Data management in WLCG and EGEE

F. Donno, M. Litmaath
C E R N
1 February 2008
v1.2 – 31 March 2008

1. The WLCG infrastructure

The Worldwide LHC Computing Grid (WLCG) [19] is the largest Grid infrastructure in operation today, comprising more than 250 sites spread over 45 countries on 5 continents. Its main mission is to provide resources for the storage and analysis of the data generated by the experiments at the Large Hadron Collider (LHC) facility currently being commissioned at CERN, the European Laboratory for Particle Physics. There are four major experiments at the LHC: ALICE, ATLAS, CMS, and LHCb. Each experiment has its own computing model, but all rely on the WLCG for the necessary storage and computing resources. The WLCG itself comprises a federation of sufficiently compatible Grids. The main contributors currently are the Enabling Grids for E-sciencE (EGEE) project [20], the Open Science Grid (OSG) [21], and the Nordic Data Grid Facility (NDGF) [22]. It is important to note, however, that each of the Grids contributing to WLCG has been explicitly funded to provide infrastructures for e-Science in general, in particular for sciences other than particle physics, in contrast with the main mission of the WLCG. Other disciplines include biomedical research, computational chemistry, nuclear fusion, astronomy, geophysics, meteorology, and digital libraries. They usually are new to the ways of working in large international collaborations that are normal in particle physics since tens of years. While many of the practices used in particle physics analysis may simply be copied, other disciplines also bring additional requirements, in particular with respect to security and privacy. Furthermore, most WLCG sites also participate in national or other international Grids, which may yet pose other requirements on the services that some of the sites need to provide. Finally, the large WLCG sites have a history of providing computing and storage resources for pre-Grid scientific projects and typically have built up significant storage infrastructures that will also have to be used by grid projects. That is, data storage and retrieval on the Grid will have to be compatible with pre-existing basic infrastructures that will even differ significantly from site to site. In particular, CERN and other big computing centres each have their own mass storage systems, typically since many years. The different tape back-end systems include CASTOR, DMF, Enstore, HPSS, and TSM. The corresponding HSM front-end systems in use are CASTOR [10] and dCache [11]. Disk-only storage is provided by dCache, DPM [23] and StoRM [24] systems at this time. Some OSG sites use BeStMan [25], which supports HPSS amongst other back-ends.

1.1 The Tiers model

The four experiments together are expected to produce between 10 and 20 PB of data per year. To aid in handling such very large quantities efficiently, the WLCG infrastructure has been divided into "Tiers". At the lowest level is the Tier-0, CERN itself, where the experiments record their data and perform a first-pass analysis over it. One copy of this data is saved on tape in the computer centre at CERN, while another copy is spread over 11 Tier-1 centres. These are large computer centres in Europe, the USA, Canada, and Taiwan, all connected to CERN through dedicated network links of at least 10 Gbps. Each Tier-1 centre will save its fraction of shipped data on tape as well. Later the Tier-1 centre will typically reprocess the data with better calibration parameters or improved software versions. The subsequent output is saved on tape and may need to be copied

to one or more partner Tier-1 centres for better availability of the data for subsequent analyses. At the next level there are about 100 Tier-2 sites. A Tier-2 site normally comprises a CPU farm and disk storage of up to a few tens of TB. Most of the analysis is expected to be done at Tier-2 sites, which will download the necessary input data files from the Tier-1 sites. A Tier-1 site will have a cloud of Tier-2 sites around it, often in the same country or larger region, and dependent on support from their Tier-1 centre. Some of the experiments foresee their Tier-2 sites to download large amounts of data from Tier-1 centres in other regions, though. A Tier-2 site will also be used to produce simulated data, which it will upload to its Tier-1 centre for safekeeping and further distribution as needed. A Tier-3 site typically amounts to a CPU farm at a university participating in one of the experiments. Tier-3 resources typically are used opportunistically. To complicate matters further, CERN also acts as a Tier-1 centre and any Tier-1 centre can also act as a Tier-2 site.

