The SAM-Grid / LCG Interoperability Project Report of the trip to Lyon

Gabriele Garzoglio, Parag Mhashilkar

July 21, 2005

Abstract

Goal of the SAM-Grid / LCG interoperability project is making available to DZero the resources provided by the LHC Computing Grid (LCG) through the SAM-Grid system.

In the months between March and June, the collaboration has put in place a "proof-of-principle" test bed, which demonstrated the feasibility of the integration project. The trip to Lyon was an intensive, strict collaboration with the LCG experts, in order to move the test-bed to a "proto-production" stage.

This document reports the achievements of the trip, describes the status of the current test bed, and suggests a path toward production.

Contents

1 Motivations

The SAM-Grid system is an integrated data, job, and information management infrastructure. The system was designed to address the distributed computing needs of the experiments of RunII at Fermilab. In particular for the DZero experiment, SAM-Grid provides a user-friendly environment to run typical production applications, such as montecarlo generation and data reprocessing.

On the other hand, the deployment of the SAM-Grid infrastructure to the collaborating computing centers is an expensive process. People from the SAM-Grid team, from the experiment, and from the site administrators must be involved. The deployment requires the configuration of key computing machines at the site and global site resources, such as the network. It also often requires expert knowledge of the SAM-Grid system to adapt the software to work in unique site resource configurations.

The goal of the SAM-Grid / LCG interoperability project is extending the SAM-Grid system to use the LCG resources deployed by EGEE. The idea is to handle the jobs within the SAM-Grid framework, retaining the characteristics of user friendliness described below. In addition, the deployment of the SAM-Grid can be reduced to a few "forwarding" nodes, which act as interfaces to the LCG system. The actual deployment to the cluster becomes responsibility of EGEE.

We conclude this paragraph analyzing more in detail the features unique to the SAM-Grid, which makes the system user friendly for the RunII experiments. The convenience in the use of the SAM-Grid stems mainly from the following two features

- Transparent access to the data: the SAM-Grid system is fully integrated with SAM, the data handling system of the experiment;
- Integrated Application Management: the SAM-Grid system has knowledge of the typical applications running on the system. The SAM-Grid provides
  - Job Environment Preparation: dynamic software deployment, configuration management, and workflow management (via the integration with the Runjob system);

– Application-sensitive Policies: the SAM-Grid allows the implementation of different policies on data access and local job management. More in detail, different types of applications can access data through different data access queues, each configured with its own policy settings. In addition, different types of applications can be submitted to a local scheduler using different local policies (generally enforced using different job queues)

– Job Aggregation: the job request to the system is automatically split at the level of the local scheduler into multiple parallel instances of the same process. The multiple jobs are aggregated and presented to the user as the single initial request. This allows resource optimizations and user friendliness in the management of the job.

2 The Goals For The Trip to Lyon

The project for the time spent in Lyon was organized around two goals. Both goals have been met and this paragraph describes them in detail.

The integration project is based on the following architecture. Forwarding nodes act as an interface between the SAM-Grid and LCG. To the SAM-Grid, a forwarding node is an execution site, or, in other words, a gateway to computing resources. Jobs submitted to the forwarding node are submitted in turn to LCG, using the LCG user interface. A SAM station offers services to the jobs running on LCG.

Figure 1 shows the test bed configuration before the trip to Lyon. All the components of the "proof-of-principle" test bed were deployed at the University of Wuppertal. Jobs submitted to the SAM-Grid could run on the Wuppertal computing cluster, using the SAM services offered by the Wuppertal SAM station. Submission to the cluster used the LCG user interface and resource broker. Since all the components of the test bed are deployed within the same site, the system is not a good test bed for widely distributed computing, rather it is a "proof-of-principle" that the fundamental components interact properly.

