software component as a resource into the NetSolve system.
NetSolve

• Target not computer scientist, but domain scientist
• Hide logistical details
  – User shouldn’t have to worry about how or where (issues about reproducibility)
• Not intended for running a task across a Grid of machines, but on a single system on the Grid.
• Present the set of available remote resources as a “multi-purpose” machine with a wealth of scientific software

This slide is courtesy of Professor Jack Dongarra
NetSolve: The Big Picture

Client

Agent(s)

Schedule
Database

Matlab
Mathematica
C, Fortran
Java, Excel

S1
S2
S3
S4

Request
S2

Answer (C)

A, B, C

A

C
NetSolve Example - Matlab

%load matrices
load 'mat1';.....;load 'mat16';
Matfiles = {mat1,mat2,....,mat16};
clear 'mat1';.....;clear 'mat16';
n = 16;
for i=1:1:n
    request(i) = netsolve_nb('send', 'square', Matfiles(i), Matfiles(i));
end

..... do some work locally ..... 
for i=1:1:n
    square(i) = netsolve_nb('wait', request(i));
end
Use Case 2: Enterprise Integration

• A distributed organisation wants to make all its compute resources (cycles, data, storage) accessible to all its members.
• Large companies are often interested in this.
• Assumes everyone in the organisation is trusted.
• Corresponds to a simple mini-grid.
Enterprise Integration: Advantages

• Reduce total cost of ownership.
• Improve productivity.
• Extend capability.
• Reduce design costs
• Reduce time to market.
Enterprise Integration: Issues

• Need some form of access control – not all members of the organisation should have access to all resources.
• Need some form of scheduling so that most important tasks have priority, and resources are accessed fairly.
Use Case 3: Virtual Organisations

- Many tasks requiring collaboration and resource sharing that spans organisational boundaries.
- Need mechanisms for controlling access and ensuring security.
- Same organisations may co-operate on one project and compete on another.
- VO is also called an “extended enterprise”
Design of a new aircraft

Collaborating organisations need to share:

• Digital blueprints of the design
• Supercomputers for performing multi-disciplinary simulations
• Computer code that performs those simulations (software)

This involves over 10,000 collaborating engineers
“Post-Genomic” Bioinformatics

- Based around many large federated databases
- Linked to the simulation of large molecules – protein folding affects how new drugs dock with receptors
- Linked to results of large numbers of micro-array experiments.

Gene Park initiative is an example of Grid-enabled research linking Cardiff/UWCM, Bangor, Aberystwyth, and Swansea.
Example of an Extended Enterprise: GECEM

• GECEM: Grid-Enabled Computational Electromagnetics.
• Partners are BAE Systems, HP, Singapore Institute of HPC, Swansea and Cardiff universities.
• Partners form a globally-distributed extended enterprise.
GECEM Production Grid

BAE SYSTEMS
Create geometry

Geometry data

UWS
Generate mesh

Output

Mesh

Output

Other locations

CEM simulation

Singapore

Output
Why CEM?

- CEM is important in civil and defence sectors.
- Complex electronic systems are key to platforms such as the More Electric Aircraft and the All Electric Ship.
- Response of systems to lightning strikes and EM pulses.
- CEM simulations of these systems are usually computationally intensive.
Why Use the Grid?

- Industrial and academic partners form an “extended enterprise” in which resources are intrinsically distributed, and only partially shared.
- Partners may be prepared to share data, but not the hardware and proprietary software that produces the data.
GECEM Objectives

• To use and develop Grid technology to enable large scale and globally distributed science and engineering.
• Key areas of interest include:
  – Performance and fault tolerance.
  – Secure remote execution.
  – Collaborative analysis and visualisation.
  – Easy and uniform access to resources via a problem-solving environment.
Issues in Remote Execution

- Performance and resilience. Mainly network issues.
- Security. Would like to migrate code to remote resource, execute it, and return results with minimum risk of unauthorised interference.
The Output

• Status results are returned to the client throughout execution.
• Results of CEM simulation are returned to client and visualised.
Crisis Management

A crisis management team put together to control and eradicate a virulent strain of disease. Such a team might be drawn from government, the emergency and health services, and academia. Need to share:

- Information on the individuals who have caught the disease
- Information on the resources available to tackle the infection
- Epidemiological simulations for predicting the spread of the infection under different assumptions.
Collaborative Science

Physicists collaborating in an international experiment need to share:

• Experimental data and storage resources.
• Computers and software for extracting information from this data.
• Computers and software for interpreting the data using large-scale computer simulations.