1.2 The WLCG Data Management Services and Clients

The WLCG infrastructure makes available a set of data management services and interfaces that the LHC experiments can use in order to implement their data models.
The gLite data management client tools allow a user to move data in and out of the Grid, replicate files between Storage Elements, interact with a File Catalog and more. High level data management clients and services shield the user from the complexities of the storage services and catalog implementations as well as transport and access protocols. Low level tools and services are also available to achieve uncommon tasks.
The **File Transfer Service (FTS)** [1] allows for the scheduling of the transfer of data files between sites. The service is configured to allow for transfers only on predefined channels between peers configured at service startup. The FTS uses low level services and tools to perform data transfers and related operations.
Another data management library worth mentioning is the **Grid File Access Library (GFAL)** [1]. It interacts with Grid File Catalogs and storage services via the available control protocols. It allows applications to access files using abstractions such as the "Logical File Name" (LFN), a human-readable identifier of a file in the Grid. Once presented with an LFN, the GFAL library contacts a Grid File Catalog (LFC) to retrieve a handle to the best replica available. Then, it negotiates with the corresponding storage service to determine which file access protocol will be used (POSIX, gsiftp, rfio, gsidcap, etc.).
Because of historical reasons and the untimely availability of general solutions, the WLCG experiments have developed their own data management frameworks to various degrees. For instance, a transfer service such as PhEDEx [2] developed by the CMS collaboration, could have evolved to fulfill the role of the FTS, i.e. scheduling not only within but also between concurrent experiments. Instead, PhEDEx has been modified to drive the common FTS from the CMS perspective. Another protocol for efficient file transfer and access has been developed by SLAC. This is the xrootd [3] system that is being considered in particular for end-user analysis.

2. High Energy Physics Use cases

Although there are different HEP experiments within the WLCG project, all of them follow a common way of organizing their basic distributed computing model. We first describe the general computing and data model that is applicable to all four experiments and outline experiment specific differences later whenever necessary.

2.1 Constraints for Distributed Computing and Storage

All four experiments have the following items in common which can be regarded as the main constraints for a distributed computing model:

- **Central data recording**: Data coming from the experiment detectors (raw data) is recorded at CERN. Data is typically written once and never updated (i.e. **read-only data**).
- **Large data storage**: Each experiment produces a few (5 to 10) Petabytes of data each year that need to be stored permanently.
- **Data processing**: Raw data needs to be processed in order to extract and summarize information that has relevance to physics. This processing of data is called ***reconstruction*** in HEP terminology and is typically very computing intensive. The storage requirement for reconstructed data is smaller than for raw data but still in the order of many Terabytes to a few Petabytes per year [4][5][6][7].
- **Distributed computing and storage centers**: CERN is considered to be the main centre to provide storage and processing power. However, each of the 4 experiments consists of a collaboration of many countries, almost all of which provide storage and computing capacity that is dedicated to one or more of the experiments. In this way, the overall computing power and storage capacity available to a single experiment are increased.
- **Distributed user community**: A few hundred research institutes and universities participate in the LHC experiments, with physicists (the actual end users) distributed over the globe. Their common goal is to analyze physics data as if all of it were available locally, having transparent access and good response time also when accessing data remotely.

3. High Level Physics Use Cases

In the following section we give an overview of the basic high-level use cases for the computing usage in High Energy Physics. These use cases are representative of the data model explained in the previous section.

3.1 Reconstruction

The raw data, whether real or simulated, must be reconstructed in order to provide physical quantities such as the identities, positions and momenta of the particles of interest. The pattern recognition algorithms in the reconstruction program make use of calibration and alignment parameters to correct for any temporal changes in the response of the detectors and their electronics. This process is computationally very intensive and needs to be repeated a few times in order to accommodate improvements in the algorithms, in calibration and alignment parameters. Therefore, it cannot be executed entirely at CERN. Raw data is stored on tape at CERN and streamed to Tier 1 sites where the reconstruction program should start shortly on data just arrived. For this use case the storage requirements are the following:

