FILE-METADATA MANAGEMENT SYSTEM FOR THE LHCB EXPERIMENT

C. Cioffi, Department of Physics, University of Oxford, United Kingdom
J. Closier, M. Frank, F. Loverre, S. Ponce, CERN, Geneva, Switzerland

Abstract

The LHCb experiment [1] needs to store all of the information about datasets and their processing history from particle collisions at the Large Hadron Collider at CERN [2], as well as for simulated data.

To achieve this functionality, a database design based on data warehousing techniques has been chosen, where several user-services can be implemented and optimized individually without losing functionality or performance. This approach results in an experiment-independent and flexible system. It allows fast access to the catalogue of available data, to detailed history information and to the catalogue of data replicas. Queries can be made based on these three sets of information. A flexible underlying database schema allows the implementation and evolution of these services without the need to change the basic database schema. The consequent implementation of interfaces based on XML-RPC allows access and modification of the stored information using a well-defined encapsulating API.

In this document, we discuss the definition of metadata describing datasets and how these data are used by LHCb physicists to retrieve and access data from simulated particle collisions

INTRODUCTION

LHCb is a high-energy physics experiment at the future CERN accelerator LHC that will start data taking in 2007 [1]. LHCb is expected to produce roughly 3 PB of data per annum resulting from particle collisions and simulations. This large amount of data requires innovative solutions to ease the physicists' access to these data according to physics criteria. A system that stores information about datasets and provides tools to browse this information is essential.

METADATA DEFINITION

For the LHCb experiment, metadata are data characterising datasets, such as the type of event contained in a file, the size of a file and the file location.

In LHCb three different kinds of metadata are identified:

(a) Job Provenance data, which contains all the information of the processing history of a given dataset.
(b) Bookkeeping information, a subset of the job provenance data. Physicists base the selection of the datasets they want to use for their data analysis on this information.
(c) File catalogue information, describing the physical copies of a given dataset.

The metadata are essential to simplify the process of accessing a file. In 2007, LHC will start to produce real events and the analysis of them will commence, resulting in millions of files. The Metadata provides a way of identifying the location of the files of interest avoiding unnecessary access to data storage.

The File-Metadata Management System provides services for handling the Metadata domain.

BOOKKEEPING DESIGN

The design of the File-Metadata Management System, commonly known as Bookkeeping within LHCb, is explained in the following 5 subsections;

- The two-schema strategy
- Warehousing DB schema
- Populating the Warehousing Database
- Generation of specialised Views.
- Web Service to access the metadata information

The two-schema strategy

When LHCb started to work on the Bookkeeping the requirement was for a system capable of delivering high performance and at the same time to be very flexible. Having a flexible system would decrease the need for revision in the future. To develop a database schema that satisfies these two orthogonal requirements is difficult. To this end it was decided to split these two aspects in to different schemas.

One schema has been design (fig 1) to allow the storing of data and this schema is associated to a database that we call a Warehousing DataBase (WDB). This schema does not present a complex structure and can be adopted to store any information describing all data processing activities. This is possible because key-value pairs are used to describe parameters.

Examples of how flexible this schema is can be taken from the 2004 data challenge. Moving towards grid technology, LHCb needed to associate to its files a Grid Unique Identifier (GUID) that is additional information to be stored in the WDB. The only action taken was to define a name for this information and associate a new pair (name, value) to the files.

The Warehousing Database is well suited for looking-up single objects such as files, jobs and their parameters. It is accessed in read and write mode.

