Journal of Physics: Conference Series **396** (2012) 032005 doi:10.1088/1742-6596/396/3/032005

Providing global WLCG transfer monitoring

J. Andreeva (1) , D.Dieguez Arias (1) , S. Campana (1) , J. Flix (2) (4), O. Keeble (1), N. Magini (1) , Z. Molnar (1) , D. Oleynik (3) , A. Petrosyan (3) , G. Ro (1) , P. Saiz (1) , M.Salichos (1), D. Tuckett (1), A. Uzhinsky (3) , T. Wildish (5)

(1) CERN (Switzerland), (2) CIEMAT (Spain), (3) JINR (Russia), (4) PIC (Spain), (5) Princeton University (US)

E-mail: Julia.Andreeva@cern.ch

Abstract. The WLCG[1] Transfers Dashboard is a monitoring system which aims to provide a global view of WLCG data transfers and to reduce redundancy in monitoring tasks performed by the LHC experiments. The system is designed to work transparently across LHC experiments and across the various technologies used for data transfer. Currently each LHC experiment monitors data transfers via experiment-specific systems but the overall cross-experiment picture is missing. Even for data transfers handled by FTS, which is used by 3 LHC experiments, monitoring tasks such as aggregation of FTS transfer statistics or estimation of transfer latencies are performed by every experiment separately. These tasks could be performed once, centrally, and then served to all experiments via a well-defined set of APIs. In the design and development of the new system, experience accumulated by the LHC experiments in the data management monitoring area is taken into account and a considerable part of the code of the ATLAS DDM Dashboard is being re-used. The paper describes the architecture of the Global Transfer monitoring system, the implementation of its components and the first prototype.

1. Introduction

Overall success of the LHC distributed data processing depends, to a large extent, on fast, smooth and reliable data distribution. Huge data volumes are transferred between distributed sites. Aggregated transfer rate exceeds 10 GB/s. Various technologies are used for data movement and data storage and this contributes to the complexity of the data transfer organization and daily operations. Naturally, complete and reliable monitoring of the data management tasks, which include not only physical data transfer but also transfer bookkeeping procedures, is a vital condition for ensuring reliable operations and for efficient debugging of potential problems. Currently, for data transfer monitoring, all four LHC experiments rely on experiment-specific monitoring systems which are coupled with the data management systems of the experiments, namely ATLAS Distributed Data Management (ATLAS DDM) [2] system for ATLAS, AliEN[3] for ALICE , PhEDEx [4] for CMS, and Dirac [5] for LHCb. All those systems are mature and satisfy the needs of the experiments. However, all of them work only in the scope of a single virtual organization (VO) and therefore the global cross-VO picture of data transfers performed on the WLCG infrastructure is missing. Moreover, in order to provide the required information, all VO-specific transfer monitoring systems perform similar tasks. Those which use File Transfer Service (FTS) [6] as an underlying transfer system (ATLAS, CMS and LHCb) have to

regularly pull information from the distributed FTS services. Sometimes this implies screen-scraping since not all information is available in a convenient format. Naturally, this creates additional load on the FTS services. Some information, which is required for optimising data transfers, is not available though is being stored in the FTS local databases.

The new global monitoring system is not coupled with any of the existing VO-specific data management systems and provides a global view of all data transfers on the WLCG infrastructure regardless of the VO which initiates a particular transfer or the technology used for a given transfer operation. The system should perform common monitoring tasks such as collecting monitoring data, generating data statistics, and exposing data through the UI and APIs. It should assist the LHC computing community to perform data transfers in the most effective way and should allow potential problems to be easily detected and debugged. The LHC experiments share network links, and common transfer and storage services, therefore collecting monitoring information in a single repository will make it possible to perform data analysis on the complete transfer statistics and to find correlations between parallel transfer activities of the LHC experiments.

Currently, there are two main approaches to performing data transfers on WLCG. As already mentioned, ATLAS, CMS and LHCb rely on FTS, while the ALICE experiment uses xRootD [7] technology. Considering this situation, the initial development of the WLCG Transfers Dashboard monitors global data transfers performed using FTS. Monitoring of data transfers performed with xRootD is the next step. Though the current implementation of the WLCG Transfers Dashboard is focused on FTS and xRootD monitoring, the system design foresees the possibility of extension in order to enable monitoring of any other transfer technology which may be used in the future on the WLCG infrastructure.