- Specific data transfer servers with *WAN access* and adequately large buffers need to be in place in order to efficiently receive data coming from the Tier 0.
- *Discovery functions* should allow for the identification of the data services and buffers dedicated to the given experiments.
- Data transfer services should allow for *reliable* and *secure transfer* of big buffers of data. Such services should provide users with transfer scheduling and retry functionalities.
- Data transfer servers must be connected to the *tape storage* systems for persistent storage of the data.
- A proper *storage interface* to mass storage systems should be available in order to trigger and control store and stage operations in an implementation independent way.
- Given the amount of data involved, it is desirable to avoid making multiple copies of the data. Therefore, the data needs to remain on disk for a time sufficient to reconstruct it,

before it is deleted to make space for new data. The *"pinning" functionality* allows for specifying a lifetime associated to the data stored in a given space.

- For a critical operation such as reconstruction of physics data, it is mandatory not to compete for resources with other experiments. Therefore, *dedicated resources* are normally required by the experiments.
- Furthermore, it is important that user activities do not interfere with production or import/export activities. Support is required for access control lists on spaces provided by the storage services, as well as mechanisms to block unwanted types of access to specific data buffers.

3.2 Main Stream Analysis

This use case can be considered as the standard, scheduled activity of a physics group in a certain university. The research group is interested to analyze a certain data set (typically consisting of many Giga- or several Terabytes of data) in a certain Tier 1 centre that has free computing capacity. If the data is not available at that site, it needs to be transferred in a scheduled way and the operation might last for a few days. Once the data has arrived, computing-intensive physics analysis operations can be done on the specified data. For instance, the validation of reconstructed data is a process in which the validity of the used algorithms and parameters is assessed. This process implies access to 1-2% of the total reconstructed data of an experiment. It might imply running variations of the program several times on the same set of data. Once the process is finished, the result is stored on tape.

The implicit storage, data and transfer requirements are as follows:

- Data needs to be accessible from the storage system, i.e. mass storage systems, disk systems as well as the corresponding data servers need to provide the required performance.
- Data transfer tools need to be in place that have access to the source storage system and can transfer data to another storage system at a different site/Tier. Since the physics activities and therefore also the data transfers are scheduled, the latter can be optimized: bandwidth can be "reserved" by *prioritizing* the requests of a particular physics group and reducing the ones of other physics groups or individual users.
- Once data has arrived at the site, computing and storage *resources* must be dynamically or statically *reserved* for a particular user group.
- It should be possible to express *ownership of resources* and specify *authorization patterns*.
- In order to ensure resource sharing, *quotas* should essentially be enforced in a transparent way so that several groups within the experiment or even multiple experiments can concurrently use the resources at a site.
- Resource usage and *status* should be *monitored* and published so that busy resources are not selected for further computing and/or storage tasks.
- If the needed data is on tape, it must first be transferred to disk for online access. Therefore, transparent staging tools must be available.
- Specific *file access protocols* need to be supported by the storage facility, so that applications limited to using only those protocols can be executed.
- Once data is analyzed, the relevant output can be saved on tape if deemed important. Therefore, tools to archive the results on tape and register them on the Grid are necessary.
- Physicists should be provided with the necessary tools to *manage space*, for instance in case the storage system does not remove unneeded files automatically.

Grid operators and/or site administrators that take care of the execution and monitoring of data transfers as well as the allocation of CPU power to the physics group can further support and optimize the actual execution of this use case scenario.

3.3 Calibration Study

During the run of the LHC, particles pass through detectors that have to be aligned and calibrated in order to allow for correct physics analysis. A physics group might work on the calibration study and detect problems with the calibration and alignment parameters. In such a case, some of the reconstruction algorithms need to be rerun and new reconstructed data needs to be stored.
In this use case, there is a request for fast access to a substantial subset of the data and for a large amount of computing power at peak times. This may involve transferring raw data from tape to disk. Many tape drives can thus be busy in this task that typically has high priority. Once the set of calibration parameters prove to be accurate, they are stored in experiment specific databases that are distributed to a few sites for reasons of performance and fault tolerance.

