Developing LHCb Grid Software: Experiences and Advances

I. Stokes-Rees, A. Soroko, C. Cioffi, University of Oxford
A. Tsaregorodtsev, V. Garonne, CPPM
R. Graciani, University of Barcelona
M. Sanchez, University of Santiago de Compostela
M. Frank, J. Closier, CERN
G. Kuznetsov, RAL

Abstract

The LHCb grid software has been used for two Physics Data Challenges, the most recent of which will have produced 50 TB of data and required over 500 processor-years of computing power. This presentation discusses the group's experience with developing Grid Services, interfacing to the LCG, running LHCb experiment software on the grid, and the integration of a number of new technologies into the LHCb grid software. Our experience and utilisation of the following core technologies will be discussed: OGSI, XML-RPC, Web Services, LCG middleware, Jabber Instant Messaging, GridSite, Clarens, GSI, Globus, and Condor ClassAds.

The LHCb experiment has evolved its physics simulation and analysis software from largely standalone Python-based Agents and Services, distributed across various physics institutes, into a *grid-aware* package which makes use of numerous Grid middleware developments. It has incorporated advances in the available grid middleware to make use of LCG computing resources and functionality, while also, critically, allowing the integration of non-LCG computing resources. This presentation will recount our experience of integrating our software into the LCG grid environment, making use of existing middleware services and libraries, and advances from incorporating new technology such as Instant Messaging into our architecture.

The latest version of DIRAC builds on earlier successes of an Agent based physics simulation system using job pools. This new version embraces a **Service Oriented Architecture**, with decomposition of the system into independent services which interoperate through well defined APIs. Independent services increase simplicity, which improves robustness, allows distribution of services across machines for better load balancing, and facilitates

rapid development and deployment. We investigated numerous routes to service development, including OGSI, XML-RPC, Web Services, and the CMS project's Clarens Web Services Framework. We will discuss our experiences with each of these.

Further to our goal of independent, interoperating grid services, as proposed by OGSA, we have incorporated the Jabber Instant Messaging Framework into DIRAC. This allows Services to asynchronously communicate between themselves, and for Agents and Clients to contact Services. A number of features such as performance, security, development, and deployment of Instant Messaging for Grid Services will be discussed.

Appropriate integration of security into the DIRAC software has been challenging. The use of GSI, Service vs. Host vs. User certificates, Access Control Lists, and security policies will be discussed. To support this security infrastructure systems such as GridSite and GACL have been explored, as well as authentication mechanisms for file and service access.

We will present the current status of both the project software, and the performance of the DIRAC system for the 2004 LHCb Data Challenge, which has the triple goals of producing LHCb physics simulation data, evaluation the physics software, and stress testing the grid hardware and software. DIRAC software will be installed at dozens of sites around the world, both LCG and non-LCG. The practical experience of deployment and operation will be discussed.

Finally, we will conclude with a summary of how DIRAC fits in to the EGEE and ARDA project blueprint for Grid Services development, our requirements for LCG in the future, and the integration of new and upcoming technologies such as IM, P2P, and the predecessor to OGSI, WSRF.