All experience accumulated in the area of operating the LHC data management systems and in the development of VO-specific monitoring systems is being taken into account in the development of the WLCG Transfers Dashboard. This fact along with the active participation of experiment computing experts and developers, who provide a lot of feedback, ensures that the new system will satisfy the needs of the LHC community.

The first sections of this paper describe the concept of the system design, its architecture and user interface. The next section gives an overview of the FTS transfers monitoring implementation and of the current status of the first prototype. The following section is devoted to the monitoring of the xRootD federation and describes the plans to integration the xRootD monitoring flow in the global system.

2. Architecture

Similarly to the ATLAS DDM Dashboard [8] which was taken as a model for the implementation of the new system, the WLCG Transfers Dashboard is being developed in the Experiment Dashboard framework [8]. The main design principles are loose coupling of all components of the system and asynchronous communication with the information sources. Though the information sources formally do not belong to the WLCG Transfers Dashboard, they should be enabled with a capability to report monitoring data. The communication channel between the monitoring system itself and the information sources is the Messaging System for the Grids (MSG) implemented as ActiveMQ message brokers [9].

The common format defined in order to describe transfer events includes the source and destination endpoints, the status change of the transfer, corresponding time stamps, the amount of transferred data, the names of the transferred files and, where relevant, the exit code. Data reported from the information source to the MSG is consumed by the Dashboard collector and is stored in the central data repository. Periodic aggregation procedures generate transfer statistics which are exposed via web UI to users and via web APIs to client applications. The implementation of the central repository schema, the statistic generation procedures and the user interface was to a large extent inherited from the DDM Dashboard which allowed the first prototype to be deployed in a very short time of an order of a couple of months. The architecture of the system is presented in Figure 1.

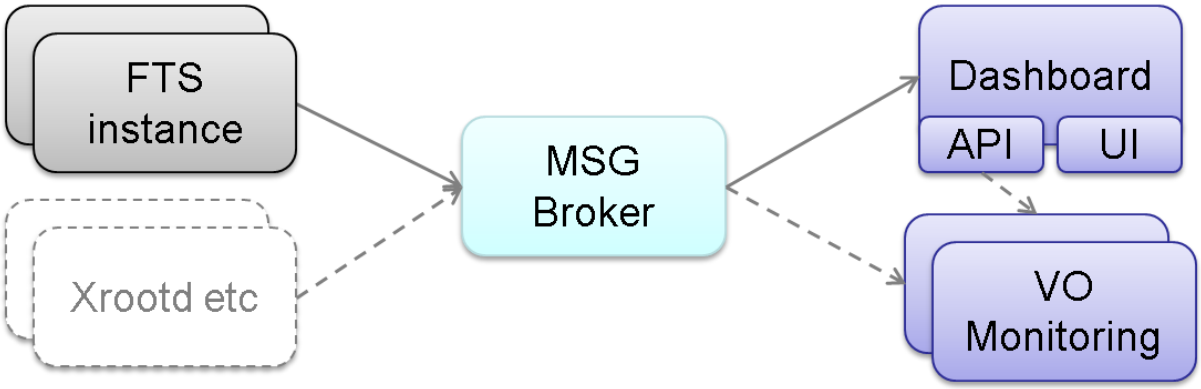

Figure 1. Architecture of the WLCG Transfers Dashboard.

The new system foresees that the potential clients (most probably monitoring systems of the LHC VOs) can obtain aggregated information from the WLCG Transfers Dashboard through the APIs or can consume raw events directly from the MSG.