3.4 Chaotic Analysis

In contrast to the scheduled "main stream analysis" of a particular physics group, here a single physicist working on a specific analysis might request access to a data set which can be of "any" size, i.e. it is not known a priori how much data would need to be made available locally or accessed through the WAN.
This use case is of particular importance for physicists, system administrators, operators, and developers, since it can create worst-case scenarios that stress the system. This use can also help detect scalability issues in many parts of the data access and storage system.
Because of this unpredictable behaviour, it is very important to be able to control storage resource usage and access accurately in order to prevent problems. In particular, *quota and dynamic space reservation* become essential. Also important is the ability to control data and resource access through local policies and access control lists. For instance, the capability of staging files from tape to disk or to store results permanently on tape should be allowed only to users with certain roles and belonging to specific groups. Data processing managers within each experiment are allowed to check the resources available and ensure correct usage. They need to check for file ownership, correct placement, sizes, etc. They can delete files or move them to storage with appropriate quality of service whenever needed.

4. Storage Requirements

In this section we describe the current state and the continuous evolution of storage services available on the WLCG and EGEE infrastructures.

4.1 The classic SE and SRM v1.1

A grid-enabled storage facility is called a Storage Element (SE). Originally an SE was nothing more than a GridFTP server in front of a set of disks, possibly backed by a tape system. Such a facility is called a Classic SE. It was the first storage service implementation in the WLCG infrastructure. Many tens of Classic SEs are still there today, but they are used by virtual organizations (VOs) other than the LHC experiments. Each supported VO has access to a part of the name space on the SE, typically corresponding to a file system dedicated to the VO. A shared file system with quotas would also work. A large MSS with a tape back-end may be configured such that files in certain subsets of the name space will be flushed to tape and recalled to disk as needed.

There are at least the following issues with the Classic SE:

- It usually is not easy to enlarge the amount of disk space available to a VO indefinitely. Various modern file systems can grow dynamically when new disks are made available through some logical volume manager, but a single file system may become an I/O bottleneck, even when the file system is built out of multiple machines in parallel (e.g. as a SAN or cluster file system). Furthermore, commercial advanced file systems are expensive and may lead to vendor lock-in, while open-source implementations have lacked maturity (this is steadily improving, though). Instead, multiple file systems could be made available to a VO, mounted on different parts of the VO name space, but it usually is impossible to foresee in which parts more space will be needed. A site could grow its storage by setting up multiple GridFTP servers, all with their own file systems, but that may leave some of those servers idle while others are highly loaded. Therefore the desire is for an SE to present itself under a single name, while making transparent use of multiple machines and their independent, standard file systems. This is one of the main reasons for developing the Storage Resource Manager concept.

- GridFTP servers lack advance space reservation: a client will have to try and find out which fraction of its data it actually can upload to a particular server, and look for another server to store the remainder.

- A GridFTP server fronting a tape system has no elegant means to signal that a file needs to be recalled from tape: the client will simply have to remain connected and wait while the recall has not finished. If it disconnects, the server might take that as an indication that the client is no longer interested in the file and that the recall should therefore be cancelled. Furthermore, there is no elegant way to keep a file pinned on disk to prevent untimely cleanup by a garbage collector.

- A GridFTP server has no intrinsic means to replicate hot files for better availability. The host name of a GridFTP service could be a round-robin or load-balanced alias for a set of machines, but then each of them must have access to all the files. This could be implemented by some choice of shared file system, or by having the GridFTP server interact with a management service that will replicate hot files on the fly, making space by removing replicas of unpopular files as needed. Such functionality is naturally implemented by a Storage Resource Manager.

