Xrootd Data Transfers Monitoring for ALICE in Dashboard

S. Belov
e-mail: Sergey.Belov@jinr.ru,
Laboratory of Information Technologies, JINR, Dubna

Introduction

LHC experiments are continuously producing huge amounts of data (both real physics and simulation). To process these data distributed Worldwide LHC Computing Grid [1] computing is in use. The computing models of the LHC experiments are gradually moving from hierarchical data models with centrally managed data pre-placement towards federated storage which provides seamless access to data files independently of their location and dramatically improve recovery due to fail-over mechanisms. Files transfers have a significant part in this distributed data processing. Construction of the data federations and understanding the impact of the new approach to data management on user analysis requires complete and detailed monitoring. Monitoring functionality should cover the status of all components of the federated storage, measuring data traffic and data access performance, as well as being able to detect any kind of inefficiencies and to provide hints for resource optimization and effective data distribution policy. Data mining of the collected monitoring data provides a deep insight into new usage patterns. Here is presented an approach of integration of file transfers monitoring for ALICE experiment into the unified WLCG monitoring framework provided by the Dashboard [2]. All the collection chain is discussed: information export from MonALISA system, statistics aggregation, sending to the Dashboard via message broker, visualization.

Dashboard project

The Dashboard project for LHC experiments aims to provide a single entry point to the monitoring data collected from the distributed computing systems of the LHC virtual organizations. The Dashboard system is supported and developed in the CERN IT. From the 2007 the LIT JINR staff members contribute in the development of this system in the frames of collaboration with the CERN. Collected data are sent to the servers of the Dashboard system, where they are combined with data from other sources (the job submission tools, job wrappers and so on). It allows to get the right general picture of functioning of the infrastructure and the work of virtual organizations.

Federating storages with Xrootd

While the LHC data movement systems have demonstrated the ability to move data at the necessary throughput, two weaknesses were identified: the latency for physicists to access data and the complexity of the tools involved [3]. To address these, LHC experiments have begun to federate regional storage systems using Xrootd [4]. Referring to a protocol and implementation, Xrootd allows to provide data access to all disk-resident data from a single virtual endpoint.

In the WLCG context, there are several federations currently based on the Xrootd technology. Last time Xrootd federations are heavily used for data access and distribution by the main LHC experiments - ALICE, ATLAS, CMS, so federations monitoring is now of high importance for all these experiments. This concerns data transfers, data access and per-server statistics. The work was accomplished within the Dashboard monitoring project, which integrates different monitoring activities for all LHC experiments.

Data transfers monitoring for ALICE experiment

For ALICE, Xrootd federation includes about 70 sites having 300 servers in total, handling daily more than 1 million file transfers with 15 TB of data (according to ALICE's monitoring web-site [5]). Monitoring information in several fixed views is represented on the experiment's monitoring site. But having more detailed analysis or integration to other monitoring systems is complicated: experiment's software can not provide for export a real-time information on data transfers. This information could be gathered in a machine-sensible way from the MonALISA (or ML) [6] monitoring system using particular procedure. Each ALICE grid site has its own MonALISA service instance, where all ALICE specific monitoring information is being sent to and then is being integrated by so called repositories. On-site Xrootd servers are also being monitored in this manner.

The increase in Xrootd traffic has generated two new main requirements from the data traffic monitoring perspective. On the the first hand, the administrators of an Xrootd federation need a specific monitoring tool [7] with technology-specific features and the finest information granularity possible which includes local and remote data access.