3. User Interface

The user interface is a very important component of every monitoring system and one of the most time consuming parts of the development process. Thanks to the fact that the ATLAS DDM Transfers Dashboard has been successfully used in production for several years and its implementation took into account accumulated operational experience and the feedback of the user community, implementation of the user interface for the WLCG Transfers Dashboard was a straight forward process. As it was already stated above, to a large extent it shares its implementation with the ATLAS DDM Dashboard. A core design principle of the Experiment Dashboard framework is the existence of an abstraction layer between the data repository, which is normally implemented in ORACLE, and any system component which needs to access the data repository either for reading or writing. Clients send requests via a web server, data is provided in machine readable format (XML or JSON) which is either exposed to the client applications via APIs or is used as content for the web user interface. The visualization of the user interface is implemented in the xbrowse [10] visualization framework. The xbrowse visualization framework is a client-side JavaScript implementation developed for two dimensional data structures which should be represented in the form of matrix. The framework is currently used for the UIs of the WLCG Transfers Dashboard and ATLAS DDM Dashboard, but is generic enough to be used for other applications. The key features of the UI are flexible filtering and grouping, a matrix view presenting a snapshot of the transfer status between any selected set of sources and destinations, the possibility to drill down to error samples, and customizable plots for transfer metrics shown as a function of time.

The system uses the GOCDB/OIM [11] naming convention for sites in the cross-VO view and VO-specific naming for single-VO views. An example of the snapshot matrix view is shown in Figure 2 and a variety of plots showing transfer statistics as a function of time is shown in Figure 3.

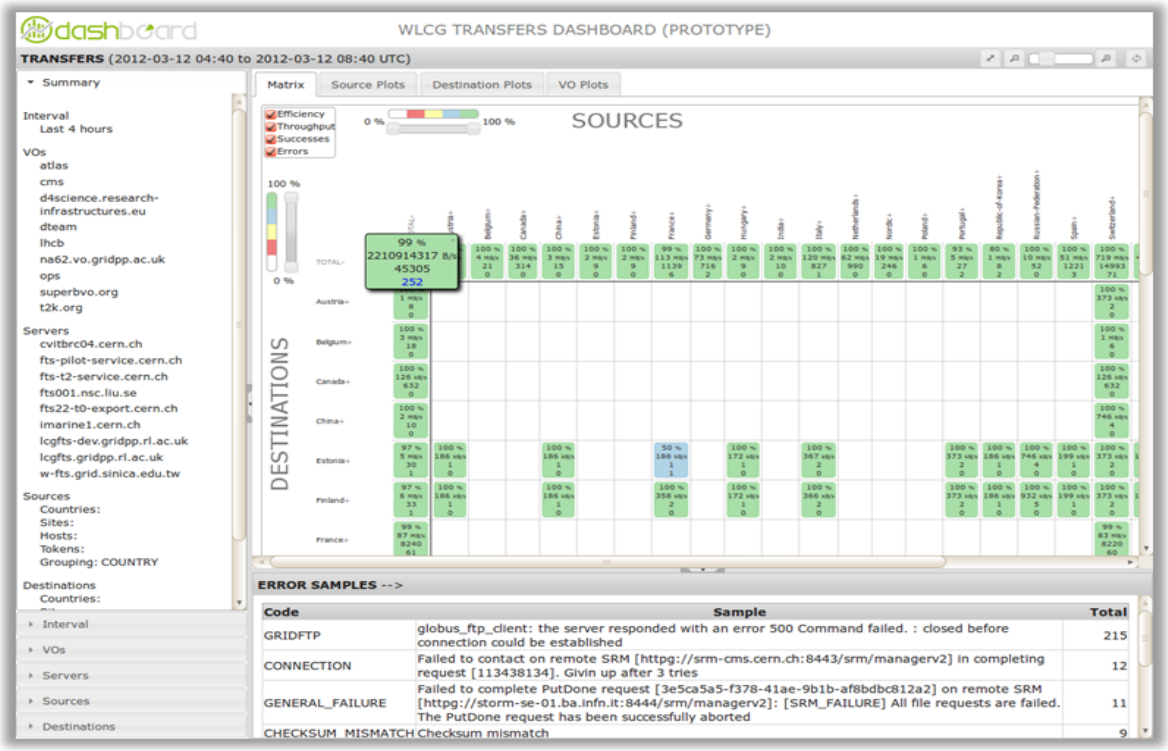

Figure 2. Snapshot matrix view

4. FTS monitoring

4.1 Implementation

In order to enable global monitoring of FTS transfers, FTS servers should be instrumented for reporting monitoring information to the MSG server. FTS transfer workhorse (transfer-url-copy) has been instrumented to collect and produce information for individual transfer monitoring messages.

