A LIGHTWEIGHT MONITORING AND ACCOUNTING SYSTEM FOR LHCB DC'04 PRODUCTION

M.Sánchez, USC, Santiago de Compostela, Spain
V.Garonne, CPPM, Marseille, France
R.Graciani Díaz, R.Vizcaya Carrilo, UB, Barcelona, Spain.
J.J. Saborido Silva, USC, Santiago de Compostela, Spain

Abstract

The phase 1 of the LHCb Data Challenge 04[1] includes the simulation of 200 million simulated events using distributed computing resources on 63 sites and spanning over 4 months. This was achieved using the DIRAC [2] distributed computing Grid infrastructure. Job Monitoring and Accounting services have been developed to track the status of the production and to evaluate the results at the end of the Data Challenge.

The end user connects with a web browser to Web-Server applications showing dynamic reports for a whole set of possible queries. These applications in turn interrogate the Job Monitoring Service and Accounting Database by means of dedicated XML-RPC interfaces, querying for the information requested by the user. The reports provide a uniform view of the usage of the computing resources available. All the system components are implemented as a set of cooperating python classes following the design choice of LHCb. The different services are distributed over a number of independent machines. This allows several thousand concurrent jobs monitored by the system.

INTRODUCTION

Goals

The main goal is to provide the dedicated production grid of LHCb with Monitoring and Accounting subsystems.

What is understood by monitoring is a service capable of reporting the current status of jobs in the Workload Management System[3] (WMS) of DIRAC at any given moment. Here the status of a job includes both parameters reported by the WMS and parameters filled by the job itself. Examples of the former are the site where the job is being executed while an example of the latter is the name of the application being run by the job. This information is used by the overall production manager of LHCb to monitor the performance and status of the participating sites, but also by site managers monitoring the health of their site or trying to find out why a particular job has failed. The goal for 2004 is for the system to be able to scale up to a few thousand concurrent jobs.

Accounting, on the other hand, is concerned with the accumulation of statistics on relevant parameters for the jobs that go through DIRAC. These statistics are to be used, for example, in order to compute the contribution of each participating site to the total amount of resources available to LHCb over a certain period of time, or to assess the performance of the different components of the Workload Management System. The scalability goal for 2004 is for the system to be able to cope with a few hundred thousand jobs.

Design choices

In the design of the Monitoring and Accounting services we have strived towards simplicity, going for more complex structures only when real world experience suggests so.

Job information is provided to the monitoring mainly by the WMS and the job itself, since they are the most knowledgeable about each job. However, other agents may provide additional information. The information is then stored centrally for all jobs. Clients may connect to the central service and obtain a complete view of the system.

This model is simpler than having producers of data and consumers registered against them to conform a hierarchical flow of information. Robustness can be achieved by introducing redundancy at the level of the central services. Regarding scalability, during the Data Challenge it has been shown to scale to 3500 simultaneous jobs, which could be improved with optimisation on the monitoring clients and hardware solutions at the server side.

From the client's point of view the Monitoring service is passive. There is no way to subscribe to the service and automatically get updates when parameters change.

Different infrastructure is used for monitoring and accounting, so jobs can never be on both systems at the same time. Monitoring is dynamic in the sense that a job can change its state, while accounting is static. Having different services for accounting allows things like accumulating summary information for each job as soon as the accounting server receives it.

MONITORING

Web interface

A web interface[4] is available for users to query the Monitoring service. The main entry page to this interface is shown in the figure below

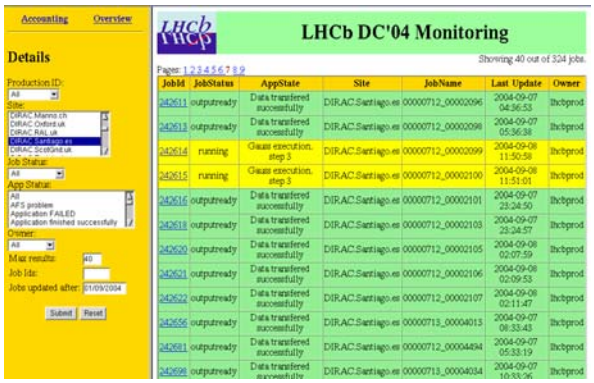

Figure 1:Monitoring web interface

The web page is divided in two frames: a control frame on the left and a results frame on the right. On the control frame there are links, at the top, for quick access to the accounting and several predefined plots showing an overview of the system.

