Grid Reliability

P. Saiz, J. Andreeva, C. Cirstoiu, B. Gaidioz, J. Herrala, E.J. Maguire, G. Maier, R. Rocha

CERN, European Organization for Nuclear Research, 1211 Geneve 23, Switzerland

E-mail: pablo.saiz@cern.ch

Abstract. Thanks to the Grid, users have access to computing resources distributed all over the world. The Grid hides the complexity and the differences of its heterogeneous components. In such a distributed system, it is clearly very important that errors are detected as soon as possible, and that the procedure to solve them is well established.

We focused on two of its main elements: the workload and the data management systems. We developed an application to investigate the efficiency of the different centres. Furthermore, our system can be used to categorize the most common error messages, and control their time evolution.

1. Introduction

Grid infrastructures have been evolving very quickly in recent years. The amount of resources provided by the different centres is also increasing, thus making the Grid more powerful. At the same time, the growing number of services and the complexity of the Grid infrastructure increase the probability of failures, thus making the debugging more difficult.

In order to have such a vast system running effectively, it is vital to monitor information on the status of the different components, and to react very fast to any type of error. The goal of our project is to facilitate the discovery of these errors, and whenever possible to point to recipes that could be used to solve them.

The way we do this is by looking at the information that the Experiment Dashboard [1] collects from different systems, study the different type of failures, and present the results.

At the time of writing this article, our efforts are concentrated in two major components:

- Workload Management system
- Data Management system

For the workload management, we investigate the jobs sent to the LCG Resource Broker [2] by different Virtual Organizations (VOs), and evaluate the efficiency of each site as the ratio between failed and successful jobs. For the data management, we collect statistics about the transfers that ALICE [3] is doing using gLite FTS [4].

The Grid Reliability Dashboard is intended for three different types of clients:

- Site administrators, who will focus on their site and can use the sytem to solve configuration problems.
- VO administrators, since they can check the effective use or resources, and monitor VO specific services.

- End users, who can track their own jobs, and disentangle their application failures from Grid failures.

Moreover, the Grid Reliability can also help middleware developers, since they can see the reliability of the different components, check the most common error messages and follow up their evolution.

The rest of this contribution will describe the system that we have developed to monitor the Grid efficiency. It is currently used by the four LHC experiments and the biomed VO Vlemed [5]. We will also present the results and benefits that the Grid Reliability Dashboard has provided.

2. Job Reliability

Our starting point is the information collected by the Experiment Dashboard [1]. It gathers data from different sources, like R-GMA [6], Imperial College Realtime Monitoring [7] or MonALISA [8]. The Dashboard then presents all this information in a coherent way, as if all of it came from one source.

When a job is submitted to the Grid, it is possible to specify a resubmission policy in case the job fails. It is a useful feature, since the Resource Broker will resubmit the job according to the submission instructions, thus increasing the success rate seen by the user. It is particularly useful in the case when the failures were caused by site problems. From the monitoring point of view, each resubmission has to be studied independently and it is counted as a job attempt. Figure 1 illustrates the difference between a job and a job attempt. In this particular example, there is one job with four job attempts. Three of these attemps failed, but the final one was succesful.

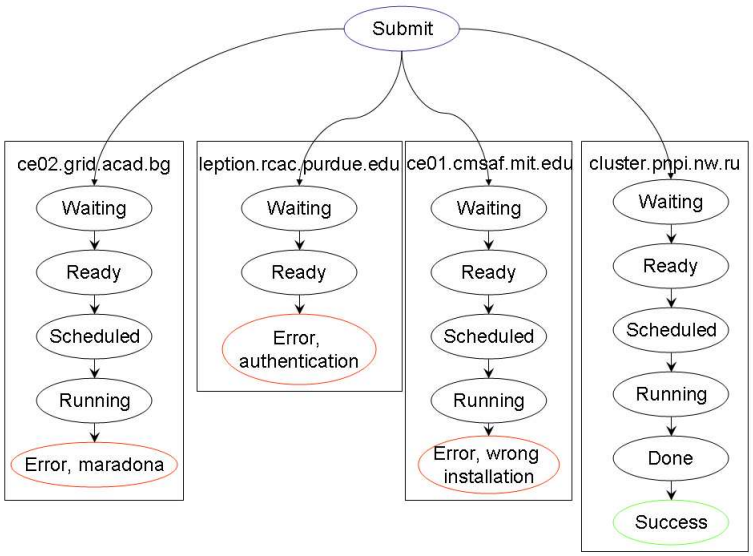

Figure 1. Job vs Job Attempt

If we consider how the different clients would interpret this particular example, we would have: a happy end user because the job was executed properly; three site administrators that have detected a problem in their site; and a VO administrator worried that three sites are not conributing effectively.

