Monitoring System for the GRID Monte Carlo Mass Production in the H1 Experiment at DESY

Elena Bystritskaya\textsuperscript{1}, Alexander Fomenko\textsuperscript{2}, Nelly Gogitidze\textsuperscript{2}, Bogdan Lobodzinski\textsuperscript{3}

\textsuperscript{1} Institute for Theoretical and Experimental Physics, ITEP, Moscow, Russia
\textsuperscript{2} P.N.Lebedev Physical Institute, Russian Academy of Sciences, LPI, Moscow, Russia
\textsuperscript{3} Deutsches Elektronen-Synchrotron, DESY, Hamburg, Germany

E-mail: bogdan.lobodzinski@desy.de

Abstract. The H1 Virtual Organization (VO), as one of the small VOs, employs most components of the EMI or gLite Middleware. In this framework, a monitoring system is designed for the H1 Experiment to identify and recognize within the GRID the best suitable resources for execution of CPU-time consuming Monte Carlo (MC) simulation tasks (jobs). Monitored resources are Computer Elements (CEs), Storage Elements (SEs), WMS-servers (WMSs), CernVM File System (CVMFS) available to the VO HONE and local GRID User Interfaces (UIs).

The general principle of monitoring GRID elements is based on the execution of short test jobs on different CE queues using submission through various WMSs and directly to the CREAM-CEs as well. Real H1 MC Production jobs with a small number of events are used to perform the tests. Test jobs are periodically submitted into GRID queues, the status of these jobs is checked, output files of completed jobs are retrieved, the result of each job is analyzed and the waiting time and run time are derived. Using this information, the status of the GRID elements is estimated and the most suitable ones are included in the automatically generated configuration files for use in the H1 MC production.

The monitoring system allows for identification of problems in the GRID sites and promptly reacts on it (for example by sending GGUS (Global Grid User Support) trouble tickets). The system can easily be adapted to identify the optimal resources for tasks other than MC production, simply by changing to the relevant test jobs. The monitoring system is written mostly in Python and Perl with insertion of a few shell scripts.

In addition to the test monitoring system we use information from real production jobs to monitor the availability and quality of the GRID resources. The monitoring tools register the number of job resubmissions, the percentage of failed and finished jobs relative to all jobs on the CEs and determine the average values of waiting and running time for the involved GRID queues. CEs which do not meet the set criteria can be removed from the production chain by including them in an exception table. All of these monitoring actions lead to a more reliable and faster execution of MC requests.

1. Introduction
System monitoring is essential for any production framework used for large scale simulation of physics processes. The computing efficiency of a large scale production system critically depends on how fast misbehaving sites can be found, and how fast all affected jobs can be resubmitted to the new sites. In the highly distributed GRID environment this aspect is very crucial and
affects the overall success of the mass computing scheme in High Energy Physics in a significant manner.

In this article we describe a monitoring tool, which is an integral part of the MC production scheme implemented by the H1 Collaboration for the WLCG (Worldwide LHC Computing GRID) resources. Details of the large scale H1 GRID Production Tool are presented in [1] and [2].

Monitoring the status of accessible GRID sites is one of the most important ingredients of the H1 MC Framework. The monitoring part, named the Monitoring Test System (MTS), allows a detailed check of GRID site status including local and WLCG resources. This check is performed off-line using periodically started test-jobs and can be dynamically migrated to quasi real time mode during the MC simulation.

The system is used for monitoring of GRID sites with gLite [3] and EMI [4] instances, i.e. the GRID resources available for direct job submission and so called ARC-type computing elements (Advanced Resource Connector Computing Element [8]).

2. Overview of the Monitoring Tool

The H1 monitoring tool was initially based on the official Service Availability Monitoring (SAM). It soon became clear, that the SAM system was rather unstable and insufficiently equipped for the particular checking mechanism required by the H1 monitoring. The successor of SAM, based on Nagios [7], also does not fulfill all requirements posed by the H1 monitoring background. The H1 MC simulation on the GRID is a chain of sequential processes controlled by the MC manager, described in [1] (see Figure 4 therein). In addition to the default set of tests offered by the SAM we needed to check the correctness of the full MC chain, including the access of remote processes to the MC database snapshot, served by the internal H1 servers. Therefore, the H1 monitoring service tool MTS was developed in the framework of the H1 MC production infrastructure.

The H1 MTS can be divided into several areas with different scope and functionality. They are:

(i) Site monitoring - describes the status of the GRID sites available for the HONE VO, their computing infrastructure in terms of available resources, their “health” according to the efficiency of execution of defined test tasks.

