



Contribution ID: 27

Type: Poster

## Building Grid client applications with the g-Eclipse framework

The g-Eclipse is an integrated workbench framework to access the power of existing Grid infrastructures. g-Eclipse can be used on user level or application level. On user level g-Eclipse is just a rich client application with user friendly interface which allows users to access Grid resources, operators to manage Grid resources and developers to speed up the development cycle of new Grid applications. On application level the g-Eclipse components can be used as a base for writing customised grid application using g-Eclipse model and g-Eclipse common grid library.

There are three ways to prepare and manage grid applications job with g-Eclipse:

- provide XML application description,
- enhance g-Eclipse with specific application editor and submission service,
- prepare Rich Client Platform (RCP) application using g-Eclipse as a common grid library.

g-Eclipse supports the first approach by offering XML-based application descriptions and corresponding tools to specify application parameters in much more friendly way than just filling "Executable" and "Arguments" fields in JDL or JSDL. g-Eclipse offers tools to share such job description documents within application communities by using repositories like CVS or SVN. The same is valid for workflow description files. With the help of the g-Eclipse framework application group manager or VO manager can prepare applications, both simple or workflow, and share them with the rest of the group.

The second approach enhances the g-Eclipse framework with new plugins for application specific job description languages and submission methods. Many applications use job description files in the specific format (e.g. running Gaussian jobs requires ".inp" file with all data). The new plugins registers such files as grid job descriptions and supporting editors (textual or graphical ones) which can handle this file format. Using the existing g-Eclipse infrastructure, such job description are submitted to the Grid by transforming the application specific description in a middleware specific one.

The last method is the most advanced and can be used to prepare custom application. By extensively using the Eclipse extension mechanism in combination with object-oriented design patterns, the framework can be easily extended by application-specific implementations. The application is not plugged into g-Eclipse, but rather built on top of it. It follows the approach of the Rich Client Platform (RCP) of Eclipse. Dedicated applications can be build by using parts of g-Eclipse as a common library to handle resources and grid access. RCP application can use the g-Eclipse core model only, or can use some GUI components, too. With such an approach existing applications can be gridified. The g-Eclipse core model is grid middleware independent. By electing the a appropriate set of plug-ins, the same application can access resources from different grid middleware. The g-Eclipse model defines interfaces for accessing virtual organisations, information system, job submission services, job monitoring services and grid file systems. There is also support for managing authentication tokens, storing submitted jobs locally, automatic notifications about job statuses and many others. If the results of computation needs to be visualised then another plug-in can be added that will handle the output file format and will visualise it, possibly using g-Eclipse visualisation.

g-Eclipse also include a Developer Perspective which contain tools that can be used while developing grid application for seamless integration of the remote development process into the existing Eclipse development environments. This include remote debugging support, application monitoring support, visualisation support, and application deployment.

**Primary author:** WOLNIEWICZ, Pawel (PSNC)

**Co-author:** GARCIA, Ariel (FZK)

**Presenter:** WOLNIEWICZ, Pawel (PSNC)

**Track Classification:** 1. Computing Technology