Project Name: **g- Eclipse** (*www.eclipse.org/geclipse*)
Contact:
Date: **October 5ᵗʰ , 2007**

Introduction

The g- Eclipse project aims to build an integrated workbench framework to access the power of existing Grid infrastructures. The framework is built on top of the reliable eco- system of the Eclipse community to enable a sustainable development. The framework will provide tools to customize Grid users' applications, to manage Grid resources and to support the development cycle of new Grid applications. The g- Eclipse framework will be middleware agnostic and its architecture is designed to *extend it for many different Grid middlewares (such as gLite, UNICORE, Globus toolkit), starting with implementations for the gLite middleware. As of October 2007, the g- Eclipse project will initiate support for the GRIA middleware.*

The g- Eclipse framework aims for full support of the JSDL standard to be independent from the underlying Grid middleware. JSDL is the main description language used in g- Eclipse. Therefore the g- Eclipse team developed wizards and a multipage editor for JSDL files.

Job Description Wizard

With the help of a JSDL wizard provided by the g- Eclipse framework, the user can easily create the functional skeleton of a JSDL file. Futher details can be entered with the help of the JSDL editor, which will automatically be opened, when the wizard is finished.

JSDL Editor

The g- Eclipse project developed a user- friendly, fully functional JSDL editor for editing JSDL documents. The editor follows Eclipses' multi- page editor style and consists currently of 5 pages (Overview, Job Definition, Application, DataStaging and Resources) plus a page showing the resulted XML file of the JSDL document (changes here will be reflected in the other pages).

NOTE: As of September 31ˢᵗ 2007, **g- Eclipse 0.5.0** is publicly available. g- Eclipse is an official technology project at the Eclipse Foundation.

Implemented JSDL Elements:

Please check the elements that you implemented in the table below. Additional comments are welcome. For example, problems implementing an element; restrictions placed beyond what was specified in the specification; reasons for not implementing a feature.

JSDL Element	Yes	No	N/A	Comments
JobIdentification				
JobName	X			
JobAnnotation	X			
JobProject	X			
Application				
ApplicationName	X			
ApplicationVersion	X			
Resources				
CandidateHosts	X			
HostName	X			
FileSystem		X		Will be implemented in next milestone releases.
MountPoint		X		Will be implemented in next milestone releases.
MountSource		X		Will be implemented in next milestone releases.
DiskSpace		X		Will be implemented in next milestone releases.
FileSystemType	X			
ExclusiveExecution		X		Will be implemented in next milestone releases.
OperatingSystem	X			
OperatingSystemType	X			
OperatingSystemName	X			
OperatingSystemVersion	X			
CPUArchitecture	X			
CPUArchitectureName	X			
IndividualCPUSpeed		X		Partially implemented.
IndividualCPUTime		X		Partially implemented.
IndividualCPUCount		X		Partially implemented.
IndividualNetworkBandwidth		X		Partially implemented.

JSDL Element	Yes	No	N/A	Comments
IndividualPhysicalMemory		X		Partially implemented.
IndividualVirtualMemory		X		Partially implemented.
IndividualDiskSpace		X		Partially implemented.
TotalCPUTime		X		Partially implemented.
TotalCPUCount		X		Partially implemented.
TotalPhysicalMemory		X		Partially implemented.
TotalVirtualMemory		X		Partially implemented.
TotalDiskSpace		X		Partially implemented.
TotalResourceCount		X		Partially implemented.
DataStaging				
FileName	X			
FilesystemName		X		Will be implemented in next milestone releases.
CreationFlag	X			
DeleteOnTermination	X			
Source	X			
Target	X			
POSIXApplication				
Executable	X			
Argument	X			
Input	X			
Output	X			
Error	X			
WorkingDirectory	X			
Environment	X			
WallTimeLimit	X			
FileSizeLimit	X			
CoreDumpLimit	X			
DataSegmentLimit	X			
LockedMemoryLimit	X			
MemoryLimit	X			
OpenDescriptorsLimit	X			
PipeSizeLimit	X			
StackSizeLimit	X			
CPUTimeLimit	X			
ProcessCountLimit	X			
VirtualMemoryLimit	X			
CPUTimeLimit	X			
ProcessCountLimit	X			

JSDL Element	Yes	No	N/A	Comments
VirtualMemoryLimit	X			
ThreadCountLimit	X			
UserName	X			
GroupName	X			
HPCProfileApplication				
Executable			X	Currently there is no plan for implementation of HPC Profile Applications.
Argument			X	
Input			X	
Output			X	
Error			X	
WorkingDirectory			X	
Environment			X	
UserName			X	

Other problems encountered:

Did you have any problems with the specification besides comments you may have added to the table above?

Mappings to existing systems:

If you mapped JSDL to an existing system (batch or otherwise), please provide a list of systems you mapped to (with online pointers if available).
Were there any specific problems mapping to any of these systems?

With the help of the g-Eclipse framework, the JSDL files can easily been submitted to any gLite based Infrastructure like EGEE. For the submission, a working webservice based Workload Management System (version 3.1) of gLite is required.

With the integration of support for the GRIA middleware, the same approach will be followed.

Enhancements:

If you extended JSDL with your own features, please provide a list; and a short description or online pointer to features you added.

Participation in interoperability tests:

NO

Security:

What security model did you use with JSDL? For example, how did you secure submissions of JSDL documents?
Did you include any security information in JSDL documents?
Were there any problems combining JSDL with your security solution?

g-Eclipse expects a Job Sumbmission service (Resource Broker, ...) including a Grid security model. Currently, g-Eclipse provides the VOMS based security system based on X.509 certificates and delegated proxies. The g-Eclipse framework checks before jobsubmission if an valid security token exists. If not, the user is asked to create the requested security token (i.e. a VOMS proxy).