Application development on EGEE with P-GRADE Portal

Gergely Sipos
MTA SZTAKI
sipos@sztaki.hu

EGEE Training and Induction
EGEE Application Porting Support

www.ipds.sztaki.hu/gasuc
www.portal.p-grade.hu
Running jobs in EGEE from command line

Write JDL, Submit workload

Retrieve status & (small) output files

Workload Management System

EGEE VO

Workload:
- Single job
  - One executable & one or more files
  - Sequential or MPI
- Workflow
  - DAG of jobs
- Parameter study
  - Independent, similar jobs
Defining a single job in JDL

$ vi OR nano OR ...  my.jdl

JobType = "Normal";
Executable = "gridTest";
StdError = "stderr.log";
StdOutput = "stdout.log";
InputSandbox = {"/home/sipos/gridTest"};
OutputSandbox = {"result.txt", "stderr.log", "stdout.log"};

$ glite-wms-job-submit my.jdl  JobID

$ glite-wms-job-status JobID

$ glite-wms-job-output JobID
Defining a workflow in JDL

```plaintext
[ Type = "dag";

InputSandbox = {
    "sharedFile1"; . . .; "sharedFileN"
};

nodes = [
    job1 = [
        description = [
            JobType = "Normal";
            . . .;
        ],
    ]; . . .
];

dependencies = {
    {job1, {job2, job3}}, {job2, job4}, {job3, job4}
};

Graph structure
```
Defining a parameter study in JDL

```plaintext
[ Type = "Parametric";
  ...

  ParameterStart = 0;
  ParameterStep = 1;
  Parameters= 6;  \_\_PARAM\_ run from 0 to 5

  Arguments = "figure\_PARAM\_.jpg";
  StdOutput = "transformed\_PARAM\_.jpg";
  OutputSandbox  = {" transformed\_PARAM\_.jpg ",...};
  ...
]
```

Figure0.jpg  Figure1.jpg  Figure5.jpg

Job 0   Job 1   ...   Job 5

Transformed0.jpg  Transformed1.jpg  Transformed5.jpg
Managing data in EGEE from command line

Run file management commands
Upload, transfer, register files

Download files
Delete files

$ lfc-mkdir myCatalogEntry
$ lfc-rmdir myCatalogEntry
$ lcg-cr -vo voName -d SEName -l myCatalogEntry file

EGEE VO

File catalog & Storage Elements

User Interface

Run file management commands
Upload, transfer, register files

Download files
Delete files

$ lfc-mkdir myCatalogEntry
$ lfc-rmdir myCatalogEntry
$ lcg-cr -vo voName -d SEName -l myCatalogEntry file
Short History of P-GRADE portal

- **Parallel Grid Application Development Environment**
- Initial development started in the Hungarian SuperComputing Grid project in 2003
- **Motivation:** Simplify grid usage for application developers and end users
- It has been continuously developed since 2003
  - Around 30 manyear development + training + user support
- **Detailed information:** [http://portal.p-grade.hu/](http://portal.p-grade.hu/)
- Open Source community since January 2008: [https://sourceforge.net/projects/pgportal/](https://sourceforge.net/projects/pgportal/)
  - Current version: 2.8
Current P-GRADE Portal related projects

- **GGF GIN** (Since 2006)
  - Providing the GIN Resource Testing portal

- **EU EGEE-II, EGEE-III** (2006-2010)
  - Tool recommended for application development
  - Intensively used in new users’ training

- **EU SEE-GRID-SCI** (2008-2010)
  - Interfacing to DSpace-based workflow storage
  - Infrastructure testing workflows

- **EU CancerGrid** (2007-2009)
  - Development of new generation P-GRADE (gUSE and WS-PGRADE)
  - Integration with desktop grids

- **EU EDGeS** (2008-2009)
  - Transparent access to Desktop Grid systems
Portal installations

P-GRADE Portal services:
- Several VOs of EGEE:
  - Biomed, Astronomy, Central European, NA4,...
- GILDA: Training VO of EGEE
- Southern Eastern European Grid
- Several national Grids (UK National Grid Service, HunGrid, Turkish Grid, etc.)
- US Open Science Grid, TeraGrid
- OGF Grid Interoperability Now (GIN) VO
- Under development: KnowledgeGrid of Malaysia
  - MIMOS & SZTAKI

Portal services and account request:
http://portal.p-grade.hu/index.php?m=3&s=0
Account request form on portal login page
Design principles of P-GRADE portal

- **P-GRADE Portal is not only a user interface, it is a**
  - General purpose
  - Application Development and Execution Environment

- **P-GRADE Portal includes a high-level middleware layer** for orchestrating jobs on grid resources
  - inside a grid
  - among several different grids and several VOs

- **P-GRADE Portal is grid-neutral:**
  - It is not tailored to any particular middleware grid type
  - Can be connected to various grids based on different grid middleware
    - LCG-2, gLite, GT2, GT4, ARC, Unicore, etc.
    - Implementation for GT2, GT4, gLite, ARC, LSF
  - GUI is the same no matter which type of grid is connected to it
What is a P-GRADE Portal workflow?

