Grid Computing
lecture by Camiel Plevier

• History of Computation Infrastructure
• Grid Concept
• Basic Architecture
  – Resources
  – Human Interfaces
  – Grid Middleware
• Grid Topologies
• Security
• Application Development
• Dutch Space & Grid
• Demonstration

History of Computation Infrastructure

IT evolution
• 1960’s - (super) computers with user terminals
• 1970’s - desktop PC
• 1980’s - networks and clusters (homogenous grid)
• 1990’s - Internet connectivity (WWW, HTML, optical networks) & open source (Linux)

Do you need more?
Grid Concept

- Many heterogeneous computers over the whole world can be used to provide a lot of CPU power and data storage capacity
- Applications can be executed at several locations
- Combining geographically distributed services
- Collaboration
- Seamless access, Web services

Basic Architecture

<table>
<thead>
<tr>
<th>User portals</th>
<th>General Grid Access and information</th>
<th>Problem Solving Environments Project tools and experimental devices</th>
<th>Application Science Scientific resources and devices</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Service layer (Grid services)</td>
<td>Network Security, scheduling, events, monitoring and maintenance.</td>
<td>Information Resource and data management.</td>
</tr>
<tr>
<td></td>
<td>Resource layer (Physical resources)</td>
<td>Super-computers Cluster Standalone computers Online instruments Corporate network/Intranet Databases</td>
<td></td>
</tr>
</tbody>
</table>
Resources

Anything that your application might need
- Computing – CPU type and speed, memory size
- Data Centres – capacity, sharing
- Communications Network – connectivity, bandwidth
- Software and Licences – OS, local tools and libraries
- Special equipment, services

Non-Technical Characteristics
- Authorisation Policies
  - for example: for medical use only
- Accounting

Human Interfaces of Grid

- User portal or client tools
  - Job definition, submission, control, monitoring and result collection
  - Available grid capacity monitoring
- Resource Provider
  - Sharing based on characteristics
  - Installation, administration and maintenance.
- Administrator
  - Grid status monitoring
  - User and resource management
  - Middleware and server maintenance
Grid Middleware

Software that connects other software components or applications to provide the following functions:

- Run applications on suitable available resources
  - Brokering, Scheduling
- Provide uniform, high-level access to resources
  - Semantic interfaces
  - Web Services, Service Oriented Architectures
- Address inter-domain issues of security, policy, etc.
  - Federated Identities
- Provide application-level status monitoring and control

Grid Topologies

- Intragrid
  - Local grid within an organisation
  - Trust based on personal contracts
- Extragrid
  - Resources of a consortium of organisations connected through a (Virtual) Private Network
  - Trust based on Business to Business contracts
- Intergrid
  - Global sharing of resources through the internet
  - Trust based on certification
**Security**

- Authentication, Authorisation, Accounting, Assurance
- Public-key cryptography, Certificates

**Grid Application Development**

- From Existing Programs
- Parallel Speedup vs. Sequential Fraction
- Formulating new algorithms
  - Reduce dependencies between calculation parts
- Embarrassingly Parallel
  - Monte Carlo Simulations
  - Repeated application to unrelated data (for example: spatial distribution)
Parallel Speedup vs. Sequential Fraction

Amdahl’s law

- speedup assumed proportional with number of processors
- viewpoint from existing application; new applications will be fitted to the possibilities

Grid Computing by Camiel Plevier

Dutch Space & Grid
Demonstration

Tide_grid_workflow_demo.avi

Thank you for your attention
Exercise

• Pick a role from the stack.
• Play your role in an anticipating fashion; try to prepare as much as possible
• Every data communication should be on paper, passed on one-on-one (no broadcasting)
  – User <-> Middleware
  – Middleware <-> Resource
  – NO resource <-> resource (third party transfers)
• Middleware actors should discuss to divide the work. Write discoveries and agreements on the role definition paper.

Role overview and Workflow

• User
• Middleware
  – User Portal
  – Controller
  – Scheduler
  – ... Need more?
  – Broker
  – Agents
• Resources
• Static
• Dynamic resources
• Failing jobs
• Cancel application
• Different topology
  – grouping resources into cluster
  – dividing middleware into middleware groups