(ii) Overview of distributed resources - collects and visualizes test results for the whole computing infrastructure.

(iii) History - collects information about distributed resources in terms of GRID services. Allows viewing of the changes over time of the distributed computing resources.

Each of these areas is described in more detail below.

2.1. Site monitoring

The general scheme of the site monitoring is based on:

(i) Regular submission of short test jobs to each computing element of the European data GRID (EDG) computing environment available for the HONE VO.

(ii) Analysis of results of the production status of MC simulations.

As test jobs we use shortened versions of real production jobs, where the simulation task is limited to a few minutes of CPU time.

To support any requested type of tests the existing GRID environment (gLite,EMI) is used. The idea is to use the GRID user interface (UI) environment for the core infrastructure of the MTS system, i.e. the same environment which is used for the H1 simulation of MC events. This allows simplification of the monitoring system, as well as checking of every part of the GRID
The site monitoring includes tests of the information system instance (BDII [5]), tests of the CE, Cream-CE (Computing Resource Execution And Management Computing Element, [6]), ARC-CE, SE, WMS (Workload Management System) service, LFC (LHC File Catalog) and SRM (Storage Resource Manager). It also checks for available scratch space on the Working Node (WN), for availability of the $VO_HONE\_SW\_DIR$ directory and CVMFS space [9].

To locate computing, storage and WMS services available for the HONE VO, the result published by the BDII resource is used. All results of these service tests are written into the MySQL DB. In order to provide a faster access to current DB tables, the system keeps data from the last 48 trials. The data from previous tests are moved to a separate DB table (archive storage).

GRID site usability in terms of the computing elements is based on the result of test measurements of the GRID status of the job (Submitted, Waiting, Ready, Scheduled, Running, Done, Cleared, Aborted and Cancelled) and the status defined by the wrapper of the MC simulation jobs, named h1mc state. Relations between the GRID and the h1mc states, used by the MTS tool, are collected in table 1.

Table 1. Relations between the GRID and the h1mc states of the MC jobs used for production work and for test of the GRID site status. The column MARK is used for the visualization of the final result of the test job.

| GRID STATE | H1MC STATE | MARK |
|------------|------------|------|
| Aborted, Cancelled, Done (Exit Code !=0), Unknown, No output | - | ERROR |
| Not submitted | - | CRIT - fatal errors |
| Ready, Scheduled, Waiting, Running | - | NA - not available |
| Done (Success) | no errors | OK - normal status |
| Done (Success) | failed components are restarted and successfully finished | INFO - information |
| Done (Success) | more than 1 restarts of failed components | NOTE - important information |
| Done (Success) | transfer of input/output failed; after restart - successful | WARN - warning |
| Done (Success) | missing system libraries | CRIT - fatal errors |
| Done (Success) | No output | ERROR |
In addition to the error codes, the MTS system measures the time of the job execution and the scheduled (waiting) time. The execution time of the test job is determined by the download and upload time of input and output files, and therefore gives information about the access of the job to an SE. Using both times the MTS system determines the priority of a given GRID site.

To check storage elements a standard set of commands available throughout the gLite and EMI UI environment is used. The MTS system executes the check command and measures the time of the execution. The set of commands is the following:

- `lcg-cr` - upload and registration of an input file in a given SE
- `lcg-cp` - download of a file from the SE using `srm` and `lfc` protocols
- `lcg-del` - delete a file from the SE

In order to check the status of the WMS services the MTS system examines the execution of the matching, submission and cancelling commands and performs a measurement of the execution time. In the case of the SEs and WMS services the MTS uses only two MARKs: `OK` - if all tests are successful, and `ERROR` - if any test has failed. Only errorless services are used in the MC production.

The monitoring system also uses results from real time production cycles by collecting information about the status of used GRID sites. This is done by parsing requested data from log files created by all production jobs. Use of the production jobs by the monitoring system allows a quasi real time monitoring of the status of available GRID sites. The tool registers the number of job resubmissions, and calculates in real time the percentage of failed and finished jobs relative to the number of all jobs executed on a given computing service. All gathered results are used for the automatic creation and update of proper configuration files used for the production cycles of the H1 MC framework.

The MTS system creates the following configuration files

(i) The **exceptions** file - a list of available CEs rejected from MC production
(ii) The **good** file - a default list of good available CEs
(iii) The **best** file - a list of the best available CEs

The **exceptions** configuration file contains those CEs, where the percentage of the failed jobs is larger than the default value defined globally for the system. As a default value 50% is used and can be changed by the operator. This list becomes the subject of further investigation, usually equivalent with the creation of a GGUS Trouble Ticket [10]. We also have flexible control on the **exceptions** file through the possibility of external modification of the contents of this file: we can add or remove a specified CE(s) from the **exceptions** list.