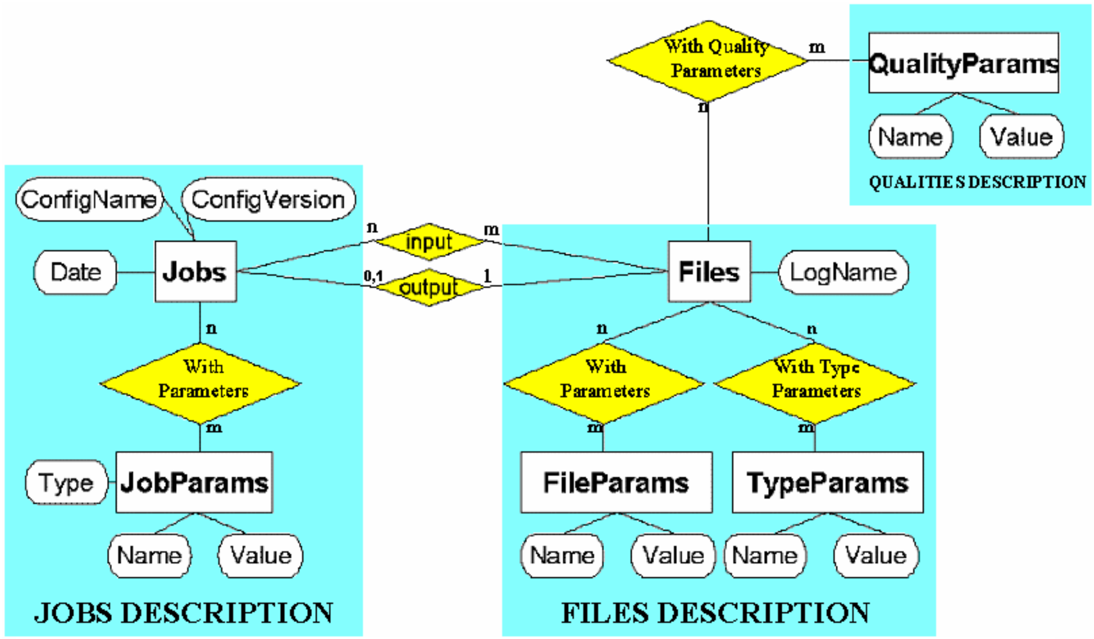

Figure 1: Entity-Relationship schema of the Warehousing Database.

The second schema is associated to a View of the WDB and is designed to be optimised for complex queries.

LHCb has two applications (fig 2) that need to interrogate the Bookkeeping to get information about datasets and both interact with physicists. This implies that a quick response by the applications is highly desirable and to achieve this, the applications access the information using Views of the WDB. The schema of the View is designed taking into account the application that is going to use it so it can be tailored to deliver the best performance.

Figure 2 shows the specialised View that sits on the back of the Ganga[3], the LHCB/ATLAS Grid user interface, and the Web Browsing applications. It is easy to notice how this schema is less intuitive than the WDB schema and this is because the schema has been designed keeping in mind the requirement of the two applications.

The Views are designed to be used by applications that want to fetch information from the Bookkeeping therefore they are read only and never updated.

Warehousing DB schema

The Entity-Relationship model (fig.1) shows that the model is designed around the two main entities: Jobs and Files.

- The job is the smallest unit, which the Bookkeeping deals with. It consists of a task, seen from the database as unbreakable, which has input and output files. Jobs are LHCb applications like Brunel and DaVinci that process data.
- Every job consumes and produces data, typically files, unless the job is part of the initial data processing chain. A file contains a collection, logically grouped together, of event data resulting from particle collisions.

While JobParams and FileParams entities are self-explanatory, TypeParams and QualityParams need a bit more explanation:

- The entity TypeParams is basically the description of the content of the file. Examples are “log file”, “DIGI”, “RAW data” or “Collection”.
- The entity QualityParams is used as a way to “sign” a given file as being useful (or useless) for a given analysis. It consists of a flag (“Good”, “Bad”, “Not Tested”) and of a recipient describing which type of analysis this flag is applicable to.

The relation between Files and Jobs is straightforward; the multiplicity of the relation has to be highlighted since it implies that a job can have more than one input file and at most one file as output.

The commonality of the entity parameters is that they have the coupled name-value as an attribute. This name-value property of the schema makes it very simple, but also very powerful and flexible.

Populating the Warehousing Database

During the 2004 data challenge[4] all of the metadata associated to the new datasets generated were not inserted directly into the bookkeeping but sent in XML file to the production manager. The XML files must follow a predefined XML schema that will guarantee that all the information is consistent. The production manager will run an application that checks the XML file against its own schema and uploads the information to the Bookkeeping being sure that the database has not been corrupted.

Generation of specialised Views

The specialised Views are a reorganisation of the metadata stored in the WDB and need to be generated every time the content of the WDB change. LHCb generates the Views every night; this is not a design constraint, it is just a decision of the experiment as to the most appropriate time for the physicists to have the generation-synchronisation of the View. This takes around 2.5 hours on our 20GB database. This is acceptable if we remember that the generation of the Views involves manipulating 2 million job entries that are related with 5.5 millions of file entries; both jobs and files are characterised all together by 57 million of parameters but because we believe that the dimensions of the bookkeeping will increase in the next years and so will the time for the generation of Views, part of our future plans is to investigate the scalability of this process.

WEB SERVICES TO ACCESS THE METADATA INFORMATION

To access the metadata, LHCb uses two services: a servlet service and XML-RPC service.

The servlet service is used for the web browsing of the metadata to identify the datasets on which analysis jobs can run (fig 2).

The XML-RPC service is used to access and modify the WDB content from applications and it is also used by Ganga to perform the selection of datasets (fig 2). The XML-RPC is implemented with two distinct interfaces: the Bookkeeping interface for reading access and the Bookkeeping editor interface for writing.