The payload (message) is represented in JSON format. Once the message is produced, it is written to a named pipe. On the other side of the pipe there is a daemon process which is reading the messages and sends them over to the broker. The content of the reports was discussed and agreed with transfer experts in the LHC experiments and took into account operational requirements. There are two types of messages. The first one which describes real time transfer events (transfer started or accomplished) and the second one is sent regularly and contains a snapshot of the status of the transfer queues. The authentication of the FTS instances reporting to MSG is performed with username/password, while the Dashboard consumers are authenticated using host certificates. Dedicated MSG brokers were deployed in order to handle the transfer monitoring flow. Specific topics were created for both types of messages. Information reported from the FTS servers can be of interest not only to the WLCG Transfers Dashboard but also to other potential clients, for example the GridView monitoring system, and the monitoring systems of the LHC experiments, therefore the corresponding virtual queues were created on the message brokers dedicated to the WLCG Transfers Dashboard consumers. The data repository contains tables for raw messages consumed from the MSG. Its structure is consistent with the message format. Raw data is used in the generation of transfer statistics which are recorded in summary tables. The minimum granularity of time bins for the transfer statistics is 10 minutes. For optimal access, the table with raw data is partitioned daily and partitions older than 90 days are cleaned, whereas statistics are kept indefinitely.