Still on the control frame there are several widgets allowing the user to specify which jobs she wants to look at. In particular the event type, execution site or sites, job status, application status and job owner can be selected. The options available in the different widgets are determined through queries to the monitoring system, in that sense they are dynamic. Once satisfied with a selection the submit button can be pressed to display the selected jobs in the results frame on the right.

Selected jobs are shown as rows of a table, where the columns correspond to the most commonly requested job parameters. Each row is coloured according to the job status.

For detailed information on a job the user can click on the job id and another window returns all known parameters for the job. One typical use case is a site manager trying to find out why a job failed. For that purpose she may click on the job id, bringing out the details window. There she can find an error message as well as the worker node where the job ran and the job id in the local batch system.

Back in the control frame there is a link at the top called "overview". Clicking on this link brings out a set of predefined plots on the results frame. The plots provide an overview of how the WMS is performing, showing the number and distribution of running jobs (see Figure 2) as well as the status of jobs grouped by site or event type.

Implementation

The Monitoring service is implemented as a public XML-RPC server exporting an interface to query the WMS for whatever job parameters are declared there. On the WMS there is a distinction between primary and secondary parameters. The former are a fixed set that is defined centrally. The latter are parameters defined by the job itself.

The fact that primary parameters are fixed allows for the storage backend to be optimised in order to allow fast queries on them. On the other hand, secondary parameters are not predefined; so they are stored as key-value pairs, the access is slower and there is no possibility to put conditions on them when querying the monitoring system.

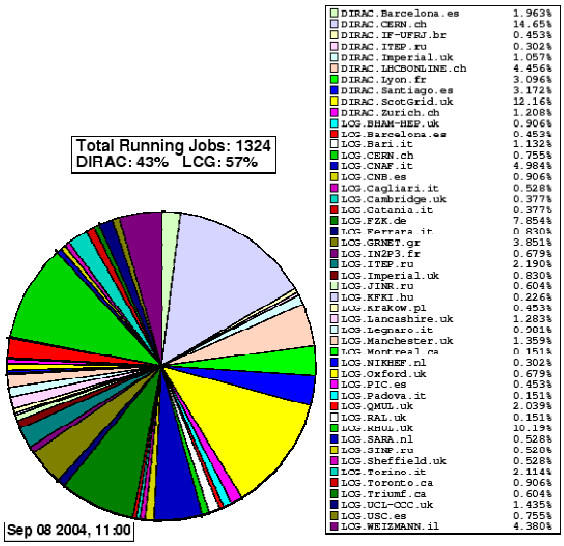

Figure 2: Running jobs per site

Job Monitoring Server API

The API to access the Monitoring Server is quite straightforward. There is a method getJobs(constraints) which returns a list of ids for jobs in the WMS verifying the constraints expressed in the argument. The constraints are a dictionary of key-value pairs. For example to retrieve the list of jobs running at CERN we would write, in python: server.getJobs({'Status': 'running', 'Site': 'DIRAC.CERN.ch'}) .

Given a job id there are methods to retrieve the different parameters associated to the job like getJobParameter(jobid, parameterName) or getJobOwner(jobid)

Lastly there is support for bulk operations where a list of job ids is given as argument and the requested parameter is retrieved for each job in the list.

ACCOUNTING

Web interface

The accounting web interface[5] guides users in generating reports from the information contained in the accounting database. Besides that, it also offers a set of predefined reports, which are commonly requested.

The main entry page is shown in Figure 3. The uppermost half of the page is devoted to driving the user through a set of menus allowing the generation of custom reports, while the lowermost part provides direct access to a few predefined, pre-generated reports.

Each report is presented to the user in three ways: graphically, as a table and as a sheet in EXCEL format. The different reports available are divided in three categories: used resources, produced data and WMS statistics.

