The Grid Analysis Environment (GAE): Experiences in developing and deploying a distributed, collaborative environment for data analysis.

Dimitri Bourilkov[♣], Julian Bunn[♦], Rick Cavanaugh[♣], Iosif Legrand[♦], Harvey Newman[♦], Conrad Steenberg[♦], Michael Thomas[♦], Frank van Lingen[♦]

The Grid Analysis Environment (GAE), which is a continuation of the CAIGEE project [13], is an effort to develop, integrate and deploy a system for distributed analysis. The current focus within the GAE is on the CMS experiment [**Error! Reference source not found.**] however the GAE design abstracts from any specific scientific experiment and focuses on scientific analysis in general. The GAE project does not intend to reinvent services, but rather to integrate existing services into a collaborative system of web services.

The first step within the GAE project was to identify key services needed to perform distributed analysis and the interaction between these services. This service design is described in [**Error! Reference source not found.**]. Some of the key services that have been identified (most of which are well known within the Grid community): (1) replica management (2) catalogs (3) workflow management (4) accounting (who has used what resource, and how much of it), (5) authorization (6) service discovery (7) scheduling (8) steering (being able to move your job to another resource).

The second step consisted of mapping existing applications that contain the functionality described in the GAE design to these services (see [**Error! Reference source not found.**]). As the focus of the GAE is the CMS experiment this puts some restrictions on what kind of applications we use within our deployment. For example within CMS, POOL [**Error! Reference source not found.**], and RefDB [**Error! Reference source not found.**] are used as a catalogs. Furthermore, MCRunJob [[4]] and RefDB [**Error! Reference source not found.**] contain functionality that can be mapped to a workflow management service. Integration and exposure of current CMS applications as web services is important within the GAE deployment for CMS. The approach taken by the GAE team, for these applications is bottom up: Analyze the applications and develop (if possible) a "neutral" web service interface for these applications. Furthermore, publish these interfaces in a language neutral format (e.g. WSDL) such that users/applications can access these services even when they are not completely compliant with "standard" interfaces.

For development and deployment the Clarens [**Error! Reference source not found.**] web service framework is used, which is a high-performance wide-area network system for web service deployment that includes powerful features for managing access control to web services, and dynamic service discovery. The choice for Clarens as the GAE backbone does not prevent services within the GAE to access other web service outside the Clarens environment e.g. Globus Toolkit [**Error! Reference source not found.**], as both are based on XML-RPC and the SOAP protocol. The Clarens web service framework is available as a Python and a Java implementation. There are currently 20 known deployments of Clarens: Caltech (5), University of Florida (4), Fermilab (3), CERN (3), Pakistan (2+2), INFN (1).

The third step involved the deployment of robust “basic” services that can be used to develop “higher level” services. Such services are for example: Access control management, service discovery, and file management.

After the basic services, focus shifted on integration of existing CMS software within the GAE. Part of this activity is deploying current GAE services and CMS software on the GAE testbed, to identify what CMS components can be exposed as a web service. Several of these components have been exposed as a web service: POOL, BOSS [**Error! Reference source not found.**], RefDB, MCRunJob, SPHINX [**Error! Reference source not found.**]. Multiple instances of these web services are being deployed on multiple hosts in the current GAE testbed to provide a robust analysis environment (if one host fails instances of a certain service are available on other hosts). Within the second phase, work has also started to integrate monitor information from MonaLisa [**Error! Reference source not found.**] into the Clarens environment. The monitoring information will be used in developing “high level” services.