- **A directed acyclic graph** where
  - Nodes represent jobs (batch programs to be executed on a computing element)
  - Ports represent input/output files the jobs expect/produce
  - Arcs represent file transfer operations

- **Semantics of the workflow:**
  - A job can be executed if all of its input files are available
Three levels of parallelism

- Job level: Parallel execution inside a workflow node (MPI job as workflow component)

- Workflow level: Parallel execution among workflow nodes (WF branch parallelism)

- PS workflow level: Parameter study execution of the workflow
Example: Computational Chemistry

~100 independent jobs to run

Department of Chemistry, University of Perugia

SOLUTION OF SCHRODINGER EQUATION FOR TRIATOMIC SYSTEMS USING TIME-DEPENDENT (Rwavepr) OR TIME INDEPENDENT (ABC) METHOD

A single execution can be between 5 hours and 10 hours

Many simulations at the same time

SEQUENTIAL FORTRAN 90
**Typical user scenario**

**Job compilation phase**

- **Client**
  - Upload job source(s)
  - Download binaries
  - Compile – Edit

- **Portal server**
  - Receive upload
  - Compile and edit

- **Grid services**

Diagram showing the process flow between the client, portal server, and grid services.
Typical user scenario
Workflow development phase

Workflow development phase:
- Portal server
- Grid services
- DSpace WF repository
- Client
  - OPEN & EDIT WORKFLOW
  - ADD BINARIES
- START EDITOR
- SAVE WORKFLOW
- IMPORT WORKFLOW

Diagram:
- Client connects to Portal server through DSpace WF repository.
- The process involves editing and saving workflows.
Typical user scenarios

Workflow execution phase

MyProxy Certificate servers

TRANSFER FILES, SUBMIT JOBS

DOWNLOAD PROXY CERTIFICATES

VISUALIZE JOBS and WORKFLOW PROGRESS

CLIENT

Portal server

MONITOR JOBS

DOWNLOAD (SMALL) RESULTS

DOWNLOAD (SMALL) RESULTS

Grid services

MONITOR JOBS

DOWNLOAD (SMALL) RESULTS

DOWNLOAD (SMALL) RESULTS

VISUALIZE JOBS and WORKFLOW PROGRESS

TRANSFER FILES, SUBMIT JOBS

DOWNLOAD PROXY CERTIFICATES
Accessing local and remote files

Use legacy executables with Grid files without touching the code

Only the permanent files!
Workflow scheduler: Extended DAGMan

Java Webstart workflow editor

Web browser

Workflow language: Extended DAGMan

DSpace workflow repository

Grid Workflow Enactment Service

Workflow scheduler: Extended DAGMan

Grid interface: command line clients + scripts

EGEE, Globus (and ARC) Grid services
(gLite WMS, LFC,…; Globus GRAM, …)

Globus and gLite information systems
(Globus GIIS, gLite BDII)

Web interface - Portlets
Email notifications

Notify portal

Email Settings:
- Enabled: Yes
- Email Address: sipos@sztaki.hu
- Email Subject: P-GRADE Portal workflow state change

Workflow Change Settings:
- Enabled: Yes
- Message:
  - Time: #now#
  - P-GRADE Portal account: #user#
  - P-GRADE Portal installation: #portal#
  - Workflow name: #workflow#
  - Old status: #oldstatus#
  - New status: #newstatus#
  - Details: #details#
Workflow portlet
Portal administrator can:
- connect the portal to several grids
- register default resources of the connected grids
User can customize the connected grids by adding and removing resources.
Graphical workflow editing

- To define a graph:
  1. Drag & drop components: jobs and ports
  2. Define their properties
  3. Connect ports by channels (no cycles, no loops)

System generates JDL for each job automatically
Properties of a job:
- Executable file
- Type of executable (Sequential / Parallel)
- Command line parameters
- Which resource to use?
  - Which VO?
  - Broker or Computing element?
Workflow Editor
Defining input-output files

File properties

Type:
- **input**: the executable reads
- **output**: the executable generates

File type:
- **local**: comes from my desktop
- **remote**: comes from an SE

File:
- **location of the file**

Internal file name:
Executable uses this
- e.g. fopen(“file.in”, …)

File storage type (output files only):
- **Permanent**: final result
- **Volatile**: temp. data channel
Scaling up a workflow to a parameter study

Complete workflow

Grid files
(files in the same EGEE catalog entry)

Results stored in grid files
Advanced parameter studies

Generate or cut input into smaller pieces

Generator component(s)

Collect result

Complete workflow

Collector component(s)