Goal 1 for the trip was reconfiguring the test bed so that the forwarding node could be on a different administrative boundary with respect to the computing cluster and the SAM station. This was the first step in decoupling the deployment of the forwarding nodes from the deployment of the LCG

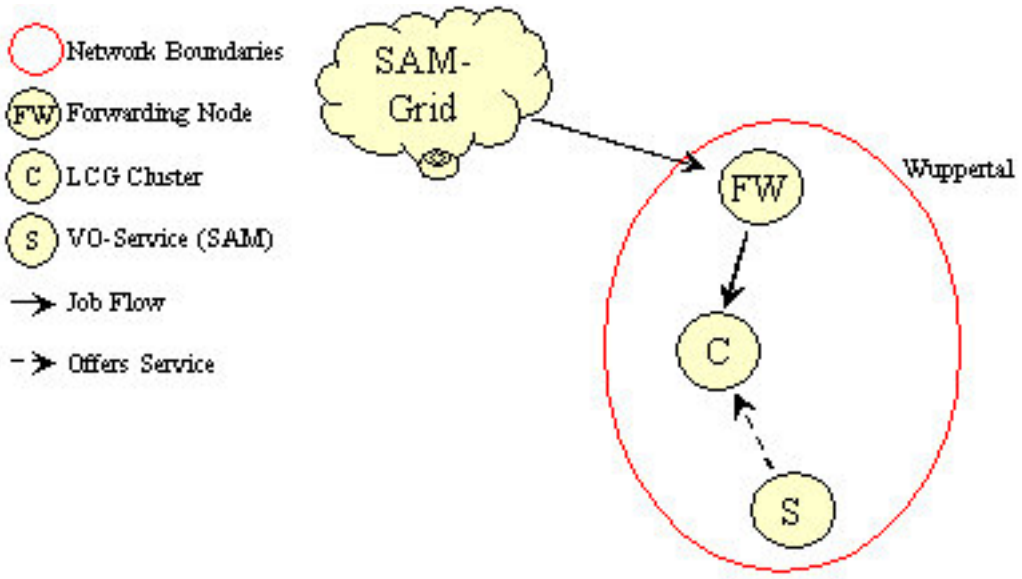

Figure 1: The "proof-of-principle" test bed. The forwarding node, the computing cluster, and the SAM station are all within the same site/network boundaries.

sites. Figure 2 shows a diagram for goal 1. The forwarding node is deployed at Wuppertal, while the SAM station and computing cluster are at CCIN2P3.

Goal 2 for the trip consisted in decoupling the deployment of the SAM station from the deployment of the computing clusters. This step was necessary to allow jobs running on a generic LCG cluster to access the SAM service provided by a "remote" SAM station. Figure 3 shows a diagram for goal 2. In order to meet this goal, the SAM station had to be modified to implement "polling" interfaces in place of the typical SAM "call-back" interfaces. Jobs running at a cluster, in fact, cannot in general receive incoming network connections, thus preventing any callbacks from the SAM station.

3 Lessons Learned

This section describes the typical job workflow used for testing the infrastructures as well as what we have learned during the configuration of the test bed to achieve goal 1 and 2.

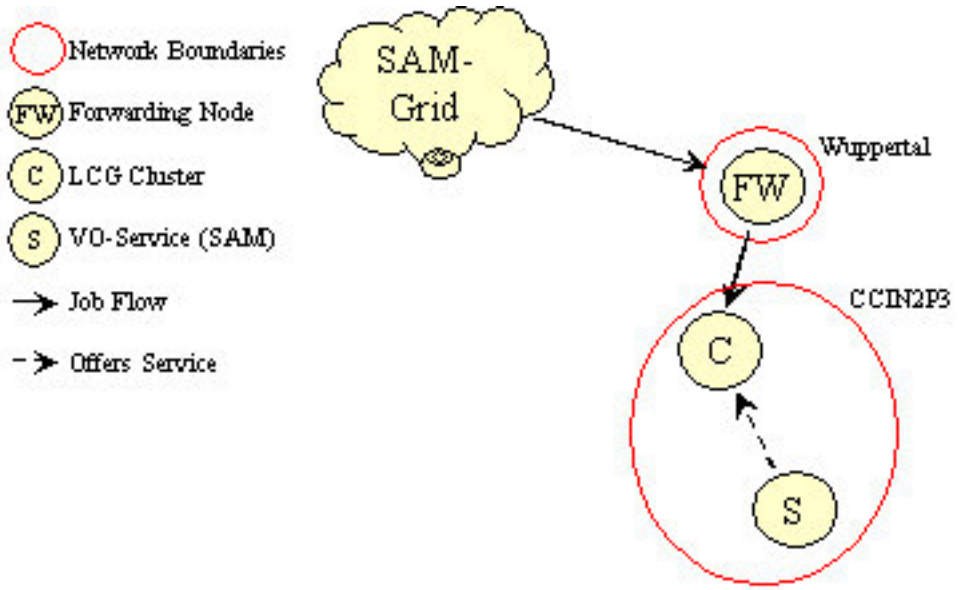

Figure 2: The test bed after goal 1. The forwarding node is on a different administrative boundary with respect to the computing cluster and SAM station.