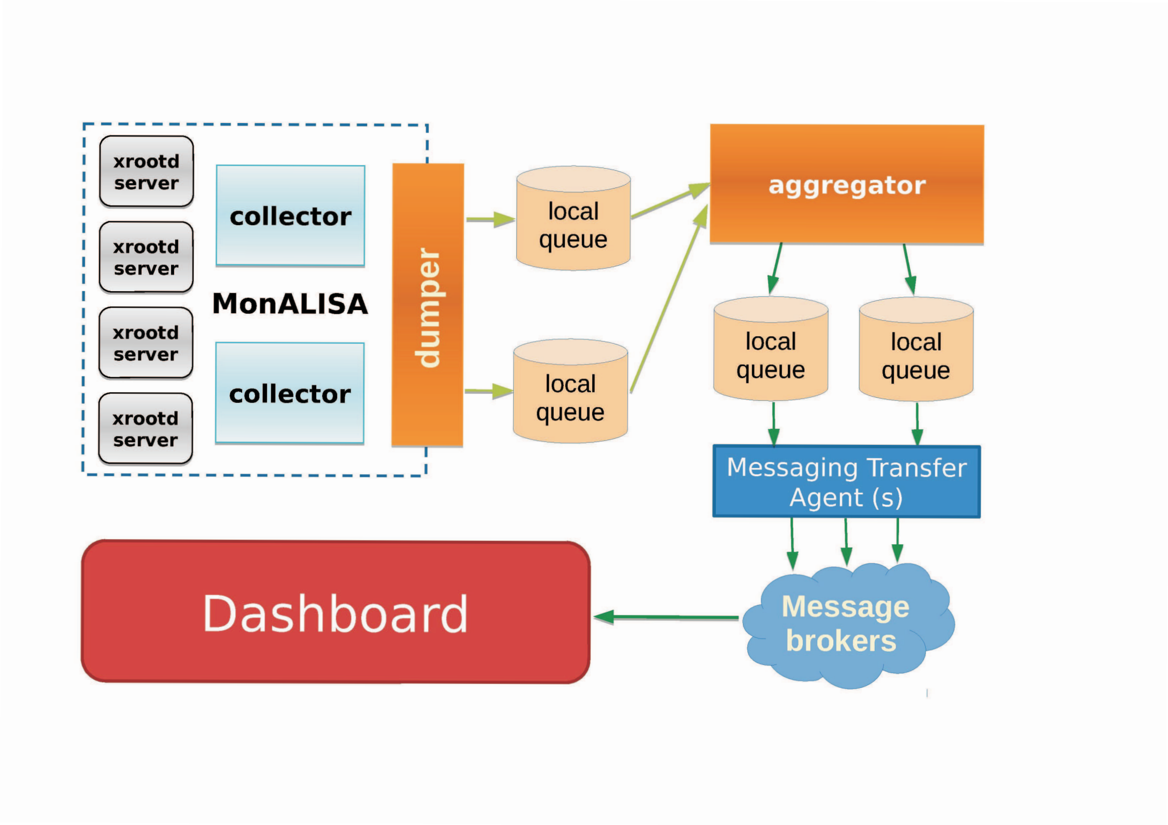

Figure 1: Xrootd monitoring information handling chain for ALICE experiment

On the other hand, all the WLCG traffic should be correlated in a single cross-vo and cross-technology user interface. To meet these requirements, the monitoring traffic of Xrootd is multicasted downstream of the message broker and then come to the Dashboard where goes to the database and then to the web interface [8].

Integration of developing Xrootd monitoring to the Dashboard impose some demands on it. Along with quite clear conditions such as no information loss and fault tolerance, there are several peculiar requirements coming from the Dashbaord framework:

- Sending information via message brokers network using AMPQ protocol;
- Messages to be presented in JSON format;
- Reliable messages handling along all the chain, assurance of information consistency;
- Reasonable behavior in sending messages: respecting connection frequency, authorization, timeouts; making few big messages instead of hundreds of small ones.

To meet these demands, following solution was proposed (Fig. 1):

- Data dumping from ML using special handlers (*dumper*) to local message queues;
- Aggregation of partitioned transfer messages or site-to-site transfers (*aggregator*);
- Sending collected information for further processing to Dashboard in reliable way by means of dedicated instance of Message Transfer Agent (MTA).

The *dumper* is a Java class catching incoming results from ML. It makes initial data transformation (decode IPs, etc.) and stores data to local directory queues. Then the *aggregator*, Python 2.4 program, consumes from the queues and assembles and groupes final messages to be sent by MTA. Directory queues use the same data structures but made in different programming languages. Both Java and Python libraries and *stompclt* used as MTA are provided by CERN IT. The Python implementation of local messages queues, *python-dirq*, and *stompclt* are also available in the standard EPEL package repository.

The *aggregator* process has three main threads: control one, worker and cleanup (for local queus). Main functions of the *aggregator* are:

- Accumulates statistics on xrootd servers (per timestamp), groups it by hostname;

- Reconstructs transfer statistics from subsequent messages, aggregates transfers by server and timestamp;
- Passes a bunch of messages (by type) in a large message to MTA;
- Removes all local queues messages involved when aggregated message is successfully sent;
- All semi-complete information chunks are to be sent on timeout, all (hopelessly) incomplete ones are wiped out.