[♣] University of Florida
[♦] California Institute of Technology

A risk of integration of third party components (in our case CMS related components) is that not every application currently has a stable interface. As a result *from time, to time existing web services need to be updated to reflect the latest versions of the third party software it interfaces to.* The GAE team has identified web service updates as important but also as a risk (as it prevents team members from doing other tasks). *It is therefore important to perform "education and outreach" to developers of third party components the GAE team thinks are necessary within the GAE, and to convince third party developers to provide a web service interface to their applications.*

Equally important to providing transparent access to applications as web services (by using WSDL interfaces and dynamic discovery), is to provide graphical front ends (in our case javascript) to users. It allows users (without installing software) to access catalogs, job submission services and dynamically discover services at the "touch of a button"

Future work of the GAE project will focus on developing "high level" services. High level services can be described as services that take input from the "low level" passive services and proactive take decisions on where to perform an analysis job, when to replicate data, etc… Part of this phase is the development of an accounting service that allows for "fair" sharing of resources. The accounting service, together with other high level services prevent such scenarios in which one users allocate all resources for a very long time (without the organizations consent)

During all phases the GAE team conducts and investigate different types of analysis scenarios based on ROOT [**Error! Reference source not found.**] and ORCA [**Error! Reference source not found.**] that reflect physics analysis within CMS. The GAE team is currently investigating how to perform analysis using the data generated during the Data Challenge 2004 (DC04) [**Error! Reference source not found.**].

In order for the GAE to be successful it is important to have a good synergy with the scientific experiments as to be aware what tools and applications they use, and to know how their analysis processes are organized. This specific domain knowledge (if necessary) can than be integrated into the GAE development and deployment. On the other side, experiments have sometimes adapted specific (in house developed) applications to tackle parts of very "generic" problems, suitable for their application domain, such as workflow management, and (meta) data catalogs. These applications might need to be "translated" and adapted to the more generic syntax, structure and semantics as is used within the Grid environment. Such efforts require not only *technical engineering* but also *social engineering*, enabling GAE developers to offer Grid middleware applications that fit the needs of experiments.

References:

[1] CMS Software Group and Stickland, D. *"CMS Reconstruction Software: The ORCA Project"* CMS note 1999/035

[2] Brun, R., Rademakers, F. "ROOT – An Object Oriented Data Analysis Framework", Proceedings AIHENP'96 Workshop, Lausanne, Sep 1996, Nucl. Inst. & Meth. In Phys. Res. A 389 (1997) 81-86

[3] Duellman, D. "*POOL Project Overview*", Computing for High Energy Physics, La Jolla, California, 2003

[4] Bunn, J., Bouriklov, D., Cavanaugh, R., Legrand, I., Muhammad, A., Newman, H., Singh, S., Steenberg, C., Thomas, M., Van Lingen, F., *"A Grid Analysis Environment Service Architecture"* , http://ultralight.caltech.edu/gaeweb/gae_services.pdf

[5] Bertram, I., Evans, D., Graham, G.E., Love, P., Walker, R. "*McRunjob: A High Energy Physics Workflow Planner for Grid Production Processing*"Proceedings of UK e-Science All Hands Meeting 2003 2-4th September, Nottingham, UK

[6] Lefebure, V., Andreeva, J. "*RefDB: The Reference Database for CMS Monte Carlo Production*" Computing in High Energy Physics, La Jolla, California, 2003.

[7] CMS: http://cmsdoc.cern.ch/cms.html

[8] Steenberg, C., Aslakson, E., Bunn, J., Newman, H., Thomas, M., Van Lingen, F., *"The Clarens Web Service Architecture"*, Computing for High Energy Physics, La Jolla, California, 2003

[9] In., J., Avery, P., Cavanaugh, R., Kulkarni, M., Ranka, S.,"*SPHINX: A Scheduling Middleware For Data Intensive Application on a Grid*," To appear in the proceedings of Computing in High Energy Physics, Interlaken, Switzerland, 2004

[10] Foster, I., Kesselman, C. "*Globus: A Metacomputing Infrastructure Toolkit*" Intl. J. Supercomputer Applications, 11(2):115-128, 1997

[11] Grandi, C., Renzi, A., "*Object Based System for Batch Job Submission and Monitoring (BOSS)*", CMS note 2003/005
[12] DC04: http://www.uscms.org/s&c/dc04/
[13] Newman, H. Branson, J. "*CMS Analysis: an Interactive Grid-Enabled Environment (CAIGEE)*" Submitted to the 2002 NSF Information and Technology Research Program Proposal #6116240