

Adding Standards Based Job Submission to a Commodity Grid Broker

David Colling, A. Stephen McGough, Jazz Mack Smith and Vesso Novov
Imperial College London, UK
d.colling@imperial.ac.uk, {asm, jms3, vesso}@doc.ic.ac.uk

Tiejun Ma, David Wallom and Xin Xiong
University of Oxford, UK
{Tiejun.ma, david.wallom, Xin.Xiong}@oerc.ox.ac.uk

Abstract

The Condor matchmaker provides a powerful mechanism for matching together both users job requirements and resource providers requirements in such a manner that not only is a paring selected which satisfies both requirements but is optimal for both. This has made the Condor system a good choice for use as a meta-scheduler within the Grid. Integrating Condor within a wider Grid context is possible but only through modification to the Condor source code. Here we describe how the standards for job submission and resource descriptions can be integrated into the Condor system to allow arbitrary Grid resources which support these standards to be brokered through Condor.

1. Introduction

The Grid provides a platform in which owners of *resources* can expose these for consumption potentially for rewards. In order for users to discover resources for their *jobs* and resource owners to obtain significant jobs a brokering service is required. A resource owner, who may manage their resources through a Distributed Resource Manager (DRM), will *advertise* their resources to a *Grid brokering service*. Likewise users will *submit* requests to a broker. The broker then *selects* resources which best match both the users and the resource owners requirements. The broker may also provide mechanisms to *monitor* the progress of the users computation on the resource and *transfer files* to and from the resource for the user.

As the number of both sites and the number of resources within these sites increase the problem of selecting the '*best*' resource to use becomes more complex. This complexity comes not only from the scale of the problem but the heterogeneity of the technologies used. Vendors solutions exist (UNICORE [3], gLite [1]), however, in general these

solutions present a tie-in to their particular approach. The use of open standards removes this tie-in. The standards work within the Open Grid Forum (OGF) [5] has lead to a number of standards for resource discovery (GLUE [2]), job description (JSDL) [4] and submission interface (BES) [6].

Condor [8] was originally developed as a batch job execution service for the purpose of high-throughput computing, utilising unused (idle) computing power within a collection of computers. The Condor broker is considered to be one of the best available due to its use of the ClassAds [10] description language coupled with the Condor Matchmaker service [7]. This allows both the user and the resource provider to match on hard requirements such as processor and also rank those matching resources (or users). Although Condor does not support standards a number of Grid resources are integrated through the use of the Grid ASCII Helper Protocol (GAHP [9]) interface and the simple mechanism provided in Condor for adding resource adverts. However, modification is required to the Condor source code to add a new resource system.

We have integrated a set of standards based interfaces into Condor allowing for integration with any standards compliant DRM. Thus the DRM system can be used with Condor without the need to modify the Condor source code – providing a clean and clear interface.

2. Grid Standards

GLUE: The Grid Laboratory Uniform Environment (GLUE [2]) schema is an information model used to describe the features and status of a particular resource within a Grid environment. It has been designed such that it is independent of the implementation to allow for interoperability between solutions. The overall schema is extremely complex and deals with a single site being made up of several different resources. Here we have considered only the representation of a single computational resource.

JSDL: The Job Submission Description Language (JSDL) [6] describes a job that a user (agent) wishes to have executed. The language is an open contents template document, describing what is required rather than how to achieve it. The job description is separated into four main sections those of job identification, application description, resource requirements and data staging.

BES: The Open Grid Services Architecture - Basic Execution Service (OGSA-BES [6]), is a partner specification to JSDL. While JSDL defines the language used to describe a job, the BES specification describes how a service can consume such a document and how to provide monitoring and job control. The BES specification defines a Web Services [11] interface for (job) submission and monitoring.

3. Grid Standards Architecture

Figure 1 illustrates the architecture which has three main components to support the Broker: advertising of resources, deployment of jobs and file staging. The user submits jobs to the broker using the Condor submission tools. However, a BES interface could be placed in-front of Condor for a complete standards compliant solution.