In the spring of 2004 the WLCG middleware releases started including SRM v1.1 client support in data management. The first SRM v1.1 service available on the WLCG infrastructure (to the CMS experiment) was a dCache instance at FNAL, the Tier-1 centre where the dCache SRM is developed. The majority of the Tier-1 centres have since adopted dCache as MSS front-end. In the autumn of 2004 the CASTOR services at CERN and 3 Tier-1 centres also became accessible through SRM v1.1, while retaining a Classic SE appearance in parallel for backward compatibility. In the spring of 2005, to assist in the configuration and operation of Tier-2 sites, the WLCG/EGEE middleware releases started including support for both dCache and Disk Pool Manager (DPM) installations. dCache has had more options for advanced configurations (e.g. separation of read and write pools), but the DPM has been simpler to operate. Early 2008 CASTOR is in use at 7 WLCG sites, dCache at about 60, and the DPM at about 130.
Though the transition to SRM v1.1 has brought significant improvements to WLCG data management, it became clear that it still has important defects:

- SRM v1.1 still lacks advance space reservation. It only allows for an implicit space reservation as the first part of a short-lived store operation. This does allow for the operation to be cancelled cleanly when insufficient space happens to be available, though.

- SRM v1.1 lacks an elegant pre-staging functionality. When a file has to be recalled from tape, the client will either have to remain connected and wait, or it would have to resort to a server-specific protocol for having the file staged in advance.

- There is no portable way to guarantee the removal of files that are no longer wanted. SRM v1.1 only has an advisory delete function, whose effects differ in different implementations. Client tools typically have to recognize the various implementations and invoke server-specific algorithms, contrary to the idea of a protocol standard.

- SRM v1.1 lacks equivalents to basic file system operations e.g. for renaming files, removing directories, or changing permissions. Directories are created implicitly.

As for the Classic SE, files and directories owned by a VO have to be made writable for their whole VO by default, so that any member of the VO can write to the SE without further administrative operations or requiring a strict organization of the VO name space. Exceptions are made for the VO production managers, who are responsible for the vast majority of the data produced by a VO. Usually they ask for dedicated subsets of the VO name space, where only they can write. At the same time they can negotiate the desired quality of service, e.g. dedicated disk pools.

4.2 The Storage Element Service

In the first quarter of 2005 the WLCG Baseline Services working group (BSWG) [8] was established in order to understand the experiment requirements for their data challenges. For each of the experiments a data challenge is a set of large-scale tests focused on verifying the readiness and functionality of its computing infrastructure. The BSWG report [8] includes an assessment of the main functionalities needed from storage services. It established that a Storage Element is a logical entity that provides the following services and interfaces:

- A mass storage system (MSS) that can be provided by either a pool of disk servers or more specialized high-performance disk-based hardware, or a disk cache front-end backed by a tape system.
- A storage interface to provide a common way to access the specific MSS, no matter what the implementation of the MSS is.
- A GridFTP service to provide data transfer in and out of the SE to and from the Grid. This is the essential basic mechanism by which data is imported to and exported from the SE. The implementation of this service must scale to the bandwidth required. Normally, the GridFTP transfer will be invoked indirectly via the File Transfer Service or via the storage interface.
- Local POSIX-like input/output calls providing application access to the data on the SE.
- Authentication, authorization and audit/accounting facilities. The SE should provide and respect ACLs for files and data-sets, with access control based on the use of extended X.509 proxy certificates with a user Distinguished Name (DN) and attributes based on Virtual Organization Membership Service (VOMS) roles and groups. It is essential that an SE provides sufficient information to allow tracing of all activities for an agreed historical period, permitting audit on the activities. It should also provide information and statistics on the use of the storage resources, according to schema and policies.

A site may provide multiple SEs with different qualities of storage. For example, it may be considered convenient to provide an SE for data intended to remain for extended periods and a separate SE for data that is transient – needed only for the lifetime of a job or set of jobs. Large sites with MSS-based SEs may also deploy disk-only SEs for such a purpose or for general use.

Since most applications will not communicate with the storage system directly, but will use higher-level applications such as ROOT [3], it is clear that these applications must also be enabled to work with storage interfaces.

4.3 Beyond the WLCG Baseline Services Working Group

The BSWG required storage services to provide the features of SRM v1.1 along with a subset of SRM v2.1. By the end of 2005, however, it became clear that sufficiently compatible implementations would not be available before the spring of 2006, by which time it was foreseen to start WLCG Service Challenge 4, the last in a series of large-scale tests to assess the WLCG infrastructure readiness for handling LHC data taking and analysis. Therefore, in February 2006 an initiative was agreed to simplify the set of requirements. A new working group was established, including storage system developers and managers, representatives from the experiments, and data management middleware developers. At the end of May the first version of SRM v2.2 was agreed to. At the same time a WLCG-specific Memorandum of Understanding (MoU) [16] spelled out that WLCG client middleware would only exercise a subset of the full functionality. This allowed the storage system developers to ignore features not required by the MoU or to postpone their implementation.