Large Hadron Collider (CERN): raw data rate = 1 Petabyte/sec
Filtered rate = 100Mbyte/sec = 1 Petabyte/year = 1 Million CD ROMs
Use Case 4: The Consumer Grid

• Computing resources and made available (at a price) in the form of services.
• Need mechanisms for specifying service contracts and paying for services.
• Issues of digital trust and reputation are important.
• Application Service Providers are a simple example of the CG concept, based on the utility computing concept.
Problem-Solving Environments

Gallopoulos et al. (1991):

A PSE is “a computer system that provides all the computational facilities necessary to solve a target class of problems.”

Note that this definition:

- Pre-dates the Web
- Ties a particular PSE to a particular class of problems
Vision for PSEs

• PSEs herald a new era in scientific computing, both in power and how resources are accessed and used.
• PSEs will become the main gateway for scientists to access terascale computing resources.
• PSEs will allow users to access these resources from any web connection.
• PSE’s support for collaborative computational science will change the pervading research culture, making it more open and accountable.
Aspects of PSEs

- Distributed
- Collaborative because interesting problems are often complex and draw on many types of expertise.
- Heterogeneity pervades network environments
- Transparency is usually desirable
- Intelligence is needed to deal with various types of complexity – especially complexity arising from the collaborative and distributed use of PSEs in heterogeneous dynamic environments.
Problem-Solving Environments

- We can view a PSE as an environment through which end-users of a Virtual Organization exploit Grid resources.
- PSEs are application specific.
- PSEs accessed through the Web/Grid are the same as “Application Portals.”
Intelligence in PSEs

• Collaborative and distributed computing adds new types of complexity to the software environment.
• PSEs need to be intelligent to minimize impact of this complexity on users.
Who is Involved in PSEs?

• Application end users (scientists, engineers, etc.)
  – Solve a particular problem which is domain specific
  – Undertake “what if” investigations

• Developers (programmers)
  – Create components and place them in Component Repository
  – Make new algorithms and techniques available to the end users.

• Software infrastructure builders
  – Create services and interfaces
  – Provide abstractions
  – Develop standards
PSEs and the Grid

• PSEs share many characteristics of the Grid.
• When Cardiff research into distributed PSEs started there was no Grid infrastructure.
• Aim now is to build PSEs on top of generic Grid/Web services.
• Many companies interested in web services: Sun (Jini, JXTA), IBM (Web Sphere), Microsoft (.NET), etc.
Service-Based PSEs

| PSEs | Domain-specific intelligence  
|      | Problem specification tools  
|      | Recommender systems  
|      | Applications and supporting tools  
|      | (toolkits, Grid-enabled libraries, etc)  
|      | Application development and execution support services (CORBA, Globus, Jini, etc)  
|      | Generic web services (monitoring, security, brokering, fault management, etc)  
|      | Distributed resources (computers, storage, instruments, sensors, visualisation, etc)  

PSEs: Problem Specification and Solution

• Use either
  – Visual programming environment to link software components
  – High-level language specification

• Recommender systems can be used to help user choose best way to solve problem and locate software.
Collaborative Code Development Environment

• The collaborative code development environment uses a visual programming tool for seamlessly integrating code from multiple sources.
• Applications are created by plugging together software components.
• Different developers can place their components in a shared repository.
• Legacy codes in any major scientific programming language can be handled.
Plug and Play Components

• Can link the output of one component to the input of another.

• Store components in a repository.

See Triana for example [http://triana.co.uk/](http://triana.co.uk/)
Triana ToolBox Window
Here, two sine waves, 1kHz and 2kHz are added and then passed to a NoisyFFT component and displayed with the Grapher.
In Triana, any units can be grouped and groups can contain subgroups. Here we show the NoisyFFT group used in our network.
Graphical Output

FOR MORE INFO...