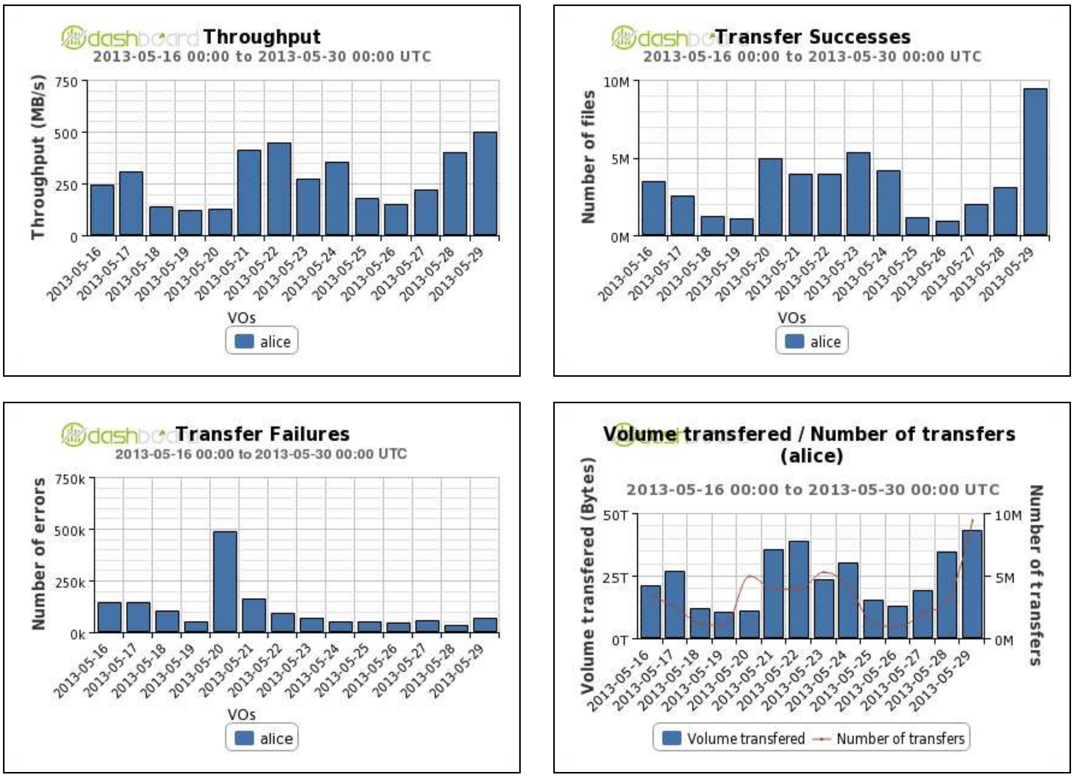

Figure 2: Web interface charts for ALICE transfers

Results and conclusion

Having in Dashboard data on individual file transfers and site-to-site statistics allowed to extract additional information on usage of Xrootd federation for ALICE experiment. First of all, along with number and volume of data files transferred, it is a per-site success and failure rates for transfers, both incoming and outgoing (Fig. 2). Monitoring information flow was integrated to the Dashboard just the same way as for other main LHC experiments, ATLAS, CMS and LHCb [9]. This made possible having a complete picture of WLCG traffic with no dependence of technology used (FTS, Xrootd or whatever) .

As a result of this activity, it was developed a way of integration of previously not covered monitoring of file transfers in ALICE experiment. This software solution was added to the Dashboard monitoring system in CERN and is in use now. The proposed approach of a reliable organization of information gathering chain will be also applied for monitoring of Xrootd servers for ATLAS and CMS experiments which are also using MonALISA framework for Xrootd servers monitoring.

Acknowledgments

Author is very grateful to the JINR Directorate for partial support of his work by providing a grant for young scientists and specialists (N 13-603-01).

References

[1] WLCG web page: http://wlcg.web.cern.ch

[2] J. Andreeva et al., *Dashboard for the LHC experiments*, J.Phys.Conf.Ser.119:062008, 2008.

[3] Xrootd project page: http://www.xrootd.org

[4] L. Bauerdick et al., *Using Xrootd to Federate Regional Storage*, J. Phys.: Conf. Ser., Volume 396, Issue 4, article id. 042009 (2012)

[5] MonALISA repository for ALICE: http://alimonitor.cern.ch

[6] MonALISA project: http://monalisa.cern.ch

[7] J. Andreeva et al., *Monitoring of large-scale federated data storage: XRootD and beyond*, Proceedings of CHEP'2013

[8] Dashboard web page: http://dashboard.cern.ch

[9] J. Andreeva et al., *Providing global WLCG transfer monitoring*, J. Phys.: Conf. Ser. 396 032005, 2012