Aggregate result
Concept of parameter study workflows

Generator part generates the input parameter space

Parameter study part

Collector part evaluates and integrates the results
Example: Image processing

\begin{itemize}
\item \(/\text{grid/gilda/sipos/InputImages/}\text{Image.0/}\text{Image.1/}\text{Image.2/}\)
\item \(/\text{grid/gilda/sipos/XCoordinates/}\text{XCoordinate.0/}\text{XCoordinate.1/}\)
\item \(/\text{grid/gilda/sipos/YCoordinates/}\text{YCoordinate.0/}\text{YCoordinate.1/}\)
\end{itemize}

\[3 \times 2 \times 2 = 12\]
execution of the whole workflow

CROSS PRODUCT of data items
### How to refer to an I/O file?

<table>
<thead>
<tr>
<th>Input file</th>
<th>Output file</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Local file</strong></td>
<td><strong>Local file</strong></td>
</tr>
<tr>
<td><strong>Remote file</strong></td>
<td><strong>Remote file</strong></td>
</tr>
</tbody>
</table>

#### Input file:
- Client side location: `c:\experiments\11-04.dat`
- LFC logical file name: `lfn:/grid/gilda/sipos/11-04.dat`
- GridFTP address (in Globus Grids): `gsiftp://somengshost.ac.uk/mydir/11-04.dat`

#### Output file:
- Client side location: `result.dat`
- LFC logical file name: `lfn:/grid/gilda/sipos/11-04_result.dat`
- GridFTP address (in Globus Grids): `gsiftp://somengshost.ac.uk/mydir/result.dat`
Upload a workflow from client side or from FTP server.

STORED on FTP server
Importing an application

INCOMPLETE WORKFLOW → Open it in editor and save it again
Import a workflow from DSpace repository
External access to DSpace

http://pgrade-dspace.sztaki.hu
Proxy creation: The first step of grid usage

User Interface (UI) or Portal
- Write JDL, Submit job (executable + small inputs)
- Retrieve status & (small) output files

Workload Management System (WMS)
- Submit job
- Retrieve output
- Logging
- Job status

Information System (BDII)
- query
- publish state

VO Management Service (VOMS)
- create proxy

Site X of YOUR VO
- Computing Element (CE)
- Storage Element (SE)
- process
- Read/write files
- Job status

Logging and bookkeeping (LB)
Certificate and proxy management Portlet

Certificate successfully set for gilda_GLITE_BROKER.

February 26, 2009
OGF GIN interoperability portal by P-GRADE
Accessing Globus, gLite and ARC based grids/VOs simultaneously
Application execution
Detailed view of a PS workflow

- Generator job(s)
- Overall statistics of workflow instances
- Workflow instances
- Collector job(s)
Fault-tolerant execution

- Utilizing
  - Condor DAGMan’s rescue mechanism
    - Resubmit only failed jobs, not completed jobs!
- If the EGEE broker leaves a job stuck in a CEs’ queue, the portal automatically
  - kills the job on this site and
  - resubmits the job to the broker by prohibiting this site.
- As a result
  - the portal guarantees the correct submission of a job as long as there exists at least one matching resource
  - job submission is reliable even in an unreliable grid
### Information system visualization

#### Grid: voce-BDII  VO: voce

<table>
<thead>
<tr>
<th>Site Name</th>
<th>Sites</th>
<th>Computing Element</th>
<th>Storage Element</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Total</td>
<td>Free</td>
</tr>
<tr>
<td>bmegrid</td>
<td>32</td>
<td>32</td>
<td>2</td>
</tr>
<tr>
<td>budapest</td>
<td>321</td>
<td>213</td>
<td>213</td>
</tr>
<tr>
<td>cyfronet-lg2</td>
<td>552</td>
<td>221</td>
<td>221</td>
</tr>
<tr>
<td>egee.fesb.hr</td>
<td>24</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>egee.grid.niif.hu</td>
<td>9</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>egee.irb.hr</td>
<td>24</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>egee.srece.hr</td>
<td>27</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>ete</td>
<td>5</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>gup-jku</td>
<td>8</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>hephy-vienna</td>
<td>158</td>
<td>122</td>
<td>122</td>
</tr>
<tr>
<td>iiias-bratislava</td>
<td>73</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>pearl-amu</td>
<td>18</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>prague_cesnet_lcg2</td>
<td>64</td>
<td>46</td>
<td>46</td>
</tr>
<tr>
<td>psnc</td>
<td>690</td>
<td>62</td>
<td>62</td>
</tr>
<tr>
<td>tu-kosice</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>warsaw-egee</td>
<td>208</td>
<td>155</td>
<td>155</td>
</tr>
</tbody>
</table>

*July 12, 2008*
Learn once, use everywhere
Develop once, execute anywhere

Thank you!

www.portal.p-grade.hu
pgportal@lpds.sztaki.hu