TUTORIAL:
Managing Computational Activities on the Grid - from Specifications to Implementation

Tutors:
Sergio Andreozzi (INFN-CNAF, Italy)
Aleksandr Konstantinov (University of Oslo, Norway)
Balazs Konya (Lund University, Sweden)
Morris Riedel (FZJ, Germany)
Gabor Roczei (NIIF, Hungary)

OGF 23, 2-6 June 2008, Barcelona (Spain)
OGF-Europe
Tutorials@OGF23

• Thursday, 5 June, Afternoon
  – Managing Computational Activities on the Grid
    - from Specifications to Implementation
  – 3 sessions – 90' each

• Friday, 6 June, Morning
  – Accessing and Integrating Structured Data using OGSA-DAI 3.0 and Links with WS-DAI
  – Managing Files & Storage Spaces using the Storage Resource Manager (SRM) Interface
Structure of This Tutorial

• Session 1 (90'):
  – Introduction to Job submission
  – OGF Standards for Job Management - part 1

• Session 2 (90'):
  – OGF Standards for Job Management - part 2
  – Implementation of standards and extensions to them

• Session 3 (90'):
  – Limitations and gaps
  – Security Aspects
  – Exercises
Outline

• Introduction to the Tutorial
  – Motivation
  – Goal
  – Structure
Motivations

• Several production Grid infrastructures are running around the globe
  – Lack of interoperability
  – Barrier to collaboration, resource usage optimization

• The Open Grid Forum has defined a number of standards
  – managing High-Performance Computing activities is covered

• On-going adoption by several parties
Goal

• Increase awareness of adoption process for OGF standards
• Engage new middleware providers
  – knowledge transfer
  – stimulate adoption
• Feedback to OGF community
  – implementation experience
  – limitations/gaps
  – extensions
Tutors

• Sergio Andreozzi
• Balazs Konya
• Aleksandr Konstantinov
• Gabor Roczei
• Morris Riedel

• Contributors:
  – Moreno Marzolla, INFN-PD, provided material
  – Dawid Seijnfeld, PSNC, GridSphere portal
Evaluation

• Your opinion is important
• Fill the web survey before to leave the tutorial

http://www.zoomerang.com/Survey/?p=WEB227VNDMAWGF
Planning for Exercises

- In third session, live exercises
- If you want your hands on Grid technology based on OGF standard, you need:
  - computer
  - network connection
  - ssh client
- We can also run the exercises in demo mode
Introduction to Job Submission

Tutor: Sergio Andreozzi (INFN)
HPC

• HPC: High Performance Computing
  – Computing on
    • Supercomputers
    • Commodity clusters

• Core HPC
  – Batch scheduling on HPC systems

http://www.ogf.org/documents/GFD.100.pdf
HPC Use Cases

• HPC Base Case:
  – High-throughput compute cluster used only within the organization boundaries
  – Once a job has been submitted it can be canceled, but its resource requests can't be modified

• HPC Grid Case:
  – Job submission spanning the organization boundaries
Types of Jobs

• Single Job

• Parallel Job
  – Instantiate many activities to different compute nodes
  – Activities talking to each other

• Workflow
  – Instantiate many activities to different compute nodes
  – Activation based on Static/Dynamic dependencies
HPC Scheduling: Direct Submission

- The client knows what is the final HPC system to which it should submit
- It submit directly to the resource manager interface
HPC Scheduling: Indirect Submission
HPC Scheduling: Indirect submission
“A Thousand Way of Managing Jobs”

everybody knows at least one way of submitting grid jobs

- the problem domain is rather simple:
  - authenticate against the resource (optional)
  - start up a computational activity (called submission)
    - optionally stage-in data
  - manage the activity
    - get monitoring info and status updates, cancel
  - fetch results, stage-out data
- large variety of pre-OGF standard solutions appeared
“Thousand Way of Managing Jobs”: Traditional Batch Systems with ssh

- Traditional batch systems:
  - LSF, Torque/PBS, Condor, SGE, Loadleveler,
- Capable managing jobs locally within a cluster
  - optionally from submission nodes
- All have their own communication protocols, job description format
- Typical set of commands:
  - qsub, qdel, qstat, qalter, qrun, qhold,
- Some of the batch systems offer APIs as well
  - DRMAA - OGF recommendation (GFD.130)
- batch system functionality can be utilized via remote ssh
“Thousand Way of Managing Jobs”: Grid Solutions