3.1 Job Workflow

The test bed configuration for each goal has been tested running real DZero reconstruction jobs. Jobs are submitted to the SAM-Grid, they enter the forwarding node and are in turn submitted to the LCG site of choice (depending on what goal was pursued). Jobs running at the computing cluster use the services offered by the SAM station at CCIN2P3, as defined by the forwarding node, in order to download the executable and data files. The output is stored in the durable location set at the forwarding node. Push-based monitoring information are managed by the XML database deployed at the forwarding node.

3.2 Goal 1

Goal 1 consisted in separating the site boundaries of the forwarding node (at Wuppertal) from the boundaries of the computing cluster and of the SAM station (at CCIN2P3).

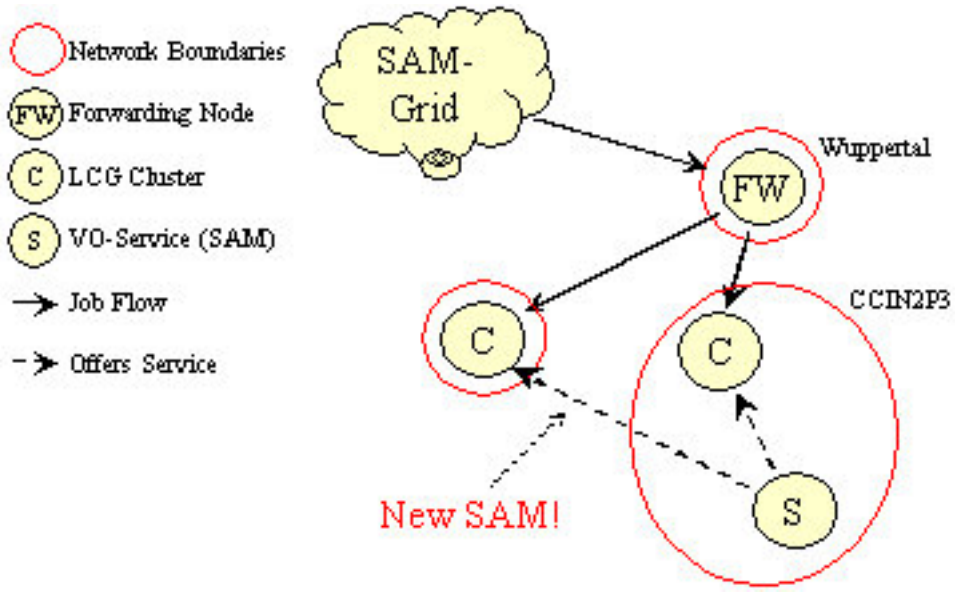

Figure 3: The test bed after goal 2. The forwarding node, the computing cluster, and the SAM station are each on a different administrative boundary.

The SAM station at CCIN2P3 ("ccin2p3-grid1") was configured with a single disk cache. More complex configurations, such as using the HPSS mass storage system as a cache, have not been considered at this point for the difficulty of accessing HPSS from LCG clusters external to CCIN2P3.

The forwarding node was reconfigured to initialize the environment of the jobs with the configuration of the CCIN2P3 station. In detail, the site configuration was changed to refer to ccin2p3-grid1 and a new batch adaptation configuration for ccin2p3-grid1 was created.

In general, sites maintaining a SAM station for the LCG community must allow incoming TCP communication between the forwarding node and the station machine on a restricted set of ports (generally 4500-4505). The forwarding node, in fact, triggers data pre-staging at the station by starting a "sam project", when a job is being forwarded to LCG. The firewall configuration at CCIN2P3 had to be changed accordingly.

The protocol to start a sam project involves a notification of the station (at CCIN2P3) to the client (at Wuppertal) that the project has started successfully or not. In the standard station code, this notification is a UDP-

based call back of the station to the client. Despite the fact that this communication was not blocked, on the wide area network we observed that the UDP packet was consistently lost. The standard station code is not adequate to achieve goal 1. Goal 1 was achieved by installing the station code that implements the new "polling" interfaces. The notification status for starting a project is "polled" by the client from the station using CORBA calls.