The new set of requirements essentially was the following:

- Only permanent files, i.e. only the user can remove files.
- Advance space reservation without streaming, initially only static, later also dynamic.
- Quotas. Unfortunately not yet accepted as an SRM feature.
- Permission functions with POSIX-like ACLs, for directories and files. It must be possible to match permissions to the contents of the client's proxy credentials, i.e. the distinguished name and/or a set of VOMS groups and roles.
- It must be possible for privileged users, groups and roles to have a better quality of service, e.g. dedicated disk pools, higher priority.
- Basic directory functions: mkdir; rmdir; rename (on the same SE); remove; list (up to a server-dependent maximum number of entries may be returned).
- Data transfer control functions: stage-in and stage-out type functionality; pinning and unpinning; request status monitoring; request cancellation.
- Paths relative to an implicit VO-specific base directory.
- Paths should be orthogonal to quality of service (e.g. retention policy, access latency).
- A method to discover the supported transfer protocols.

4.4 The Storage Classes

In the summer of 2006 the WLCG Storage Classes Working Group was established to understand the requirements of the LHC experiments in terms of ***quality*** of storage (Storage Classes) and how such requirements could be implemented in the various storage solutions available. For instance, this implies understanding how to assign disk pools for LAN or WAN access and trying to devise common configurations for VOs and recipes tailored per site.

The ***Storage Class*** *determines the essential Quality-of-Service properties that a storage system needs to provide for given data.*

The LHC experiments have asked for the availability of combinations of the following storage devices: Tapes (or other reliable storage system always referred to as tape in what follows) and

Disks. A file residing on Tape is said to be in Tape1. A file residing on an experiment-managed disk is said to be in Disk1. Tape0 means that the file does not have a copy stored on a reliable storage system. Disk0 means that the disk where the copy of the file resides is managed by the system: if such a copy is not being used, the system can delete it.
The Storage Classes Working Group decided that only certain combinations (or Storage Classes) are needed for the time being, corresponding to specific choices for the Retention Policy and the Access Latency as defined by SRM v2.2:

- Custodial-Nearline = Tape1Disk0 class.
- Custodial-Online = Tape1Disk1 class
- Replica-Online = Tape0Disk1 class

Tape0Disk0 is not implemented. It is pure scratch space that could be emulated using one of the available classes and removing the data explicitly once done. However, it could be handy for LHC VOs to have such a type of space actually implemented eventually.
In the ***Custodial-Nearline*** storage class data is stored on some reliable secondary storage system (such as a robotic tape or DVD library). Access to data may imply significant latency. In WLCG this means that a copy of the file is on tape (Tape1). When a user accesses a file, the file is recalled in a cache that is managed by the system (Disk0). The file can be "*pinned*" for the time the application needs the file. However, the treatment of a pinned file on a system-managed disk is implementation dependent, some implementations choosing to honour pins and preventing additional requests, others removing unused on-line copies of files to make space for new requests.
In the ***Custodial-Online*** storage class data is always available on disk. A copy of the data resides permanently on tape, DVD or on a high-quality RAID system as well. The space owner (the virtual organization) manages the space available on disk. If no space is available in the disk area for a new file, the file creation operation fails. This storage class guarantees that a file is never removed by the system.
The ***Replica-Online*** storage class is implemented through the use of disk-based solutions not necessarily of high quality. The data resides on disk space managed by the virtual organization.
Through the Storage system interface, it is possible to schedule Storage Class transitions for a list of files. Only the following transitions are allowed in WLCG:

