San Francisco, California, USA

André Sailer, Marko Petric (CERN)
On Behalf of the CLICdp Collaboration

Motivation

The ILC VO is a merger of OSG and WLCG VOs for linear collider studies. Initial computing demands could be satisfied with the resources available in the WLCG via glide-WMS, CREAM, and ARC computing elements. However, the adoption of the iLCDirac tool by more and more members of the linear collider community, and the untapped but accessible resources in the OSG encouraged the development of the interfaces between DIRAC and the HTCondor-CE and Globus computing element middlewares.

DIRAC

The DIRAC interface gives homogeneous access to heterogeneous resources:

- Job environment provided through Pilots
- Common issues solved by DIRAC: Workload Management; central Productions; File Catalog, asynchronous Requests; access to different computing elements, batch systems, or clouds
- Can be extended to meet needs specific to VOs

Further information:

- DIRAC in Large Particle Physics Experiments; Oct 13, 14:00 Track 7
  - diracgrid.org, http://github.com/DIRACGrid

Using HTCondor-CEs inside DIRAC

DIRAC Computing Element API

- Computing Element classes need to fulfill the following interface
  - submitJob: Submit one or many pilots to a given CE
  - getJobStatus: Status of individual job on CE
  - getJobOutput: Get pilot output and error file
  - killJob: kill the (pilot) job
  - getPilotLoggingInfo: Get log for pilot

Submission and Configuration

- condor_submit --terse subFile.jdl
- m.setInputFile("SimEvents.slcio")
- m.setSteeringFile("Steering.xml")
- m.setOutputSandbox("recEvents.slcio")
- m.submit(d)

Requirements

- DIRAC instance
- Running condor daemons (condor_master) on one of the DIRAC servers
- DIRAC Workload Management system takes care of matching payloads to jobs, pilot status is monitored automatically

Implementation

- The HTCondor-CE computing element class is implemented by calling plain condor commands
- Using raw commands for easier debugging; keep DIRAC and condor issues separate

Job Control

DIRAC Workload Management system takes care of matching payloads to jobs, pilot status is monitored automatically

- Environment variable HTCONDOR_JOBID, unique for each pilot, is used to identify pilots
- condor_q and condor_history to obtain pilot status
- condor to kill and remove held pilots to prevent restart
- Output log files automatically downloaded by HTCondor and passed into DIRAC on demand (see Monitoring to the right)

Problems and Open Issues

- Pilots and payloads do not support check-pointing, if a job is held it has to be killed
- periodic_remove needs to be tweaked for specific CEs
- Monitoring CE status: Query the CE how many jobs from given VO are running, currently using DIRAC’s own count of pilots at CE

Monitoring

- Monitor pilot status, read log files and output via the DIRAC WebApp

OSG Resource Usage by the ILC VO

CPU used by Grid

<table>
<thead>
<tr>
<th>Week</th>
<th>2015</th>
<th>2016</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Mar 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Jun 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Sep 1</td>
<td>2015</td>
<td>2016</td>
</tr>
</tbody>
</table>

CPU used by Site

<table>
<thead>
<tr>
<th>Week</th>
<th>2015</th>
<th>2016</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Mar 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Jun 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Sep 1</td>
<td>2015</td>
<td>2016</td>
</tr>
</tbody>
</table>

Jobs by Site

<table>
<thead>
<tr>
<th>Week</th>
<th>2015</th>
<th>2016</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Mar 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Jun 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Sep 1</td>
<td>2015</td>
<td>2016</td>
</tr>
</tbody>
</table>

Global CE

- Link from DIRAC to Globus CEs implement via globus-job-** commands
- globus-job-submit, globus-job-clean, globus-job-status, globus-job-get-output
- Number of Globus CEs is decline since HTCondor-CE available
- Only one (1) left for the ILC VO

Summary

- OSG Computing Elements (HTCondor-CE, Globus) fully integrated in DIRAC
- Minimal effort for Dirac instance administrators to use HTCondor-CEs or Globus CEs
- Completely transparent to end users

Acknowledgement

Thanks to Iain Steers (CERN), the HTCondor team, and the DIRAC developers for their advice and support.