Advertising of Resources: Frequent information collection is essential to ensure accurate scheduling decisions by the broker. The standard method for achieving this is through queries to the resource's information system, such as an LDAP-based information server (BDII), or an XML-based resource information registry (Grimoies [12]). The retrieved information can be translated into a ClassAd for submission to the Condor Collector. We have also designed and built a webservice provider of GLUE information for the Microsoft HPC Server system.

Job Submission and Monitoring: A new JSDL GAHP client has been developed which runs within Condor. This is a thin client which passes the ClassAd to the JSDL server. The JSDL server translates the ClassAd into a JSDL document which it submits to a BES enabled service.

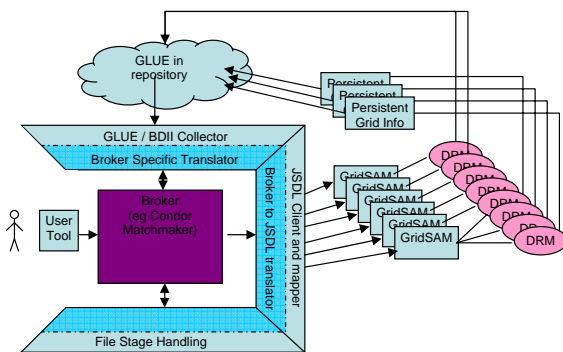


Figure 1. Grid Brokering System Architecture

Data Staging: Condor lists those files that need to be staged to and from the resource and this information is mapped to the JSDL document. If the location where the user has placed their files is already exposed through a file transfer protocol (such as FTP, GridFTP, HTTP(S)) then these locations can be used otherwise the file is copied to a location where it can be accessed.

4. Conclusion

We have shown how the Condor system can be integrated with standards for job submission (JSDL and BES) and resource description (GLUE). This allows a powerful brokering service to be used with resources exposed through standards based interfaces. This is a more scalable solution than using GAHP alone as different DRM systems need only develop a BES interface over their existing software rather than developing both a Client and Server interface for use with Condor. This provides the added advantage that their BES interface can be used by others on the Grid.

References

- [1] EGEE. gLite Middleware. <http://www.glite.org>, Aug 2008.
- [2] GLUE-WG. The GLUE working group. <https://forge.gridforum.org/sf/projects/glue-wg>, May 2008.
- [3] S. Haubold, H. Mix, W. Nagel, and M. Romberg. *Performance Analysis and Grid Computing*, chapter The UNICORE Grid and its Options for Performance Analysis, pages 275–288. Kluwer Academic Publishers, 2003.
- [4] JSDL-WG. Job submission description language. <https://forge.gridforum.org/sf/projects/jsdl-wg>, May 2008.
- [5] OGF. The Open Grid Forum. <http://www.ogf.org>, May 2008.
- [6] OGSA-BES-WG. Basic execution service. <https://forge.gridforum.org/sf/projects/ogsa-bes-wg>, May 2008.
- [7] M. S. Rajesh Raman, Miron Livny. Matchmaking: Distributed resource management for high throughput computing. In *Proceedings of the Seventh IEEE International Symposium on High Performance Distributed Computing*, page 140, Chicago, IL., July 1998.
- [8] C. Team. Condor Project Homepage. <http://www.cs.wisc.edu/condor>.
- [9] The Condor Project. Grid ASCII Helper Protocol. <http://www.cs.wisc.edu/condor/gahp/>.
- [10] The Condor Project. Classads. <http://www.cs.wisc.edu/condor/classad/>, Aug 2008.
- [11] W3C Consortium. Web Services Description Language (WSDL) 1.1. <http://www.w3.org/TR/wsdl>.
- [12] S. Wong, V. Tan, W. Fang, S. Miles, and L. Moreau. Grimoires: Grid registry with metadata oriented interface: Robustness, efficiency, security. *IEEE Distributed Systems Online*, 6(10), 2005.