• NorduGrid ARC Classic interface
  – custom protocol based on instructions encoded in XRSL and communicated via GridFTP special plugin.
  – Appendix A “job control over jobplugin.so”
    http://www.nordugrid.org/documents/GM.pdf

• Globus GRAM component
  – Globus/Grid Resource Allocation Management (GRAM)
  – a custom protocol/API for submitting, monitoring, and terminating jobs
  – GRAM2 (non-WS) and GRAM4 (WS-based) versions are available
“Thousand Way of Managing Jobs”: Grid-level Solutions

- EGEE gLite
  - Clients indirectly interact with the Computing Elements (CE) via the Workload Management System (WMS); upload job requests formulated in JDL, monitor/manage jobs
  - WMS exposes WS-interface (called WMProxy)
  - CE: two flavours:
    - GT2 GRAM, CREAM (WS-based)
- UNICORE 6
  - Covered by UNICORE Atomic Services (UAS)
    - Numerous proprietary Web Services for job management, storage management & file transfer
      - TargetSystemService (takes JSDL) to submit jobs
      - JobManagementService for handling the job itself
Interoperability Problem

- Interoperability
  - systems/organizations
  - able to provide/accept services
  - from other systems/organizations
  - and to use the services exchanged
  - to enable them to operate effectively together

(US defense department)

Approaches to Interoperability

- **Adapters-based:**
  - The ability of Grid middlewares to interact via adapters that translate the specific design aspects from one domain to another

- **Standard-based:**
  - The native ability of Grid middleware to interact directly via well-defined interfaces and common open standards
Value of Implementing Standards

- For vendors
  - meet customer demand for interoperability
- For developers
  - leverage the expertise of other developers
  - offer a choice of tools and platforms in order to speed implementations
  - only need to support one integration interface
- For end-users
  - reduce the costs and risks of adopting grid technology
  - get insight into the best practices of the industry at large
Type of Standards

• Information Schemas
  – defines the information that is passed between clients and services, or between peer services
  – provides a single vocabulary used to describe resources and activities

• Protocol Specifications
  – defines the messages that pass between clients and services, and how the ordering of these messages effect a certain behaviour (i.e., operations)
  – the state of the interaction is as important as the messages passed

• Application Programming Interfaces
  – define programming language interfaces that expose desired service functionality within a client program
  – generally focused on users (clients) of services, rather than service providers
Standard: API vs. Protocols

Workload Manager Client

- Native API
- DRMAA/SAGA
- WS-I compliant SOAP toolkit

- proprietary API
- standard API
- proprietary protocol
- proprietary protocol
- standard protocol

Native Protocol Engine
OGSA-BES

Workload Manager
Standards for Managing Jobs on HPC resources

- JSDL:
  - Job Submission Description Language
- It describes the requirements of computational jobs
  - for submission to Grids and other systems
- A JSDL document describes job requirements
  - What to do, not how to do it
- JSDL does not define
  - a submission interface
  - what the results of a submission look like
  - how resources are selected
Standards for Managing Jobs on HPC resources

• OGSA-BES
  – Basic Execution Service
• It defines a Web Services interface for creating, monitoring and controlling “Activities” such as OS processes, service instances or parallel programs
• Extensible state model for modeling the life-cycle of Activities
• Extensible information model for the BES itself and for the Activities it manages
Standards for Managing Jobs on HPC resources

• HPC Basic Profile
  – defines how to combine other specifications (JSDL, OGSA-BES)
  – for batch job submission on cluster with shared file system

• The HPC Basic Profile purposefully reduces the scope of JSDL and BES in order to ensure base line interoperability by clarifying ambiguities
Standards for Managing Jobs on HPC resources

• GLUE:
  – An information model for describing characteristics and status of Grid resources

• It enables to describe relevant characteristics required for resource selection and high-level management

• It can be used in
  – OGSA-BES services to expose resource properties
  – JSDL documents to express requirements on resources
Important Remark

• BES/HPC-P/JSDL/GLUE are OGF specifications that enable the job submission to HPC systems
• They are not yet fully integrated for real-world production scenarios
• The missing profiles will be defined and implemented in the coming months