WWW: http://ilcdirac.cern.ch

Mail: andre.sailer@cern.ch, marko.petric@cern.ch

Heterogeneous resources
andre.sailer@cern.ch, marko.petric@cern.ch

San Francisco, California, USA

San Francisco, California, USA

André Sailer, Marko Petric (CERN)
On Behalf of the CLICdp Collaboration

Motivation

The ILC VO is a merger of OSG and WLCG VOs for linear collider studies. Initial computing demands could be satisfied with the resources available in the WLCG via glide-WMS, CREAM, and ARC computing elements. However, the adoption of the iLCDirac tool by more and more members of the linear collider community, and the untapped but accessible resources in the OSG encouraged the development of the interfaces between DIRAC and the HTCondor-CE and Globus computing element middlewares.

DIRAC

The DIRAC interface gives homogeneous access to heterogeneous resources:

- Job environment provided through Pilots
- Common issues solved by DIRAC: Workload Management; central Productions; File Catalog, asynchronous Requests; access to different computing elements, batch systems, or clouds
- Can be extended to meet needs specific to VOs

Further information:

- DIRAC in Large Particle Physics Experiments; Oct 13, 14:00 Track 7
  - diracgrid.org, http://github.com/DIRACGrid

Using HTCondor-CEs inside DIRAC

DIRAC Computing Element API

- Computing Element classes need to fulfill the following interface
  - submitJob: Submit one or many pilots to a given CE
  - getJobStatus: Status of individual job on CE
  - getJobOutput: Get pilot output and error file
  - killJob: kill the (pilot) job
  - getPilotLoggingInfo: Get log for pilot

Submission and Configuration

- condor_submit --terse subFile.jdl
- m.setInputFile("SimEvents.slcio")
- m.setSteeringFile("Steering.xml")
- m.setOutputSandbox("recEvents.slcio")
- m.submit(d)

Requirements

- DIRAC instance
- Running condor daemons (condor_master) on one of the DIRAC servers
- DIRAC Workload Management system takes care of matching payloads to jobs, pilot status is monitored automatically

Implementation

- The HTCondor-CE computing element class is implemented by calling plain condor commands
- Using raw commands for easier debugging; keep DIRAC and condor issues separate

Job Control

DIRAC Workload Management system takes care of matching payloads to jobs, pilot status is monitored automatically

- Environment variable HTCONDOR_JOBID, unique for each pilot, is used to identify pilots
- condor_q and condor_history to obtain pilot status
- condor to kill and remove held pilots to prevent restart
- Output log files automatically downloaded by HTCondor and passed into DIRAC on demand (see Monitoring to the right)

Problems and Open Issues

- Pilots and payloads do not support check-pointing, if a job is held it has to be killed
- periodic_remove needs to be tweaked for specific CEs
- Monitoring CE status: Query the CE how many jobs from given VO are running, currently using DIRAC’s own count of pilots at CE

Monitoring

- Monitor pilot status, read log files and output via the DIRAC WebApp

OSG Resource Usage by the ILC VO

CPU used by Grid

<table>
<thead>
<tr>
<th>Week</th>
<th>2015</th>
<th>2016</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Mar 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Jun 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Sep 1</td>
<td>2015</td>
<td>2016</td>
</tr>
</tbody>
</table>

CPU used by Site

<table>
<thead>
<tr>
<th>Week</th>
<th>2015</th>
<th>2016</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Mar 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Jun 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Sep 1</td>
<td>2015</td>
<td>2016</td>
</tr>
</tbody>
</table>

Jobs by Site

<table>
<thead>
<tr>
<th>Week</th>
<th>2015</th>
<th>2016</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Mar 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Jun 1</td>
<td>2015</td>
<td>2016</td>
</tr>
<tr>
<td>Sep 1</td>
<td>2015</td>
<td>2016</td>
</tr>
</tbody>
</table>

Global CE

- Link from DIRAC to Globus CEs implement via globus-job-** commands
- globus-job-submit, globus-job-clean, globus-job-status, globus-job-get-output
- Number of Globus CEs is decline since HTCondor-CE available
- Only one (1) left for the ILC VO

Summary

- OSG Computing Elements (HTCondor-CE, Globus) fully integrated in DIRAC
- Minimal effort for Dirac instance administrators to use HTCondor-CEs or Globus CEs
- Completely transparent to end users

Acknowledgement

Thanks to Iain Steers (CERN), the HTCondor team, and the DIRAC developers for their advice and support.

WWW: http://ilcdirac.cern.ch

Mail: andre.sailer@cern.ch, marko.petric@cern.ch