See website: http://triana.co.uk/
Scheduling Issues

• Need to be able to monitor hardware resources.
• Need to be able to locate software resources.
• May need to negotiate for use of resources.
Scheduling

• Need to schedule components on distributed resources to achieve goals such as
  – Minimum execution time, or
  – Maximum throughput

• Achieving fair and efficient transparent access to resources is a difficult problem in a dynamic environment.
e-Everything

• The Grid is being used for in a variety of areas.
• The Grid challenges our current ways of doing things.
• The Grid is a transformative technology.
Impact : e-Science

From the EPSRC e-Science web site:

"In the future, e-Science will refer to the large-scale science that will increasingly be carried out through distributed global collaborations enabled by the Internet. Typically, a feature of such collaborative scientific enterprises is that they will require access to very large data collections, very large scale computing resources and high performance visualisation back to the individual user scientists."
Healthy, Wealthy, and Wise?

- **e-Health**: electronic patient records, distributed and/or remote diagnosis, collaborative surgical planning.
- **e-Business**: streamline, distribute, and enhance business processes.
- **e-Commerce**: use the Grid as a marketplace for both traditional and innovative goods and services.
- **e-Learning**: remove barriers to education and training.
Cardiff’s Role in the Grid

- COMSC hosts the Welsh e-Science Centre.
- WeSC is one of 8 regional e-Science centres in the UK.
- Funded by DTI, WDA, and CU.
- Manages a portfolio of DTI and industry funded projects worth £2.3M
UK Grid Network

- National Centre in Edinburgh/Glasgow
- 8 regional centres
- Grid support centre
- 7 Centres of Excellence
Cardiff e-Science Projects

- **Biodiversity World**: creating a Grid-based problem solving environment for collaborative exploration and analysis of global biodiversity patterns. (BBSRC)
- **GridOneD**: creating Java middleware for Grid applications. (PPARC)
- **GridLab**: Grid middleware project to support applications through GAT. (EU)
EPSRC-Funded Cardiff e-Science Projects

- WOSE: creating workflow-oriented middleware for the Grid. (with Imperial College and DL)
- **PASOA**: examining execution and service provenance in relation to workflow enactment. (with Univ. of Bath)
- GENSS: use of agents in scientific computation services on the Grid. (with Univ. of Southampton)
- GSiB: Creation, deployment, monitoring, and managing services on the Grid.
Looking to the Future

• Is the Grid inevitable – yes, at least in some areas.
• Barriers to adoption of Grid are cultural as well as technical.
• Grid is being pursued by commercial companies such as IBM and Oracle.
• “Grid revolution” expected to be complete by 2010.
Take-up of Grid Technologies

- PriceWaterhouseCoopers
- IBM
- Gartner
- Foster & Kesselman

Importance to business

- Critical
- Low/Zero

Timeline:
- 2000: i) production grids for research
- 2002: ii) ~35 private Grid deployments announced
- 2004:
- 2006: Grid service providers (GSP)
- 2008: 15% of corporates using GSPs
- 2010:
- 2012:

Key Points:
- Technology maturity
- "significant momentum"
- "Great Global Grid"
- Science & non-science sectors
- "mature", real deployments
- Computing as a utility
- Pervasive computing
- "full revolution begins"
- "Great Global Grid"
Concluding Remarks

• This talk has only touched on a few examples of Grid applications.
• The vision of the Grid is ambitious and far-reaching in its potential impact.
• The Grid is an engine for economic progress and social change driven by a confluence of technologies.
• We are at the start of the Grid era.
• Establishment of the Grid is a long term project.
The Last Slide

• We have covered the first section of the module:
  – Grid analogies
  – Virtual Organisations and the Consumer Grid
  – Problem-Solving Environments

• You should be able to start the independent study work.

• Take a look at the examples of Grids in Section 2 of the module.

• Next time we’ll look more at computing technologies, especially XML.
Welsh e-Science Centre
Canolfan e-Wyddoniaeth Cymru