- Tape1Disk1 → Tape1Disk0. On some systems this can be implemented as a metadata operation only, while other systems may require more operations to guarantee such a transition.
- Tape1Disk0 → Tape1Disk1. This transition is implemented with some restrictions: the request will complete successfully but the files will remain on tape. The files will be actually recalled from tape to disk only after an explicit request is executed. This is done in order to avoid that a big set of files is unnecessarily scheduled for staging and therefore to smoothen operations in particular for those Mass Storage Systems that do not have a scheduler.
- Tape0 ↔ Tape1 transitions are not supported at the start of LHC (if ever). For physics validation operations, since the amount of data to transfer to tape after the validation is not big (only 1-2% of total data) a change from Tape0Disk1 to Tape1DiskN can be approximated by copying the files to another part of the name space, specifying Tape1DiskN as the new storage class, and then removing the original entries.

4.5 The Grid Storage Systems Deployment working group

In January 2007 the Grid Storage Systems Deployment (GSSD) [9][13][14][15] working group was established with the following main goals:

- Testing of SRM v2.2 implementations for compliance and interoperability.
- Establishing a migration plan from SRM v1.1 to SRM v2.2 so that the experiments can access the same data through the 2 protocols transparently.
- Coordinating with sites, experiments, and developers the deployment of the various SRM v2.2 implementations and the corresponding Storage Classes.
- Coordinating the definition and deployment of an information schema for the Storage Element to allow for the relevant aspects of SRM v2.2 services to be discoverable through the WLCG information system.
- Coordinating the provision of the necessary information by the storage providers in order to monitor the status and usage of storage resources.

It took until the autumn of 2007 before CERN and the Tier-1 sites started putting SRM v2.2 services into the WLCG production infrastructure, with known deficiencies to be corrected in later versions, in particular when significant operational experience has been gained during a final set of large-scale tests, the Common Computing Readiness Challenge foreseen for February and May 2008. The work of the GSSD is expected to carry on throughout 2008 and probably further.

5. Beyond WLCG: data management use cases in EGEE

The data management use cases of the many other disciplines for which the EGEE infrastructure is intended have a lot in common with those of the LHC experiments, which have been the main driving force behind the development of the various data management services and client utilities available today. Other disciplines also pose additional requirements that matter less to the LHC experiments, at least for the time being:

- Fully encrypted communication and storage for much improved privacy. For example, in biomedical research it is vital that no data be accidentally exposed, whereas LHC experiments do not mind if the vast majority of their data happens to be world-readable. They do not want the high-level analysis results to be exposed, but such results typically can be stored and processed locally without the need for grid services. On the other hand, encrypting and decrypting their huge volumes of low-level data on the fly probably would be an unacceptable overhead still for a long time. For the biomedical community encrypted storage has been developed as a plug-in for the DPM Storage Element and the GFAL client suite. An additional necessary component is the Hydra [17] distributed key service.

- Standard POSIX access to data served by Storage Elements. Most of the EGEE communities have legacy applications that expect to read and write files through standard POSIX I/O instead of GFAL, ROOT, XROOTD, or server-dependent protocols like RFIO [10] and DCAP [11]. It was fairly straightforward for a Classic SE to make its data available to a local batch farm via NFS, but this has not been possible for SRM services. An SRM should be able to replicate hot files and direct clients to any disk server in a load-balanced set. It should also be able to clean up a replica without the risk that a client might still be reading it unbeknownst. These problems can be overcome, though. The main objective for the StoRM implementation of SRM v2.2 is to allow secure POSIX access by legacy applications, typically through a cluster file system like GPFS and with just-in-time access control lists, so that a file is only accessible for a duration that a client has to negotiate with the SRM in advance. All of the SRM implementations in use

on the EGEE infrastructure have expressed interest in developing NFSv4 interfaces, which would allow for standard POSIX access by clients without reducing vital server functionality. Finally, a transparent remote I/O library like Parrot [18] may be enhanced with plug-ins for some of the popular protocols that can be served. This will not help statically linked applications, though.

- Availability of client utilities on many platforms. To optimize the use of available resources the high-energy physics experiments and laboratories have moved their computing infrastructures almost exclusively toward a few flavours of Linux distributions. Many other disciplines, however, find themselves with applications that will only run on platforms of other types. NFSv4 would help in this regard as well, but SRM clients and other data management utilities would still have to be ported to the most popular other platforms.

