

# Task and Resource Descriptions for Computational Grids

I. Stokes-Rees, University of Oxford

## Abstract

A common requirement of computational grids is the need to describe tasks and resources to facilitate scheduling. This paper describes the features necessary for successful task management, contrasts various existing approaches, and proposes a unified extensible description language.

A new syntax is presented which addresses the symmetry of task and resource matching, embedded authorization information, billing, temporality, and task life-cycle management. This syntax is the basis for generic, scalable, grid process management. It has been developed from experience of high-volume task management in the CERN-based LHCb particle physics experiment.

The nature of computational grids is such that it is rarely possible to simply name an executable, some arguments, and a set of input in order to perform the desired computational task. In a standard time sharing system (i.e. on a typical workstation or server) such operations are possible because: they are synchronous and block until the operation completes; they can utilise the underlying locally accessible data store; any user environment settings or configuration is directly available; and, the capabilities and configuration of the local system are general known and relatively static.

Batch systems and computing clusters have required a more detailed description of the task in order to prioritise it properly, provide the appropriate configuration and data, and correctly match resource requirements. Grid computing has introduced the further complexity of externally-controlled heterogeneous computing resources and widely distributed data sets. When operating in a grid environment it is also necessary to carefully consider the mixture of security domains requiring different authorizations.

Ultimately, it is necessary to encapsulate the key characteristics of the computational task and the requirements for performing that task. This includes all necessary input, environment, software, timing constraints, authorization tokens, and output. In a similar fashion the computing resource must be satisfactorily described in order to determine if a suitable fit with a task can be

found. This must describe performance, storage, memory, data availability, access policy, and installed software, among other things.

Existing description languages have typically been for specific software packages (RSL, JCL, JDL) or so generic as to lack any semantic structure allowing checking and validation (ClassAds). Work underway by the GGF is close to releasing the XML-based Job Submission Description Language (version 1.0), however this has a very limited scope and only applies to Job Description, ignoring resource description.

The experience of the CERN LHCb grid software group has been that the description language needs a verifiable syntax, strong semantics, and still be extensible. This naturally points to an XML based syntax utilising namespaces to accommodate extensions. Further work by the author into scalable computational grid infrastructures has led to the approach of encapsulating not just initial task characteristics in the description but to allow this to be the mechanism to represent grid process state throughout the task life cycle. The same dynamic nature holds for resource descriptions.

The proposed generic description language broadly identifies task and resource sets consisting of explicitly time-limited descriptions which internally specify access control constraints and identify the set of authorization tokens required to utilise the description. They contain a set of characteristics (about themselves) and a set of requirements (about the task or resource they are prepared to interact with). A flexible billing and accounting system is incorporated. Tasks are further described with an input data set which must be available in advance of any computation, a sequence of computations, and finally an output data set which is preserved to permanent and grid-accessible storage. Details and features of this approach are discussed.

This paper presents the existing description languages for tasks and resources and outlines the requirements deemed necessary for symmetric matchmaking for computational grids. Three novel characteristics are incorporated into the proposed new description language: explicit support for multiple identity tokens; representation of task state throughout the task life-cycle; and, a billing mechanisms for data access and computing utilisation.