GRMS - The resource management system for Clusterix computational environment

Bogdan Ludwiczak
bogdanl@man.poznan.pl

Poznań Supercomputing and Networking Center
Outline:

- GRMS - what it is?
- GRMS features
  - What GRMS offers to its users?
- GRMS usage
  - How they can use it?
- GRMS architecture
What GRMS is?

- GRMS is an open source scheduling system for large scale distributed computing infrastructures.
- Designed to deal with resource management challenges in Grid environments:
  - load-balancing among clusters,
  - setting up execution environments before and after job execution,
  - remote job submission and controlling,
  - files staging,
  - More.
- Based on the dynamic resource selection, mapping and advanced grid scheduling methodologies, combined with feedback control architecture.
GRMS Features (1)

- Job submission & control (cancel, suspend, resume)
  - GRMS main objective:
    - GRMS performs remote jobs control and management in the way that satisfies Users (Job Owners) and their applications requirements.
    - Simultaneously, Resource Administrators (Resource Owners) have full control over resources on which all jobs and operations will be performed by appropriate GRMS setup and installation.
GRMS Features (2)

- Choosing the “best” resource for the Job execution, according to Job Description and chosen mapping algorithm - multicriteria algorithm
- Submitting the GRMS Job according to provided Job Description to chosen resource
- Job migration
- Job checkpointing:
  - using defined checkpoint interface implemented by application
Complex information about submitted jobs

- List of jobs submitted by user
- Information about the Job status
- Job Description used for submission
- Information about request progress
- Name of host where the Job is running
- Submission time
- Start time on resource
- Finish time
- History of job execution (migrations)
GRMS Features (4)

- Dynamic resource discovery
- Using multiple information sources about Grid environment
  - standard Globus MDS (GIIS/GRIS infrastructure)
  - Clusterix Infrastructure Monitoring System
  - iGrid/Delphoi/Mercury - GridLab Services
- Support for file staging - transferring input and output files and whole directories (GridFTP, GASS, Data Management Systems)
- Mechanism for registering for events notification
- Notifying about status changes and request progress (e.g. via e-mail or sms)
GRMS Features (5)

- Time constraints for running jobs
- Dynamically extending (during application runtime) job description by adding output data
- Support for workflow jobs: job can consist of set of independent tasks with or without precedence constraints
GRMS features developed in Clusterix

- **Support for distributed MPICH-G2 application**
  - allows users to submit jobs which will be dispersed among many nodes of many clusters,
  - makes Clusterix able to execute large, multi process application

- **Prediction of Job execution**
  - Increases the resource management effectiveness by providing estimated values
  - Allows resource broker to find out:
    - job execution time,
    - job pending time in given queue,
    - probable resource utilization by the job
    - estimation inaccuracy
GRMS Interface

- **GSI-enabled web service for all:**
  - clients
  - applications
  - middleware

- All users requirements are expressed through XML-based resource specification documents, called GRMS Job Description

- GRMS assigns a GRMS_JOB_ID to every submitted job
Job Description(1)

- XML based language
- Job executable
  - file location
  - arguments
  - file argument (files which have to be present in working directory of running executable)
  - environment variables
  - standard input
  - standard output
  - standard error
  - checkpoint files
Job Description (2)

- Resource requirements of executable
  - Name of host for job execution (if provided no scheduling algorithm is used)
  - Operating system
  - Required local resource management system
  - Minimum memory required
  - Minimum number of cpus required
  - Minimum speed of cpu
  - Network parameters: latency, bandwidth, capacity
<grmsjob appid="testjob" persistent="false">
  <simplejob>
    <resource>
      <osname>Linux</osname>
      <memory>128</memory>
      <cpucount>2</cpucount>
    </resource>
    <executable type="single" count="1">
      <file name="exec-file" type="in">
        <url>file:///bin/grep</url>
      </file>
      <arguments>
        <value>PCSS</value>
      </arguments>
      <stdin>
        <url>gsiftp://access/~/examples/datafile</url>
      </stdin>
      <stdout>
        <url>gsiftp://access/~/gclient/var/pbs.stdout.$(JOB_ID)</url>
      </stdout>
    </executable>
  </simplejob>
</grmsjob>
Job Description – MPICH-G2 example

<grmsjob appid="mpich2test" persistent="true">
  <simplejob>
    <resource>
      <hostname>access.pcss.clusterix.pl</hostname>
      <localrmname>pbs</localrmname>
    </resource>
    <executable type="mpichg" count="3">
      <file name="clx" type="in">
        <url>file:///tmp/clx_ia64_g2</url>
      </file>
      <arguments>
        <value>HOME/CLX/var/grms_demo1v2</value>
        <value>25</value>
        <value>1</value>
      </arguments>
      <stdout>
        <url>gsiftp://access.pcss.clusterix.pl/~demol.out</url>
      </stdout>
    </executable>
  </simplejob>
</grmsjob>
<grmsjob appid="mpichg2test" persistent="true">
  <simplejob>
    <resource>
      <hostname tileSize="2">access.pcss.clusterix.pl</hostname>
      <localrmname>pbs</localrmname>
    </resource>
    <resource>
      <hostname tileSize="1">access.pcz.clusterix.pl</hostname>
      <localrmname>pbs</localrmname>
    </resource>
    <executable type="mpichg" count="3">
      <file name="clx" type="in">
        <url>file:///tmp/clx_ia64_g2</url>
      </file>
      <arguments>
        <value>HOME/CLX/var/grms_demo2</value>
        <value>25</value>
        <value>1</value>
      </arguments>
      <stdout>
        <url>gsiftp://access.pcss.clusterix.pl/~demo2.out</url>
      </stdout>
    </executable>
  </simplejob>
</grmsjob>
Client applications