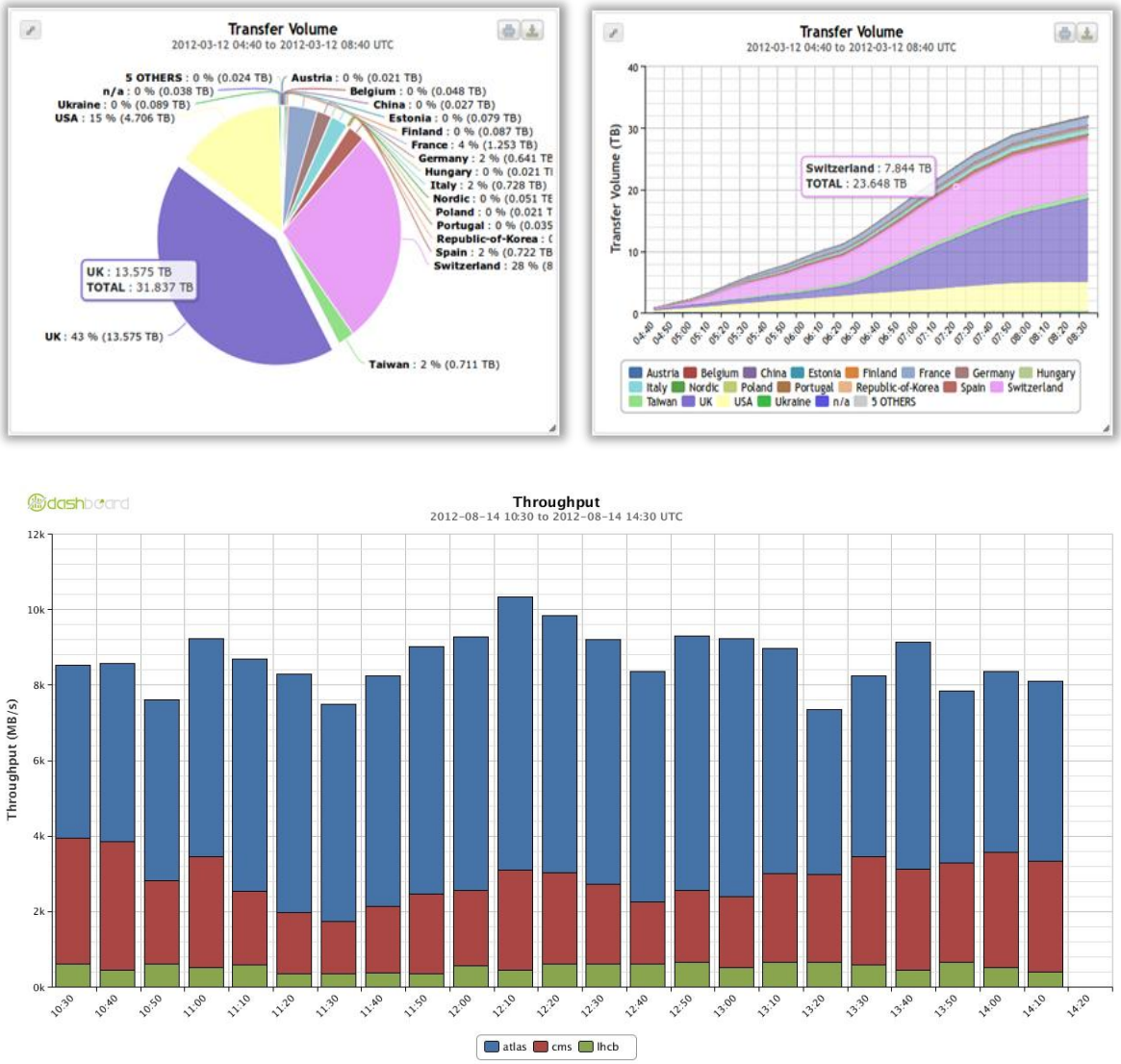

Figure 3. Examples of plots showing transfer metrics as a function of time.

4.2. Current status

The first prototype of the WLCG Transfers Dashboard, initially covering only FTS transfers going through the FTS pilot service at CERN, was setup in autumn 2011 as soon as FTS version 2.2.8 with transfer status reporting capabilities enabled was deployed to the CERN FTS pilot service. After deployment of FTS version 2.2.8 to all FTS instances, the first round of validation was performed by the FTS transfer experts from the CMS and ATLAS experiments. Problems in the UI detected during validation were fixed.

The schema of the data repository was validated by the ORACLE DBAs. After these steps, the system was moved towards a production setup. This included creating the data repository on the production DB cluster, redirecting reporting to the production message brokers, and setting up the data collectors and UIs on two hosts for redundancy. Since monitoring information is coming from the distributed FTS servers, alarms are enabled in case any of the FTS servers stops reporting for more than 2 hours. Integration and testing environments were created as well. These include the integration DB server, test message broker and test versions of the collectors and UIs running on a dedicated virtual machine. Over the last few months the system has proved to work reliably.

However, one of the most important conditions for a system to be announced in production was not satisfied. In order to ensure the quality of the provided data, consistency checks between the WLCG Transfers Dashboard and the VO-specific monitoring systems PhEDEx and ATLAS DDM Dashboard were performed. Initial results of these consistency checks showed considerable discrepancy of information provided by the global and the experiment-specific systems and indicated an evident problem with the information provided by the new system. After investigation the problem was understood. It was caused by a known bug in the activeMQ-cpp client which prevented reconnection of the client to a message broker in case of a broken connection. The problematic version of the activeMQ-cpp client was used by the FTS publisher. This naturally resulted in a large number of missing data transfer monitoring reports. A workaround was found and the FTS servers were patched in order to fix the problem. The next round of consistency checks demonstrated perfect agreement between the global and experiment-specific systems. Figure 4 shows two transfer rate plots for the same 12 hours time range interval for transfer from RAL-LCG2 to CERN . Right side plot is taken from the WLCG Transfers Dashboard the one on the left side is from PHeDeX. In order to ensure the trustworthiness of information, consistency checks are being performed daily and Dashboard operators will be notified by alarm in case of significant discrepancy. In addition, a dedicated administrative UI was setup in order provide a straightforward way to check the consistency of the WLCG Transfers Dashboard and the experiment-specific monitoring systems. Figure 5 shows a plot taken from the consistency checks administrative UI. This plot demonstrates the relative difference in the transfer rate from RAL-LCG2 to CERN between PHeDeX and the WLCG Transfers Dashboard for the same time range interval which is used for the plots shown in Figure 4. PHeDex and the WLCG Transfers Dashboard use different methods for getting their monitoring information. As a result, as shown on the plot Figure 5, some discrepancies in the transfer rate measured by both systems are explained by eventual difference in time when the monitoring reports are accounted, however, on a long enough timescale, the two agree.