Out of these job attempts, we concentrated to single out frequent patterns. Two job attempts are considered to be in the same pattern if they fail due to the same error conditions. Patterns also help in the case where some of the monitoring information is missing or incomplete.

For example, the normal status that a job goes through are SUBMITTED - WAITING - READY - SCHEDULED - RUNNING - CLEARED - DONE. If for a particular job, we do not receive one of the intermediate state transitions, the job attempt will be slightly different, but the pattern would be the same.

At the time of writing this article, the Job Reliability works for jobs submitted through the LCG resource broker.

The different aspects that can be identified from the job reliability are:

- Site Efficiency
  Site efficiency is measured by comparing the number of failed job attempts with the number of successful ones. By looking at the frequency of successful and failed attempts, assumptions can be made about the sites reliability and likelihood of a job being completed.
- Errors and their frequency
  Detecting and understanding the reasons of the job failures is a necessary condition for resolving the problems with the sites or Grid services. This is in particular very useful for middleware developers.
- Waiting Time
  The waiting time is the time it takes from job submission to job completion. This gives us useful information about the Grid performance, taking into account the latency between the submission and the start of the execution. In some cases, this could be a flag for a problem on a site.

These reliability indicators help in the discovery of site problems or inefficiencies. They also help to understand the reason and to quickly resolve the problem, which in turn will improve the overall Grid reliability. The collected data refers to:

- Job data
  This includes user who submitted the job, timestamps, the Computing Element of the last attempt, the Resource Broker, etcetera.
- Job Attempt data
  For each job attempt, we record the Computing Element that executed it, timestamps, and, if available, the worker node where the job was running.
- Attempt data
  The attempts have information about their success, the pattern they belong to, and all the status transitions.
- Pattern data
  Finally, the different patterns contain information about the corresponding errors. More important, the Dashboard database also links the errors to the recipe to cure it, whenever such a recipe exists.

One additional problem is that all this analysis can only be done once the job has finished. Since the number of resubmissions for a given job is not known in advance, it is not possible to understand whether the latest status change message received by the monitoring system for a given job is the final one. Thus, whenever information about a job status change arrives, the system has to recompute all related job patterns to compile the complete workflow for a given job. This is done by the Job Realibility agent, a Dashboard collector. An agent (or collector) is essentially a program which is set to run at various time intervals. There are other agents that automate other tasks, like creating frquently used images or the monthly reports.

Finally, by employing Oracle practices such as partitioning on several tables, it is possible to convert an otherwise large and unwieldy amount of data into a manageable one. For example,

by partitioning the job history table, the speed of queries using this table were improved by an average of 90% which is fundamental for any interactive application.

3. The Web Interface

The web interface for the job reliability has been designed taking into account the needs of the different categories of users. The intention was to make a front-end which would be easy to use and also effective in communicating useful information to all three user groups. Each experiment has its own homepage: ALICE [9], ATLAS [10], LHCb[11] and CMS [12]. The Grid Reliability Dashboard consists of:

- Home Page - This is an interactive Dashboard which shows a snapshot of Site Efficiency and Errors. Users can drag and drop the information they would like to see and arrange the information in a way which is convenient for them (see Figure 2).

Figure 2. An Overview of the Home Page with the various components

- Site Efficiency - A menu to display the site efficiency for a day, a month or view the site efficiency comparison for all the VOs.
- Errors On Site - A query page which allows users to view errors which occurred on a site during a particular month.
- Error List - Information on errors which have occurred on sites. The user can filter the error information by site, month, year or error. The page also includes a breakdown of the errors as a pie chart. Users can also the sites where a specific error occurred by clicking on the Site Details link. It is also possible to view the progression of an error over the current year, and the proportion between this error and all errors on a given site (see Figure 3).
- Waiting Time - This page displays a query on which users can filter the information to display on waiting times. Users can filter by site, minimum and maximum execution time and date, and they will receive plots showing the Waiting Time details.

