Integrating computing, data and collaboration grids: the RMCS tool

BY A. M. WALKER1,*, R. P. BRUIN1, M. T. DOVE1, T. O. H. WHITE1, K. KLEESE VAN DAM2 AND R. P. TYER2

1Department of Earth Sciences, University of Cambridge, Downing Street, Cambridge CB2 3EQ, UK
2Daresbury Laboratory, Daresbury Science and Innovation Campus, Science and Technology Facilities Council, Warrington WA4 4AD, UK

We describe RMCS as one of the first tools for grid computing that integrates data and metadata management into a single job submission system. The system is easy to use, with client tools that are easy to install. Although the RMCS system was developed as a prototype, it is now in production use and a number of scientific studies have been completed using it.

Keywords: eScience; eMinerals; grid computing; RMCS

1. Introduction: the challenges of exploiting grid systems

The advent of grid computing, facilitated in part by the availability of powerful yet low-cost hardware, provides simulation scientists with unprecedented opportunities. Tasks that, in previous decades, could be run only on high-performance facilities can now be run on the latest generation desktop processors, and thus it becomes practical to perform large studies that involve hundreds to millions of individual jobs. However, even when a basic grid infrastructure is in place there are a number of challenges to the researcher as follows.

— Choice between the available grid resources, management of the high-level load balancing and selection of an executable for the chosen resource.
— Complexity of common grid middleware. We believe that the requirement of user installation and direct usage of GLOBUS (Foster & Kesselman 1997) or similar tools is too high a barrier for researchers. Job submission should be easy from the researcher’s desktop or laptop.
— Management of the deluge of data and metadata that can be generated by grid-based science. This implies the need for close integration of computing and data management systems, to enable the data from large multi-job studies to be properly, automatically and flexibly archived after the job runs.

* Author for correspondence (a.walker@ucl.ac.uk).

One contribution of 24 to a Discussion Meeting Issue ‘The environmental eScience revolution’.
Multiple job management tools are provided across grid resources; even on one resource serial and parallel job submission may be different. An interface is needed to allow the researcher to access all these on the same footing.

The MCS system was developed to meet some of these challenges but this left the user with the need to manage a complex network configuration and install GLOBUS and CONDOR (Tannenbaum et al. 2001) on the submit machine (Bruin 2007; Bruin et al. 2008). In order to remove these limitations, we developed a Web services version of this system named RMCS, described below (Dove et al. 2007).

2. RMCS architecture and use

The architecture of the RMCS system, as illustrated in figure 1, has three tiers. Client tools interact with the RMCS server via Web services calls. These only involve outbound network connections to a single port on the server, simplifying network configuration and allowing use from within most firewalled environments. The server layer includes a relational database that stores information about the jobs, the Web services server that inserts jobs into the database and provides information on job status and a series of agents that interact with the database and remote grid resources. The agents include an updated version of MCS. This architecture allows scaling towards the concurrency limit of the chosen database server by deploying several machines in parallel at the Web services and agent layers. However, in practice, we have found that deploying all three layers on a single machine is sufficient to manage many hundreds of concurrent tasks, at which point limitations of the data and compute grid resources are exposed. The server...
tier removes from the user the need to install tools, such as GLOBUS or CONDOR, on the computer from which they are submitting jobs. Because the client–server interaction is via Web services, RMCS is client agnostic. The reference implementation is a set of highly portable command line tools and these are used by most researchers. There is also a Java GUI, PYTHON API and the opportunity for workflow engines to interact directly with the server.

The integration of the use of compute and data grids presupposes a work process for the user concerning handling the data and files. Here, we assume that the user is submitting a task to RMCS through the command line interface. The work process has seven steps, as illustrated in figure 2. (i) The researcher uploads the data files and executables to the data grid. In the case of the eMinerals project (Dove & de Leeuw 2005) this has been the Storage Resource Broker (SRB; Baru et al. 1998), although RMCS is not constrained to work only with the SRB. (ii) The researcher uses the rmcs_submit command with a job description file as the argument. This file contains a set of directives concerning information such as the name of the executable, the type of resource to be used (single or multiprocessor) and the location within the data grid for the input and output files. (iii) RMCS transfers the files from the data grid to the selected grid resource. (iv) The job is run on the selected grid resource. (v) The output XML file is parsed using the AGENTX tool to extract the metadata (White et al. 2006). The metadata are uploaded using the RCOMMANDS tool (Tyer et al. 2007). (vi) The output files are sent to the data grid. (vii) The user can view the output files in the data grid and the metadata. It may be the easiest to locate the output files from the metadata, and the metadata interface can be used to extract tables of output data (Tyer et al. 2007).

Figure 2. RMCS/user workflow highlighting the transfer of the data and the metadata.
We note that in the case of running multiple jobs, steps (i) and (ii) can be performed as a part of an automated process using the MONTY tool (Dove et al. 2007). RMCS has been used for a number of scientific studies (e.g. Dove et al. 2006; Thomas et al. 2007; Walker et al. 2007).

We acknowledge support from NERC under the eScience thematic programme.

References