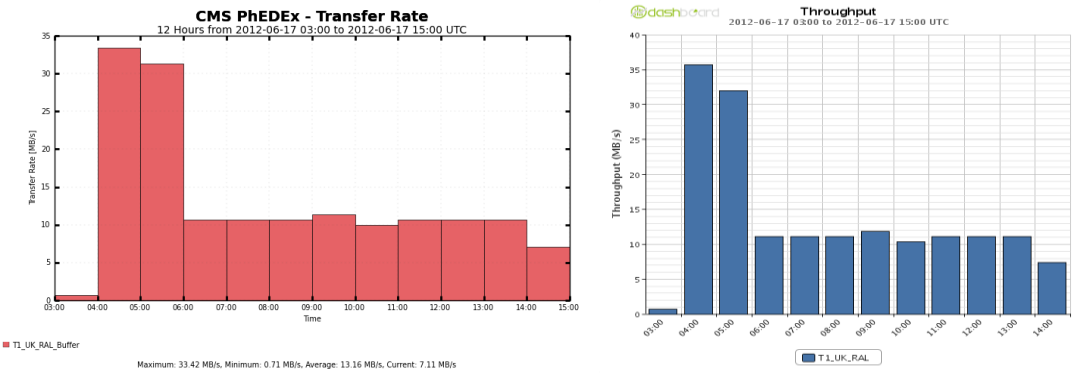

Figure 4. Plots demonstrate consistency of information provided by WLCG Transfers Dashboard and PHeDeX for transfers from RAL-LCG2 to CERN. On the right side there is a WLCG Transfers Dashboard plot, on the left side there is a PHeDeX plot.

Figure 5. Example of the WLCG Dashboard Consistency Checks Administrative Interface. Plot demonstrates relative difference of the transfer rate measured by PHeDeX and WLCG Transfers Dashboard for transfers from RAL-LCG2 to CERN

5. Enabling monitoring of the xRootD transfers

Integration of data transfers monitoring handled via xRootD in the global transfer monitoring system is a more complicated task than enabling FTS monitoring. The xRootD servers have a built-in capability to report monitoring information via UDP. Monitoring information is split into two flows: summary and detailed data flows. However, the format of the xRootD data flows is very different to the event-like FTS information describing a single transfer. Event-like monitoring information should contain the time interval when a particular transfer operation happened, source and destination, names of transferred files and the amount of transferred data. XRootD summary data flow does not contain all this information. Detailed data flow is much more complete, therefore only by processing the xRootD detailed data flow and aggregating transfer metrics reported in this flow over a given time, can the transfer attributes mentioned above be resolved and event-like messages be generated as expected by the Dashboard consumer. An additional complication for xRootD monitoring is the lack of topology information. In the case of FTS, FTS sources and destinations can be mapped to a particular WLCG site using the VO topology descriptors which are provided by every LHC VO. In the case of the xRootD transfers, source and destination are defined just with IP addresses. IP addresses are not always sufficient to resolve the name of the corresponding computing site.

Taking into account the conditions mentioned above, the suggested architecture of the xRootD transfers monitoring consists of several levels of hierarchy. Depending on the chosen deployment model there can be three levels (local site – Federation - Global monitor) or two levels (Federation – Global monitor), see the xRootD monitoring architecture with 3-level hierarchy in Figure 6. The hierarchy of the deployment model is defined by the location of the daemon which reads xRootD detailed data flow, reformats it into event-like messages and publishes these messages to the MSG. In this paper this daemon is referred to as the xRootD monitoring publisher, though it consists of two components: a collector of the xRootD monitoring data and a publisher to MSG. Deploying the xRootD monitoring publisher once per a xRootD server corresponds to the 3-level hierarchy model. Deploying the xRootD monitoring publisher once per an xRootD federation corresponds to the 2-level hierarchy model. The 3-level hierarchy deployment model has advantages with regards to scalability and the possibility to reconstruct the federation topology. Every report sent by the xRootD monitoring publisher can be complemented by the name of the computing site. This will allow the federation topology to be easily reconstructed. Moreover, in the 3-level hierarchy model, processing of the detailed data flow is not performed centrally (on the federation level), therefore the system will scale well. On the other hand, the 3-level hierarchy deployment model requires multiple distributed instances of the xRootD monitoring publisher to be monitored in order to make sure that all of them

are up and are properly reporting. The current implementation foresees the possibility of both deployment models.

