SAM-Grid: Using SAM and Grid Middleware to Enable Full Function Grid Computing

Andrew Baranovsk [1], Iain Bertram[2],Gabriele Garzoglio[1], Lee Lueking[1], Igor Terekhov[1], Rodney Walker[3]

[1]*Fermilab, Batavia, Illinois, USA*
[2]*Lancaster University, Lancaster, UK*
[3]*Imperial College, London, UK*

Key words to describe the work: Replication, catalogue, production, consumption, delivery, cache, grid.

Key Objectives: Incorporate standard Grid middleware into the production quality SAM system, to enable existing data management functionality to be used in a truly distributed computing environment.

Motivation for the work (problems addressed): Many solutions exist for the distributed production of simulation data for HEP experiments, and subsequent storage to MSS – roughly 1 system per experiment. All such *production* is characterized by negligibly small input data, CPU intensive processing, then the storage of O(GB) data files to central MSS. We believe *production* is easy. The real challenge is the distributed *consumption* of large data sets, by physics analysis processes. *Consumption* is characterized by a large input data set (many GB), which must be reliably delivered to the process. SAM uniquely addresses this complex problem.

1. Introduction

The SAM system[1] was developed at Fermilab to accommodate the high volume data management requirements for Run II Physics, and to enable streamlined access and mining of these large data sets. SAM stands for "Sequential Access to data via Metadata" where the "sequential" refers to the layout of physics events stored within files, which are in-turn stored sequentially on tapes within a Mass Storage System.

SAM is largely devoted to transparently delivering and managing caches of data. It is the sole data management system in use by the Dzero experiment at Fermilab; other experiments are either considering or planning its use in the near future. SAM is designed with a distributed architecture, using CORBA as its underlying framework. This has enabled the system to scale to meet the data distribution needs of the Dzero Collaboration which includes nearly 80 institutions and over 600 Physicists located all over the globe. Metadata and configuration information for the entire system is currently maintained in a central Oracle database repository at Fermilab, and as the system matures, it is planned to distribute this function in a way that will reduce latencies and make the overall operation more reliable. Data from the Dzero detector and Monte Carlo simulations are likely to require more than half a Petabyte of permanent mass storage within the next two years.

The CDF experiment at Fermilab is planning to use SAM, and will be heavily involved in the Grid effort at the lab. The DZero and CDF experiments are well underway at Fermilab and their data handling needs are being provided by SAM. Our challenge is to transition seamlessly this successful effort to the broader Grid environment, while continuing to provide reliable service.

2. SAM-Grid

Based on the success of SAM, a larger architecture has been conceived that encompasses Grid level job submission and management, and information services. The principal novelties of the larger architecture being developed, when compared with the European Data Grid architecture and other Grid projects include:

1. Job Definition and Management: The preliminary job management architecture is aggressively based on the Condor technology provided by through our collaboration with University of Wisconsin CS Group.
2. Monitoring and Information Services: We assign a critical role to this part of the system and widen the boundaries of this component to include all services that provide, or receive, information relevant for job and data management.
3. Data Handling: The existing SAM Data Handling system, when properly abstracted, plays a principal role in the overall architecture and has direct effects on the Job Management services.

A major goal of this project is to move toward a more grid-like architecture, using the emergent and standardized Grid components and protocols without jeopardizing production quality service for the ongoing experimental physics program. It is important to employ the current data handling system because it has been hardened under the stress of heavy use for the last 3 years and includes numerous features required in the HEP environment. Through Dzero's collaboration with PPDG we are able to work closely with the Condor team, and a major goal is to employ their technology within our system to its fullest extent, as opposed to building our own. We are also using Globus tools, such as MDS, GSI, GridFTP, and GRAM throughout the system.

A well-defined data processing model has emerged in the SAM project and it is desirable to continue to employ this. The experiments at Fermilab have globally distributed virtual organizations - collaborations of universities and research institutions - and it is vital that the needs and constraints of these sites be considered. Since many of these sites support research programs beyond those of Fermilab, for example LHC experiments now under development, it is imperative that this system comply with standards and provide the needed features to access data and perform processing in an opportunistic way, without extensive setups or installations at any site. Cross function with other grids, like EDG, must be supported. Finally, the schedule for this project is short because the needs of the experiments involved are large and imminent.

The most common Grid infrastructure for the fundamental service of *remote job execution* and monitoring is provided by the GRAM software from the Globus package. The Condor-G software[2] provides the higher level service of *reliable job submission*. For our purposes, we need to provide the *request brokering* service, referred to as resource brokering elsewhere[3].

Having reviewed the relevant Grid technologies, we have chosen to use Condor's Match Making Service (MMS)[4], as *the* request broker. The novelty of our approach is that the properly configured MMS will be the whole request broker, rather than its part, base, or advisor. Our choice is driven by the collaboration of Dzero with the University of Wisconsin under PPDG, as well by the record of success of MMS as part of the Condor system.

Such a utilization of MMS constitutes a paradigm shift for Condor-G from a personal grid manager to a whole system job manager, and for Condor from a local pool manager to a full-fledged Grid manager (note however that the EDG project is independently using Condor components for global brokering as well). This approach gives us unlimited opportunities to implement various policies, as well as resource management considerations together, by means of redefining the *ranking functions* for the jobs.

To begin the effort we have established several projects to understand the Globus and Condor software. We are enabling the Globus Security Infrastructure in our client server system. GridFTP was introduced as a transport protocol for SAM and tested using the DOE CA for data transfers to and from sites in the UK including Imperial College and Lancaster. A test bed was established to experiment with job submission using the Globus toolkit and Condor-G. Understanding the capabilities of these tools is fundamental to make the appropriate decisions when designing and establishing the architecture for remote job dispatch and SAM. We are actively working with the Condor and Globus teams and providing feedback on problems and compatibility issues we discover.

References

[1] The SAM project, A. Baranovski, D. Bonham, C. Jozwiak, L. Loebel Carpenter, L. Lueking, C. Moore, H. Schellman, I. Terekhov, J. Trumbo, S. Veseli, M. Vranicar, S. White, http://d0db.fnal.gov/sam

[2] J. Frey, T. Tannenbaum, M. Livny, I. Foster, S. Tuecke, "Condor-G: A Computation Management Agent for Multi-institutional Grids", in proceedings of The 10-th IEEE International Symposium on High Performance Distributed Computing, August 2001, San Francisco, CA.

[3] The European Datagrid project, http://www.eu-datagrid.org

[4] R. Raman, M. Livny and M. Solomon, "Matchmaking: Distributed Resource Management for High Throughput Computing", in Proceedings of the Seventh IEEE International Symposium on High Performance Distributed Computing, July 28-31, 1998, Chicago, IL.