CONCLUSION

The strategy of using two schemas in the design of the File-Metadata Management System has proven to work, especially during 2004 when the data challenge enforced a heavy interaction and introduced newly defined metadata.

The flexibility of the WDB and the efficiency of the Views can be easily modelled to accept the requirement of other HEP experiments. Future work will involve the study of a more generic and cross experiment interface to be built on top of our bookkeeping. This will allow further testing of the system and determine whether it can be a more generic solution.

ACKNOWLEDGEMENTS

My contribution to this project has been supported by the GridPP collaboration with funding from the Particle Physics and Astronomy Research Council (PPARC).

REFERENCES

[1] LHCb, S. Amato et al, LHCb Technical Design Report, CERN-LHCC-2003-030, September 2003. http://lhcb.web.cern.ch/lhcb .

[2] http://intranet.cern.ch/Public/

[3] A. Soroko Alexander et al, The GANGA user interface for physics analysis on distributed resources, *these proceedings* [357].

[4] J.Closier et al, Results of the LHCb Experiment Data Challenge 2004, *these proceedings* [403].

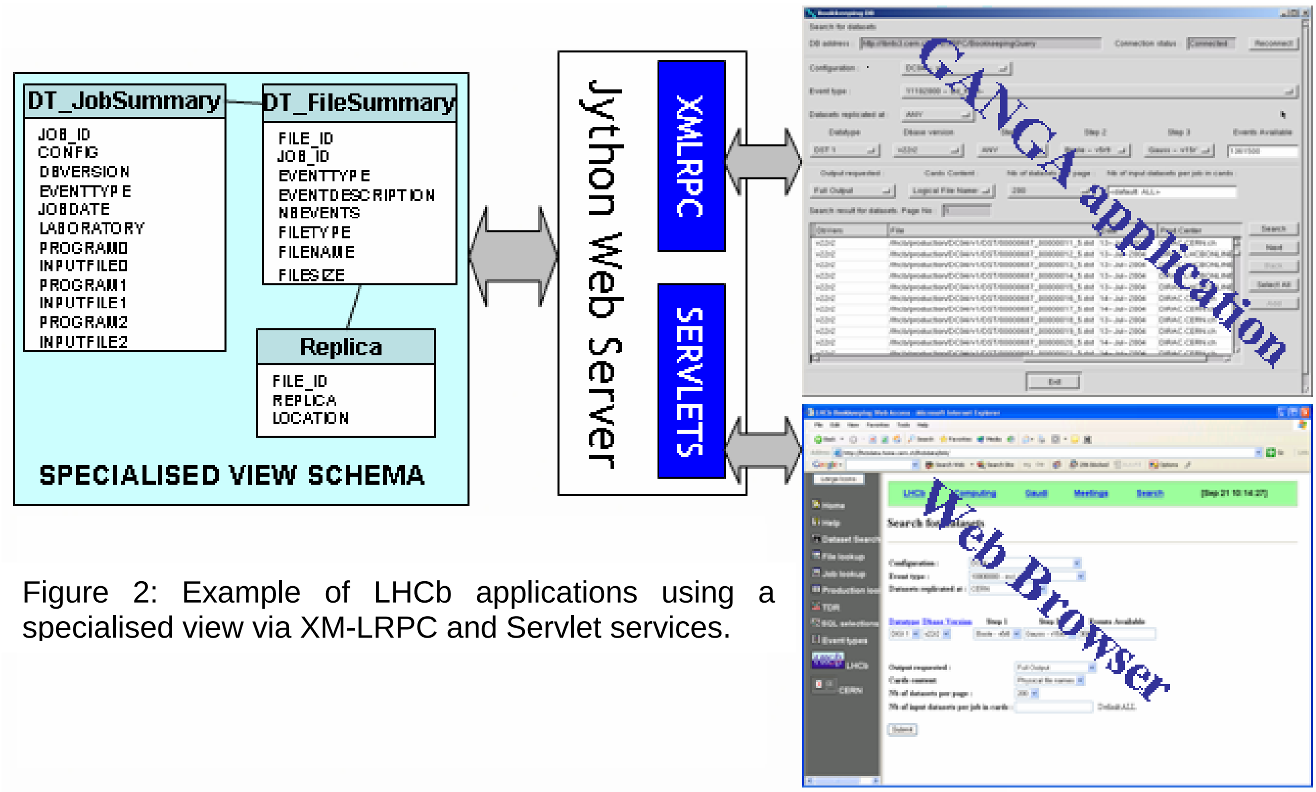

Figure 2: Example of LHCb applications using a specialised view via XM-LRPC and Servlet services.