- Command line tool
- Web Portal
- Mobile Client
Client - command line tool

```
bogdani@access:/home/staff/bogdani/gclient

bogdani@access:~/gclient$ ./bin/ws_client.sh
- wrong configuration in grns script
GRNSClient USER Knox:
ws_client.sh submit <jobDescriptionFile>
ws_client.sh migrate <jobId> [ <jobDescriptionFile> ]
ws_client.sh suspend <jobId> [ <jobDescriptionFile> ]
ws_client.sh resume <jobId> [ <jobDescriptionFile> ]
ws_client.sh cancel <jobId>
ws_client.sh list [ QUEUED | PREPROCESSING | PENDING | RUNNING | STOPPED | POSTPROCESSING | FINISHED | SUSPENDED | FAILED | CANCELED]
ws_client.sh project <projectId> [ QUEUED | PREPROCESSING | PENDING | RUNNING | STOPPED | POSTPROCESSING | FINISHED | SUSPENDED | FAILED | CANCELED]
ws_client.sh list all QUEUES | PREPROCESSING | PENDING | RUNNING | STOPPED | POSTPROCESSING | FINISHED | SUSPENDED | FAILED | CANCELED
ws_client.sh register <jobId> <service_definition> <pid>
ws_client.sh unregister <jobId>
ws_client.sh access <jobId>
ws_client.sh info <jobId>
ws_client.sh memory <jobId>
ws_client.sh resources <resourceDefinitionFile>
ws_client.sh text <resourceDescriptionFile>
ws_client.sh add_notif <jobId> STATUS|REQUEST SOAP|GASS <destination> [true|false [format]]
ws_client.sh del_notif <jobId> <notificationId>
ws_client.sh list_notif <jobId>
ws_client.sh insert_notif <jobId> <notificationId>
ws_client.sh add_output <jobId> <name> <path> PHYSICAL|LOGICAL FILE|DIRECTORY
ws_client.sh get_output <jobId>
ws_client.sh del_output <jobId> [name] <path> PHYSICAL|LOGICAL FILE|DIRECTORY
ws_client.sh add_checkpoint <jobId> <name> <path> PHYSICAL|LOGICAL FILE|DIRECTORY
ws_client.sh set_checkpoint <jobId>
ws_client.sh del_checkpoint <jobId> <name> <path> PHYSICAL|LOGICAL FILE|DIRECTORY
ws_client.sh description SHORT|FULL
bogdani@access:~/gclient]
```
bogdanl@access:~/gclient$ cat ./examples/demo1v2.xml
<grmsjob appid = "mpichg2test" persistent="true">
  <simplejob>
    <resource>
      <hostname>access.pcss.clusterix.pl</hostname>
      <localrmname>pbs</localrmname>
    </resource>
    <executable type="mpichg" count="3">
      <file name="clx" type="in">
        <url>file:///tmp/clx_ia64_g2</url>
      </file>
      <arguments>
        <value>HOME/CLX/var/grms_demo1v2</value>
        <value>25</value>
        <value>1</value>
      </arguments>
      <stdout>
        <url>gsiftp://access.pcss.clusterix.pl/~/demo1.out</url>
      </stdout>
    </executable>
  </simplejob>
</grmsjob>
bogdanl@access:~/gclient$ ./bin/ws_client.sh submit ./examples/demo1v2.xml
- Your DN: /C=PL/O=GRID/O=PSNC/CN=Bogdan Ludwiczak
- Service URL: http://access.pcss.clusterix.pl:8443/axis/services/grms
- Job submitted successfully, jobId=1126383303899_mpichg2test_4733
bogdanl@access:~/gclient$
Client - portal
<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STDIN</td>
<td>collection</td>
<td>Name</td>
</tr>
<tr>
<td>STDOUT</td>
<td>collection</td>
<td>Name</td>
</tr>
<tr>
<td>STDERR</td>
<td>collection</td>
<td>Name</td>
</tr>
</tbody>
</table>

Warning: If you switch executable type from file to application, and commit changes, you will lose file information and vice versa.

9 wrznia 2005
Client - mobile devices
GRMS deployments

- **Polish projects:**
  - Clusterix - national Linux based Grid
  - SGI Grid
  - Progress Project

- **Other**
  - HPC-Europa
  - InteliGrid
  - GriPhyN
  - Canadian Grid Initiative - testing
  - Swiss Grid, DEISA, TeraGrid being negotiated/tested
More information

- [http://www.gridlab.org/grms](http://www.gridlab.org/grms)
- **Admin Guide - how to install GRMS**
- **User Guide - how to use GRMS**
- **Source Code**
  - Branch without workflow support (1.x)
  - Final version of GRMS (ver. 2.x)
  - [http://www.gridlab.org/grms/releases.html](http://www.gridlab.org/grms/releases.html)