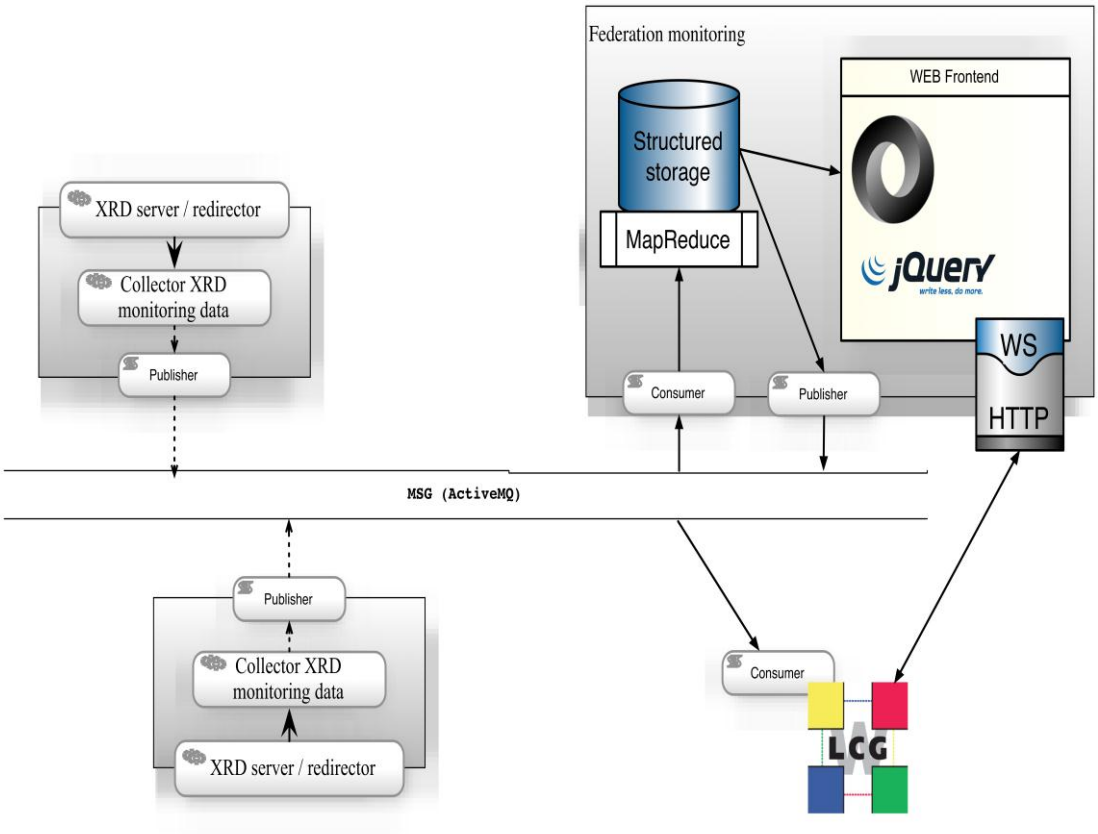

Figure 6. xRootD transfer monitoring architecture, 3-level hierarchy deployment model

Figure 6 describes the xRootD monitoring architecture with the 3-level hierarchy. The xRootD monitoring publisher sends event-like monitoring reports to the MSG. On the federation level, these reports are consumed and are used in the generation of federation statistics similar to the statistics of the global WLCG Transfers Dashboard. Hadoop is chosen for persistency and MapReduce for the generation of the federation statistics. Federation monitoring statistics will be exposed via the Federation UI which is very similar to the UI of the WLCG Transfers Dashboard. As described in section 3, the WLCG Transfers Dashboard UI is implemented as a client-side JavaScript application which uses input in JSON format. This design of the UI allows it to be easily adapted it for any kind of information source assuming that the latter provides input in JSON format. Finally, on the federation level there is an xRootD federation publisher which publishes federation aggregated statistics to MSG. The WLCG Transfers Dashboard collector consumes federation statistics and stores them in the central data repository.

Integration of the xRootD monitoring in the global WLCG transfer monitoring system is currently in the active development phase. It re-uses some components which were developed for the monitoring of the US CMS xRootD federation [6] and those which were developed in the framework of the ATLAS Tier3 monitoring project [11]. The whole chain starting from the xRootD monitoring publisher to the visualization of the xRootD transfers via the WLCG Transfers Dashboard UI should be enabled by the end of 2012.