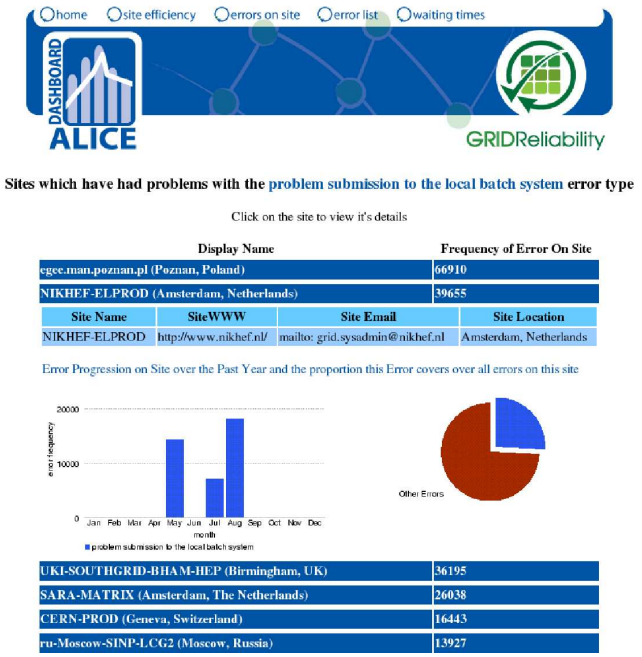

Figure 3. Plot View for each site showing the error evolution over the current year and the fraction of each error contributing to the overall number of errors at a given site.

The pages with dynamic content, such as drag and drop features, use JavaScript and CSS which is portable across all browsers. All the images inside the home page are generated every few hours by an agent, and all the information (top sites, bottom sites, power sites, and efficiency) is stored in files for quick retrieval, so no database access is required on this page. Fast access to this page is important since it is the entry point for browsing the Grid Reliability monitoring data and therefore will potentially have the highest visitor rate. Moreover, caching at proxies will speed up data loading significantly if the same page is accessed by multiple users.

Since the Grid Reliability is being developed within the Dashboard framework, it benefits from the functionality that such a framework provides. For instance, all data from the previous web pages can be retrieved in different formats, like CSV or XML. This allows an easy integration of this data with any command line tools and data retrieval by other applications.

4. Data management reliability

Another important activity of the LHC VOs on the Grid is data transfer. All LHC experiments rely on the data management tools provided by the Grid middleware, in particular FTS.

We started with the study of transfers in ALICE. Once the experiment starts running, all the data collected by the detector will be transfered to the CERN compuing centre and replicated to other sites. This is done via AliEn [13] and its File Transfer Daemon (FTD), which uses FTS. ALICE perform regular data challenges, where they exercise the whole system with simulated data.

We collected statistics from the FTD during several of these challenges. We considered every separate link (which is defined by its source and destination) and generated efficiency reports for each link. The reports also included the number of times that different errors occured on the different links.

Since the FTD is used only by ALICE, we are now switching to do the analysis at the FTS

level, so that the application can be used for other LHC VOs without any change.

5. Automatic report generation

Another important functionality is the creation of the Grid usage reports for different VOs. Since the four LHC experiments are using the Dashboard, we can compare the data that we get from them.

Each month, a process creates automatically pdf files with the site efficiency reports for the four LHC VOs. An example of such a report can be seen if figure Figure 4.