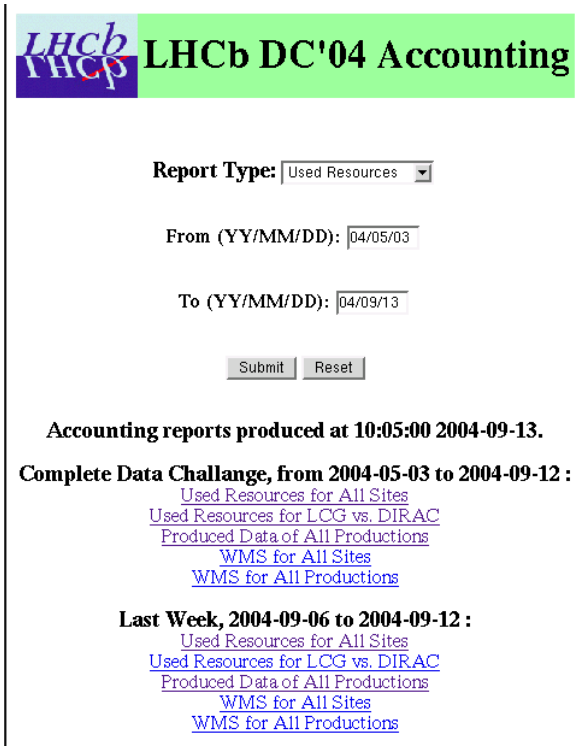

Figure 3: Accounting Home page

Used resources can be shown by production site or by event type. The reports by site cover variables like cpu consumed, number of events produced and storage consumed. The reports by event type include cpu consumed per job, storage needed per job, execution vs. cpu time and output data vs. execution time.

Regarding output data there are rate and cumulative plots for the number of events produced and the amount of data generated (see Figure 4).

Finally, the plots about WMS statistics allow the assessment of its performance.

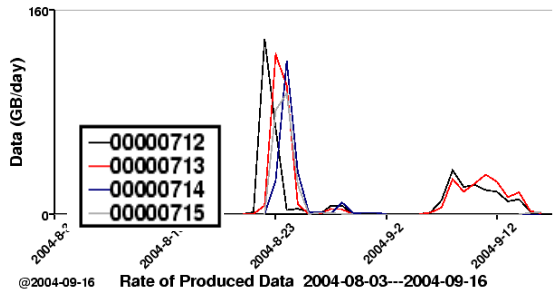

Figure 4: Rate of output data for different event types

Internals

The accounting information is stored in a database separated from the main job database used by the WMS so that both systems do not interfere. This organization allows for the layout of the accounting database to be optimised for efficient generation of the accounting reports. Furthermore, the accounting is domain specific, meaning it understands the peculiarities of LHCb production jobs, while the WMS is generic.

This database can be accessed using two different XML-RPC servers: read-only and write-only. Both servers have restricted access. The read only one can only be accessed by the accounting web interface. The write only server is restricted in order to have control on the origin of the data written to the accounting. In particular, a cleaner agent running centrally moves jobs to the accounting. This agent is also responsible of deleting them from the database used by WMS and Monitoring.

Usage

During the Data Challenge 2004, the accounting was queried an average of 10 times per day. To give an idea of the performance, the time needed to generate all the static reports daily is approximately 8 minutes. 60-70% of the time is spent in over 600 queries to the accounting database while the rest is spent in the drawing package.

Figure 5 shows the number of jobs accounted for as a function of the number of weeks since the start of the data challenge. On average 10000 jobs were entered per week while the average load in the server was below 0.2.

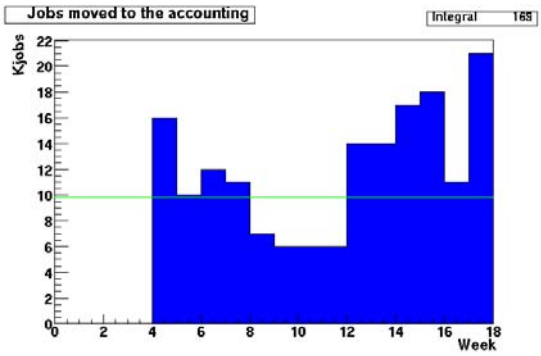

Figure 5: Jobs written to the accounting

CONCLUSIONS

The LHCb Lightweight Monitoring and Accounting services were able to handle the load of the LHCb Data Challenge 2004 with up to 4000 concurrent jobs and up to 170000 jobs accounted for.

The experience of the Data Challenge has made it possible to spot places to improve, such as optimisation in the Monitoring clients and the need for journaling at the level of Monitoring updates.

REFERENCES

[1] J. Closier. Results of the LHCb Data Challenge 2004. This proceedings #403

[2] A. Tsaragodtsev. DIRAC – The Distributed MC Production and Analysis for LHCb. This proceedings #377.

[3] DIRAC Workload Management System. This proceedings #365.

[4] http://fpegaes1.usc.es/dmon/DC04/joblist.html

[5] http://lhcb.ecm.ub.es/DC04/Accounting/