The participation of the LCG experts in the project was important to debug the test LCG jobs. In particular, it helped learning the available user interface commands and the interplay between the LCG information system and the computing element configuration.

3.3 Goal 2

Goal 2 consisted in separating the site boundaries of the SAM station (at CCIN2P3) from the boundaries of the computing cluster (at Clermont-Ferrand)

The configuration of the site firewall that hosts the SAM station must allow incoming TCP connections to all the ports of the SAM station machine from all the LCG clusters where jobs can run. This is necessary for the jobs to retrieve files via sam (connection to the "project master" processes). In the future, the SAM station should implement the control of the port range for the "project master" processes, in order to allow more restrictive firewall rules. Once this control is implemented, the firewall configuration will need to allow access to fixed SAM ports (generally 4500-4505 for SAM Station, FSS, Context server), to a range of ports for the SAM project masters (generally a couple of hundreds ports), to the control ports of the gridftp servers (4567-4568) and their data ports (a couple of hundred ports for each of the two servers).

The computing clusters that have been used in the test bed for goal 1 and 2 (CCIN2P3 and Clermont-Ferrand) run jobs on a scratch area that is a local file system. In our experience, SAM-Grid jobs often fail when they run on a scratch area that is a shared file system, because of the high I/O traffic. The I/O traffic comes from the dynamic product installation and from the data movement and processing. LCG does not impose any requirements on the sites to implement scratch management. On the other hand, it is common that sites define the location of the preferred scratch area in the environment of the jobs ($TMP_DIR variable). Some non-standard versions of the LCG job managers use this variable to implement scratch management by wrapping the jobs. Since the non-standard versions of the job managers

are not widely distributed, we plan to implement an LCG-specific scratch management script which will be deployed by the SAM-Grid before running the applications. To our mind, this is a pre-condition for the expansion of the clusters usable by DZero via LCG.

During our initial experience deploying the "proof-of-principle" test bed, we observed that problems at the LCG resource broker caused jobs to fail. During our tests to meet goal 2, we made sure that the forwarding node could use different brokers when submitting jobs through the user interface. Our tests using the Wuppertal and the CCIN2P3 resource brokers were all successful.

Other problems that we have observed during our tests were occasional failures of the LCG infrastructure to report the standard output and error of jobs. Further investigations linked the problem to failures of the "Maradona" server. Jobs that failed were automatically retried by the infrastructure, despite our request in the JDL not to retry failed jobs. We did not have the time to complete the root cause analysis of these failures.

4 Proposed Future Work

With the successful accomplishment of the goals for the trip to Lyon, we have demonstrated that jobs prepared by the SAM-Grid infrastructure can execute at the LCG clusters and can access SAM-Grid services on the wide area network. The current "proto-production" test bed is ready to be expanded with new LCG clusters. This can be done by adding the GRAM URL of the new resources in the *requirement* attribute of the LCG JDL at the forwarding node. The CCIN2P3 firewall will also need to be changed to grant access to the SAM station machine from the new resources.

Figure 4 shows a diagram of the production system. In order to move the proto-production test bed to production, there are still a few tasks that need to be implemented.

- Scratch Management on LCG: DZero applications, especially when prepared by the SAM-Grid, are I/O intensive and cannot typically run on a scratch area that is a network file system. Most LCG sites make available in the environment of the job the location of the "preferred" scratch area. The standard LCG middleware does not use this information and leaves the responsibility of enforcing scratch management