The **good** configuration file lists good available CEs, i.e. the CEs which fulfill the following criteria

- The CE is not present in the **exceptions** configuration file.
- The **GRID state** of the CE must be *Done (Success)* and the **MARK** of the CE must be **OK** or **INFO**.

The **best** configuration file collects a list of the best CEs which meet two additional requirements:

- The computing element priority is **smaller than** the default value defined globally for the system (default value is 5, with smaller values denoting higher priority).
- The total time of a test job is **smaller than** the default value defined globally for the system (default value is 60 min).
Update of the good and the best configuration files is done by the monitoring system during every checking action. In a production cycle we usually use the full list of available CEs from the good configuration file. In case of urgent requests or for faster request completion we can switch to use CEs from the best configuration file only.

2.2. Overview of distributed resources
Inside this area of functionality of the monitoring system we have the possibility to view the global status of available GRID computing, storage and WMS services. This permits the tuning of a default set of global parameters used by the MTS tool for the separation between best, good and bad services. Most available GRID sites to which jobs are submitted by the HONE VO share computing resources with the LHC VO. On these sites H1 jobs have lower priority. Therefore, this tool functionality is especially useful when a given site is overloaded by activity of the LHC collaborations. In response, we need to decrease our requirements.

2.3. History
The history aspect of the monitoring tool is often very useful and sometimes absolutely essential. The possibility of viewing history records from different sites allows a more precise determination of the H1 long term usage of the GRID infrastructure.

All information gathered from the output of the test jobs is collected in the DB and can be visualized on a website. An example of such visualization, including results of the tests, is presented in Figure 2, where histories of CEs and job status statistics are presented. Similar plots are available for all kinds of service published by the BDII.

![Figure 2](image)

**Figure 2.** Example of graphical visualization of the history presentations for CEs (left panel). The history is present for the last 48 trials. The two rows visible for each element show a collection of monitoring results for the GRID state (upper row) and the h1mc state (lower row). The right panel shows the job status statistics with the percentage of failed, running and finished jobs for each used CE, as well as the location of the CE in the configuration files. The panel also allows the selection of a given CE by hand and addition/removal it to/from the exceptions configuration file.

3. Conclusions
Since 2006 the H1 Collaboration uses the GRID computing infrastructure for MC simulation purposes. Due to the unsatisfactory stability level, and due to the missing functionalities of officially available monitoring tools, the H1 Collaboration developed the Monitoring Test System MTS as an integral part of the H1 MC production framework. The MTS tool is based on the existing GRID UI environment. The system can easily be expanded by additional developed...
extensions. It may include new monitoring probes and new parameters for better selection of the most effective GRID sites for large scale MC simulation. During the last three years the use of MTS has resulted in a more reliable and faster execution of MC requests. Currently the H1 MC simulation production can be realized at the level of $800 \cdot 10^6$ MC events per month.

References

[1] B. Lobodzinski et al. 2010 *J. Phys.: Conf. Series* **219** 072040
[2] E. Bystritskaya et al. 2012 *J. Phys.: Conf. Series* **396** 032067
[3] Paolo Andreetto at al. 2008 *J. Phys.: Conf. Series* **119** 062007
[4] European Middleware Initiative (EMI) [http://www.eu-emi.eu](http://www.eu-emi.eu)
[5] Laurence Field and Markus W. Schulz. Grid deployment experiences: The path to a production quality LDAP based grid information system. In Proceedings of 2004 Conference for Computing in High- Energy and Nuclear Physics, Interlaken, Switzerland, 2004.
[6] Paolo Andreetto et al. 2012 *J. Phys.: Conf. Series* **396** 032003
[7] [https://tomtools.cern.ch/confluence/display/SAMDOC/SAM+Overview](https://tomtools.cern.ch/confluence/display/SAMDOC/SAM+Overview)
[8] M. Ellert, M. Gronager, A. Konstantinov et al. 2007 *Future Gener. Comput. Syst.* **23** 219-240 ISSN 0167-739X
[9] [https://www.gridpp.ac.uk/wiki/RALnonLHCCVMFS](https://www.gridpp.ac.uk/wiki/RALnonLHCCVMFS)
[10] [https://ggus.eu/pages/home.php](https://ggus.eu/pages/home.php)