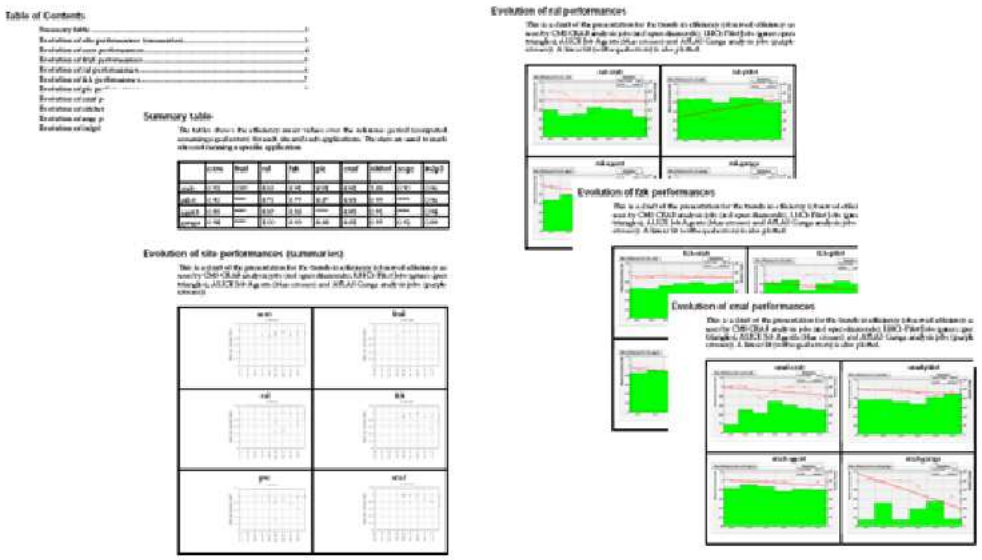

Figure 4. Automatic reports created by the Job Reliability

6. Future plans

For the workload management reliability, our next goal is to incorporate more sources of information. Currently we are working on getting information from gLite Logging and Bookkeeping system [14] using the subscription mechanism. Since ATLAS and CMS are widely using job submission via condor_g[15], we are also working on enabling job status information from condor_g submitters.

For the data management, we want to make the system more generic, so that other VOs can also profit from it.

At the moment, the LHC VOs agreed that all the information should be available for everyone. For other VOs, the data should not be public, and only people with the right authorisation should be able to see it. For that reason, we are also working on setting X509-based authentication on the web pages.

7. Conclusion

We have created a monitoring system to investigate the reliability of different Grid components, concentrating on the workload and the data management systems. The Grid Reliability application analyses the data collected by the Experiment Dashboard and generates site

efficiency reports covering job processing and data transfers activities. The results of this analysis are available to the Dashboard web sites of the LHC experiments. In addition, the same data can be retrieved in different formats, thus simplifying the usage of this data from command line tools.

Acknowledgments
This work was funded by EGEE. EGEE is a project funded by the European Union under contract INFSO-RI-031688.

References

[1] Andreeva, J. et al., Dashboard for the LHC experiments, CHEP07, paper 306
[2] LCG Resource Broker, http://lcg.web.cern.ch/LCG/
[3] The ALICE collaboration, http://aliceinfo.cern.ch
[4] FTS - The gLite File Transfer Service, http://egee-jra1-dm.web.cern.ch/egee%2Djra1%2Ddm/FTS/
[5] VleMed, http://internal.vl-e.nl/
[6] R-GMA - Relational Grid Monitoring Architecture, http://www.r-gma.org/
[7] Imperial College Rel time monitoring, http://gridportal.hep.ph.ic.ac.uk/rtm/
[8] MonALISA: An Agent based, Dynamic Service System to Monitor, Control and Optimize Grid based Applications, I.C.Legrand, et al., CHEP 2004, Interlaken, Switzerland
[9] The ALICE Grid reliability homepage, http://dashb-alice.cern.ch/jr.html
[10] The ATLAS Grid reliability homepage, http://dashb-atlas-job.cern.ch/dashboard/request.py/Home
[11] The LHCb Grid reliability homepage, http://dashb-lhcb.cern.ch/dashboard/request.py/Home
[12] The CMS Grid reliability homepage, http://dboard-gr.cern.ch/data/
[13] Saiz, P. et al., AliEn - ALICE environment on the Grid, NIM., A502 (2003) 437-440.
[14] gLite Logging and Bookkeeping homepage, http://egee.cesnet.cz/en/JRA1/index.html
[15] The Condor project, http://www.cs.wisc.edu/condor/