6. Future plans

Apart from the development effort for the integration of xRootD transfer monitoring in the WLCG Transfers Dashboard, the future development is driven by the requirements of the LHC VOs. Basic functionality enabled in the current version is enough to monitor transfer throughput and to detect transfer failures between any given destination-source. However, it is not enough for debugging inefficient data transfers or to detect and understand transfer latencies which can be caused by various reasons. Most of the VO requirements aim to address these use cases. For example new functionality which should be enabled in the WLCG Transfers Dashboard is monitoring of the FTS transfer queues and monitoring of transfer latencies caused by SRM. Additional plots, for example, quality maps and ranking plots should be added to the UI.

7. Collaborative effort

Development of the WLCG Transfers Dashboard monitoring system is an example of very effective collaboration of different groups inside CERN IT department (ES, GT, PES, DB) , between CERN IT and LHC experiments, and between CERN and JINR (Joint Institute Nuclear Research, Russia, Dubna). The participation of the user community from the LHC computing teams is mandatory for the success of the project. This includes the definition of requirements, validation and operational experience. As was mentioned above, ATLAS and CMS take a very active role in the development process. For example, functionality prototyped by the CMS computing team for the FTS monitoring will be enabled in the WLCG Transfers Dashboard [13].

8. Conclusions

The new global data transfer monitoring system which aims to monitor data transfer on the WLCG infrastructure across various VOs and across different technologies was put in place in a short time, of the order of several months. It is a result of the collaboration of developers at CERN IT, the LHC experiments and JINR. The initial version provides monitoring of data transfers performed via FTS. Enabling monitoring of xRootD transfers is in active development phase. The system was deployed in production after careful validation and consistency checks between the new system and VO-specific data transfer monitoring systems, namely PhEDEx and ATLAS DDM Dashboard. Experience accumulated in the operating of the WLCG infrastructure and in particular in operations involving data distribution was taken into account for the development of the WLCG Transfers Dashboard. A large proportion of code of the ATLAS DDM Dashboard was re-used in the new system which to a large extent accelerated the development process. Future development will follow the needs and requirements of the LHC computing community.

References

[1] *LHC Computing Grid Technical Design Report*, https://espace.cern.ch/WLCG-document-repostory/Technical_Documents/TDR/LCG_TDR_v1_04.pdf
[2] Garonne V et al, 2012 *The ATLAS Distributed Data Management Project*, J. Phys.: Conf. Ser. (not yet published)
[3] Saiz P et al, 2012 *AliEn: ALICE Environment on the GRID* J. Phys.: Conf. Ser. (not yet published)
[4] Sanchez-Hernandez A, Egeland R, Huang C-H, Ratnikova N, Magini N and Wildish T, 2012 *From toolkit to framework - the past and future evolution of PhEDEx*, J. Phys.: Conf. Ser. (not yet published)
[5] Tsaregorodtsev A et al, 2009 DIRAC3 - *the new generation of the LHCb grid software* Proc. of CHEP09
[6] Molnar Z et al, 2012 *Next generation WLCG File Transfer Service* FTS, J. Phys.: Conf. Ser. (not yet published)

[7] Bockelman B et al, 2012 *Using xRootD to federate regional storage*, J. Phys.: Conf. Ser. (not yet published)
[8] Andreeva J et al, *Experiment Dashboard for monitoring computing activities of the LHC Virtual Organizations*, J. Grid Computing, 2010 8:323-339
[9] Cons L and Paladin M, 2012 *The WLCG Messaging Service and its Future* J. Phys: Conf. Ser. (not yet published)
[10] J. Andreeva et al, 2012 *Designing and developing portable large-scale JavaScript web applications within the Experiment Dashboard framework* J. Phys.: Conf. Ser. (not yet published)
[11] Mathieu G et al, 2009 *GOCDB/OIM a Topology Repository for a Worldwide Grid Infrastructure*, Proc of CHEP09
[12] Oleynik D et al, 2012 *ATLAS off-Grid sites (Tier 3) monitoring. From local fabric monitoring to global overview of the VO computing activities*, J. Phys.: Conf. Ser. (not yet published)
[13] Flix J et al, 2012 *Performance studies and improvements of CMS Distributed Data Transfers*, J. Phys.: Conf. Ser. (not yet published)