REFERENCES

[1] J.-P. Baud, J. Casey *Evolution of LCG-2 Data Management* CHEP, La Jolla, California, March 2004.
[2] T. Barrass, D. Newbold, L. Tuura, The CMS PhEDEx System: a Novel Approach to Robust Grid Data Distribution, AHM2005 19 - 22nd September 2005 Nottingham (UK)
[3] C. Boeheim, A. Hanushevsky, D. Leith, R. Melen, R. Mount, T. Pulliam, B. Weeks *Scalla: Scalable Cluster Architecture for Low Latency Access Using xrootd and olbd Servers* 22 August 2006, http://xrootd.slac.stanford.edu/
[4] Alice Collaboration *Alice Technical Design Report of the Computing* CERN-LHCC-2005-018, ALICE, TDR-012, 15 June 2005: http://aliceinfo.cern.ch/static/Documents/TDR/Computing/All/alice_computing.pdf
[5] ATLAS Collaboration *ATLAS Computing Technical Design Report* CERN-LHCC-2005-017, ATLAS, TDR-017, 20 June 2005: http://cern.ch/atlas-proj-computing-tdr/PDF/Computing-TDR-final-June20.pdf
[6] CMS Collaboration *CMS Computing Technical Design Report* CERN-LHCC-2005-023, CMS, TDR-007 http://cdsweb.cern.ch/search?id=838359
[7] LHCb Collaboration *LHCb Computing Technical Design Report* CERN-LHCC-2005-019, LHCb, TDR-019, 20 June 2005: http://doc.cern.ch/archive/electronic/cern/preprints/lhcc/public/lhcc-2005-019.pdf
[8] The WLCG Baseline Service Working Group Report v1.0, 24 June 2005 http://lcg.web.cern.ch/LCG/peb/bs/BSReport-v1.0.pdf
[9] The WLCG SRM Development Group: http://cern.ch/LCG/SRMdev
[10] CASTOR: CERN Advanced STORage manager - http://castor.web.cern.ch/castor
[11] M. Ernst, P. Fuhrmann, T. Mkrtchyan, J. Bakken, I. Fisk, T. Perelmutov, D. Petravick, *Managed data storage and data access services for Data Grids* CHEP, La Jolla, California, March 2004; http://www.dcache.org
[12] F. Donno, Storage Management and Access in WLHC Computing Grid, Ph.D. Thesis, University of Pisa, May 2006, http://etd.adm.unipi.it/theses/available/etd-07122007-101513/
[13] The WLCG Grid Storage System Deployment Working Group: https://twiki.cern.ch/twiki/bin/view/LCG/GSSD
[14] L. Abadie et al., Storage Resource Managers: Recent International Experience on Requirements and Multiple Co-Operating Implementations, Mass Storage Systems and Technologies, September 24-27, 2007, San Diego, California, USA
[15] F. Donno et al., Storage Resource Manager version 2.2: design, implementation, and testing experience, International Conference in Computing in High Energy and Nuclear Physics, 2-7 September 2007, Victoria BC, Canada
[16] SRM v2.2 MoU: http://cd-docdb.fnal.gov/0015/001583/001/SRMLCG-MoU-day2%5B1%5D.pdf
[17] gLite Hydra Keystore: https://edms.cern.ch/file/697751/1.0/EGEE-TECH-697751-v1.0.pdf
[18] Parrot: http://www.cse.nd.edu/~ccl/software/parrot/
[19] WLCG: http://lcg.web.cern.ch/LCG/
[20] EGEE: http://www.eu-egee.org/
[21] Open Science Grid: http://opensciencegrid.org/
[22] Nordic Data Grid Facility: http://ndgf.org/ndgfweb/home.html
[23] DPM: https://twiki.cern.ch/twiki/bin/view/LCG/DataManagementDocumentation
[24] StoRM: http://storm.forge.cnaf.infn.it/doku.php
[25] BeStMan: http://datagrid.lbl.gov/bestman/