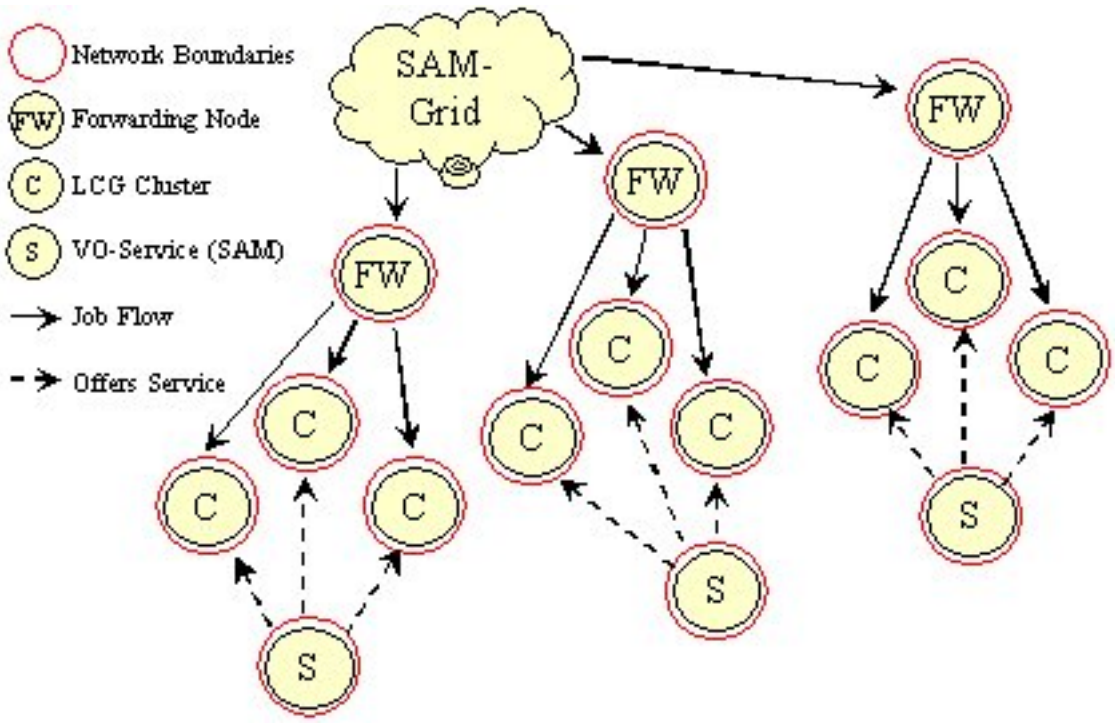

Figure 4: The production system. Station are associated to forwarding nodes. Clusters are disjoint sets associated to forwarding nodes.

to the site or the application. The SAM-Grid team will need to write an LCG-specific scratch manager that tries to run the application on a local file system, using the indicated "preferred" scratch area.

- Output Sandbox Bundling: when all the jobs spawned by a grid job terminate, the SAM-Grid middleware bundles the output of all the jobs and makes it available to the grid for the user to download. The current middleware assumes that the output is *pushed* by the jobs to the gateway machine i.e. to the forwarding node, in the case of the SAM-Grid / LCG integration architecture. Instead, jobs that ran on LCG make available their output via *pulling* interfaces. The SAM-Grid middleware will need to be enhanced to bundle properly the output of the jobs even when the output needs to be pulled.

- Output Buffers and Durable Locations: in the SAM-Grid, the job push

its output to a buffer area where SAM can access and store it. Intermediate results are stored in a "durable" location, permanent results in the enstore mass storage system at Fermilab. The durable location is generally a disk area with some form of space management. Currently buffer and durable areas are at the forwarding node. The space available is about 30 GB. The SAM-Grid infrastructure should be reconfigured to use the 500 GB of disk space available at the station ccin2p3-grid1. A further improvement would be using HPSS as a durable location. The stations accessing this area should always delegate a station at CCIN2P3 for the retrieval, since HPSS is not accessible in general from outside CCIN2P3.

- Control of the SAM Project Master Ports: jobs running at the clusters of LCG must be able to access the project master started at the SAM station. Currently, there is not control over the port range used when starting the project masters. Firewalls must be open from the LCG cluster on all the ports of the SAM station machine. The SAM station should implement a range control for the project master ports, so that firewalls could be opened more selectively.

- Extending the Test Bed to a Production System: after testing extensively the production test bed, we should devise a plan to deploy new forwarding nodes and LCG sam stations. The association between forwarding nodes and SAM station could be done dynamically using the LCG information system.

- Understanding Operations: SAM-Grid clusters undergo a phase of certification before being used for production. What does certifying LCG mean? Should the certification be done on a cluster by cluster basis? Or certifying LCG as a whole is sufficient? Should we certify cluster on the basis of their architecture/operating system? The user community needs to address the operational concerns of this new tool.

5 Acknowledgments

We want to thank Dominique Boutigny, Fairouz Malek, and Amber Boehnlein for making this trip possible; Fabio Hernandez for being an excellent host and for his help with the LCG information system and computing element

configuration; Pierre Girard and the whole LCG group for their day to day help; the networking group for their prompt help in configuring the CCIN2P3 firewall; Tibor Kurca for his continuous support; Andrew Baranovski for the implementation of the "polling" interfaces in SAM.