Grids and User Applications

Viera Sipkova
Institute of Informatics, SAS
Dúbravská cesta 9, Bratislava
Slovakia
Outline

- Grid Architecture
- Globus Toolkit 4
- Computational problems
- Execution Management
- Examples
Grid

A New Infrastructure for 21st Century Science (Ian Foster)

- An emerging infrastructure that provides seamless access to computing power and data storage capacity distributed over the world.
- It makes possible to share resources for geographically distributed groups to work together.
Virtual Organization (VO)

A set of disparate institutions and/or individuals which are enabled to share resources in a controlled fashion.

- Resource – any artifact, entity or knowledge required to complete an operation on the system.
Open Grid Services Architecture (OGSA) (Global Grid Forum)

- Integrates Grid technologies and Web services.
- Defines a common, standard behaviors and interfaces for Grid Services.
- **Grid service** – a Web service that can be created dynamically and that supports security, lifetime management, manageability, and other functions required in Grid scenarios.
Grid Architecture

- **Fabric**
  physical infrastructure – computers, storage systems, networks, various sensors, ...

- **Middleware**
  - **Connectivity** – communication and authentication protocols, enable the exchange data between resources.
  - **Resource** – protocols enabling the secure initiation, monitoring and control of resource-sharing operations.
Grid Architecture

- Tools and applications
  - Directory brokering, diagnostics, and monitoring
  - Secure access to resources and services
  - Diverse resources such as computers, storage media, networks, and sensors

- User applications
- Collective services
- Resource and connectivity protocols
- Fabric
Grid Architecture

- **Collective services** – implement interactions across collections of resources - resource discovery and allocation, monitoring, diagnostics,…

- **Applications and Tools** – top layer, constructed on any other layer, and operating within a Virtual organization.
A grid application – might proceed by:

1. **Obtaining** authentication credentials.
2. **Querying** an information system to determine availability of resources.
3. **Submitting** requests to computers and networks to initiate computations, move data,...
4. **Monitoring** the progress of computations and data transfers.
Globus Toolkit 4 (GT4)
(Globus Alliance)

The software toolkit – a set of Grid infrastructure services and libraries that support programming grid-based applications and tools.
• **Security Tools** – establishing the identity of users or services, determining access rights, protecting communication.

• **Data Management** – location, transfer, and management of distributed data.

• **Execution Management** – initiation, monitoring, management, scheduling, and coordination of remote computations.
GT4 Components

- **Information Services** – Web services to monitor and discover resources and services on Grids.

- **Common Runtime** – several Web services, libraries and tools.

Let's suppose we have a computational problem which is simply too complex to be executed on just one computer.

What is to do?
Computational Categories
(from a computer scientist's point of view)

- General computational nature
  - degree of parallelism
  - granularity

- Computation-centric
- Data-centric
- Community-centric
Degree of parallelism

- If the computation problem can be split into many smaller sub-problems that can be worked on by different processor in parallel, then the computation can be speed up a lot by using many computer resources from across a business, a company or an academic institution.
• **Distributed computing**
  any system where many computing resources solve a problem together – this network of computers is used as a single, unified resource.

• **Grid computing**
  one species of distributed computing.
High-performance computing

- **fine-grained** parallel calculations – each sub-problem is highly dependent on results of other sub-problems.
  - big monolithic supercomputer
  - cluster of computers
  - very tightly coupled clusters of computers

- For development of such applications – **Message Passing Interface** (MPI) model is used.
High-throughput computing

- coarse-grained parallel calculations – each subproblem is independent of all others.
  - loosely coupled network of computers

- For development of such applications – high-level distributed programming models are applied: Web technologies, CORBA, workflow systems, MPICH-G2, ...
Grid computing

- A combination of fine-grained and coarse-grained parallel approaches.
  - loosely coupled network of computers and clusters of computers

- For development of such applications –
  - high-performance computing technologies
  - Web technologies
  - Grid technologies
• **Computation-centric** problems
  the domain of high-performance computing and grid computing.

• **Data-centric** (data-intensive) problems
  the primary driving force behind the Grid; huge amounts of scientific data will be stored and analyzed in geographically distributed repositories, libraries and databases.

• **Community-centric** problems
  attempt to bring people and communities together for collaborations of various types.
Job Management

- Run an executable on a remote computer.
- Run a parallel program across multiple distributed computers.
- Run a set of loosely coupled tasks.
- Make a program available as a network service.
GT4 Job represents a computational task (executables, scripts) which may perform input/output operations (staging of data) while running.

- A job may be parallel - multijob - a job that is itself composed of several executable jobs.
- A job can be described through an XML Schema Resource Specification Language (RSL).
Grid Resource Allocation and Management (GRAM) represents a basic mechanism for initiation, monitoring, management, and/or scheduling of remote computations.

- GRAM implementation is built on Web services technologies (WS GRAM).
- WS Resource Framework is used to implement distributed communications and service state.
- WS GRAM services provide a job submission to an optional local job schedulers: PBS, LSF, Condor.
WS GRAM

is based on a component architecture at both the protocol and software implementation levels.

- The job management with GRAM makes use of multiple types of services:
  - Job management services
  - Credential management services
  - File transfer services
• **Job management services** represent, monitor and control the overall job life cycle.

• **Credential management services** are used to control the delegation of rights among distributed elements of the GRAM architecture on users' application requirements.

• **File transfer services** provides for reliable transfers of files between the compute resources and external data storage elements before and after the job execution.
Job Management Services

- Each submitted job is exposed as a distinct resource qualifying the generic ManagedJob service providing an interface to monitor the status of the job or to terminate the job.

- Each compute element is exposed as a distinct resource qualifying the ManagedJobFactory service providing an interface to create ManagedJob resources.
Job Management Client

- **globusrun-ws** – the official WS GRAM command line client for submitting and managing jobs to a local or remote host.

- The client provides a secure job submission for users who have the right to access a job hosting resource in a Grid environment.
The **globusrun-ws** client

- A job may be submitted by a job-command directly or by using a job-description file.
- Interactive and batch modes are supported with reattachment (recovery), and monitoring of jobs.
- Additional features are offered to
  - fetch job output files during the run
  - delegate credentials automatically
  - determine lifetime
Submitting a simple job (interactively)

- **globusrun-ws** -submit \\ -job-command /test/maxcut /test/adjmatrix.dat

Submitting job...Done
Job ID: uuid:n9a19984-55d7-...
Termination Time: 11/16/2006 12:59 GMT
Current Job State: Active
Current Job State: CleanUp
Current Job State: Done
Destroying job...Done
Submitting a simple job (in batch mode)

- `globusrun-ws -submit -batch \` -job-epr-output-file /test/maxcut.epr \` -job-command /test/maxcut /test/adjmatrix.dat

- `globusrun-ws -status \` -job-epr-file /test/maxcut.epr
Submitting a simple job to a remote host

- `globusrun-ws -submit \`
  -factory https://remotehost:8443/wsrfservices/
    ManagedJobFactoryService \`
  -job-command /test/maxcut /test/adjmatrix.dat
Submitting a simple job with job description

- `globusrun-ws -submit \n  -job-description-file /test/maxcut_desc.xml`
Job description file

<!-- maxcut_desc.xml -->
<job>
  <executable>/test/maxcut/</executable>
  <directory>/test</directory>
  <argument>adjmatrix.dat</argument>
  <stdout>stdout.maxcut</stdout>
  <stderr>stderr.maxcut</stderr>
</job>
Thank you for your attention.

http://www.eu-egee.org
http://www.ggf.org
http://www.globus.org
http://www.w3.org