Common Software Activities in Data Organisation, Management, and Access (DOMA)

Authors: Marian Babik, Martin Barisits, Andrea Ceccanti, Michael Davis, Alastair Dewhurst, Alessandro Di Girolamo, Andrew B. Hanushevsky, Oliver Keeble, Mario Lassnig, James R. Letts, Edoardo Martelli, Shawn McKee, Tigran Mkrtchyan, Mihai Patrascoiu, Andreas J. Peters, Elvin A. Sindrilaru, and Christoph Wissing

Editors: Oliver Keeble and Mario Lassnig

September 30, 2021

Contents

1 Introduction

This document represents the Data Organisation, Management and Access ("DOMA") contribution to the HL-LHC Computing Review Stage 2: Common Software Activities.

The document describes the four main areas requested in the charge. It is structured so as to the expose the relevant common themes and strategic directions before documenting considerations particular to each project.

The first section covers the planning context, our understanding of what Run 4 computing means, major themes and looks at risks and mitigations. Next, the two common software activities *Rucio* and *FTS* are discussed, including their critical dependencies GFAL and Davix. The common storage activities are then presented. These sections use the same structure for each common activity: a general description, an overview of the project management and the team, the future plans and the associated risks. The future plan for the network technologies is then presented.

2 Planning for HL-LHC

2.1 The planning context

The community is following a guided, iterative approach to reaching a Run 4 offline computing platform. This is predicated on a continual process built on dialogue between all stakeholders, including the experiments and Common Software Activities (CSAs). This process allows the ever-improving understanding of requirements from the experiments, as well as the boundary conditions set forth by the CSAs and infrastructure, to be reflected incrementally in features and scalability. Even though the requirements and boundary conditions are very dynamic, and sometimes disrupted by outside-HEP innovations, this continuous discussion has proved highly effective on the road to Run 3.

The consequence of this situation is that the planning horizon for concrete activities within participating projects is relatively short. What must be emphasised is that there is an existing process guiding the work and ensuring that all parties are in agreement. In particular, themes touching multiple experiments and CSAs, such as those followed by the current DOMA activities, can thus be quickly identified and in turn tracked effectively.

This flexible, iterative approach adopted also means that the current focus of the computing development objectives is geared towards Run 3 and not Run 4. An estimation of Run 4 needs and requirements, to which projects could already start evolving, is constrained based on many unknowns of the physics community, including detector hardware and even the physics program itself. Consequently this document cannot present a checklist or detailed milestones against which project plans can be compared or steered towards. We therefore describe the current directions adopted by the relevant projects, understanding that they are in consultation with the relevant experiments already, and characterise what can be said with some confidence about Run 4 computing. We highlight where requirements are most in need of clarification in order to direct the available development resources of the CSAs, and describe where extensions are needed.

2.2 The Run 4 environment and its requirements

Any attempt to characterise the Run 4 DOMA environment is inevitably preliminary. The editing process for this document involved consultation with the experiments from which the following, incomplete, picture emerged:

- An order of magnitude increase in data volume can be reflected in an increase in file size, number of files, or both, with the exact position adopted between the two extremes being driven by a number of factors. While ATLAS and CMS could both imagine 100GB files as a reasonable reference, these scales needs to be verified and validated with the common software activities. It is clear that while some aspects of the frameworks, for example metadata management, will drive the solution towards larger files, other domains, for example data movement, will drive the choice in the opposite direction. It is therefore crucial to consider this from both the highest level, e.g., computational workflows or data centre procurement, to the technical detail level, e.g., database scalability or scheduling algorithms.

- CMS is actively embracing the remote data access paradigm, while ATLAS has a direct copy approach (remaining open to increase remote data access in the future). These different modes of operation generate different requirements on infrastructure and services. The presence of caches, predominantly XCache, is anticipated for this endeavour.

- Activities focused on reducing costs are ever more important, however it is difficult to express these savings in terms of currency, as this varies wildly even within single countries and are sometimes not even publicly exposed. Therefore, activities related to quantifying these savings will have to do it in terms of relative advantage, e.g., a 20% increase in storage capacity due to a particular QoS mechanism being used. Technically, there will be maximum exploitation of tape and the possibilities offered more generally by QoS, e.g., erasure coding data. Potential savings through self-managing storage will have to be assessed.
- Both ATLAS and CMS are expecting to rely on Rucio and FTS, with FTS used in addition by LHCb. Ensuring that the experiments' Run 4 requirements on these projects are compatible is important and requires a sustainable development and operations effort. FTS in particular is viewed as a privileged interface with the network layer and development directions need to be established at the WLCG level.
- The experiments will continue to make opportunistic use of HPC resources.
- The administrative and operational overhead for a site of having to support a variety of experiment workflows can become inhibiting. We thus see the potential for a cross-experiment effort with regards to harmonising the computing models and adopting shared data management operations with respect to data lifecycle management, space management, and monitoring.

Certain questions of interest to the common software areas are not answerable today. These include things such as what interfaces will be preferred for what operations, and what access patterns and concurrency can be expected. On such topics, it is important to converge on a coherent set of requirements for the CSAs. WLCG, as the central coordinating body, has a crucial role in initiating the relevant discussions and providing a platform for forging consensus. As the example of analysis access patterns illustrates, systems cannot be optimised for all possible use cases.

2.3 Project Run 4 planning, evolution, and R&D

All projects are aware that Run 4 will place considerable pressure on their scalability and are typically addressing their own internal interpretations of what a factor 10 in scalability means. For example, FTS and Rucio are concerned about database scalability, roughly a consequence of increasing file numbers and interaction rates. Other areas are more concerned with volume and throughput such as the storage systems and the network activities. Latency and access performance may come under greater pressure with an increase of interactive WAN-based data access. Other areas are concerned with concurrency and access patterns. We can thus observe that in absence of an explicit signal from the community about scaling requirements, projects are independently identifying their bottlenecks and working on them as part of the standard development cycle.

This reveals the advantage of elevating some themes to the community level where a consensus can be iterated, as has already observed in the success of activities such as the DOMA TPC working group, or where effort can be contributed from multiple sources. The recognition that file sizes are likely to increase is a good example of something which could now be tackled at the community level rather than in isolation and for which concrete R&D could already begin. On the other hand, it will become important to verify if these independent efforts are not creating problems in other parts of the distributed computing system. For example, larger files will ease pressure on certain, typically metadata oriented, parts of the system while potentially causing new issues in data distribution or use of tape systems. As such, the efficacy of independent developments needs to be carefully followed, communicated, and reviewed.

To steer development one must also anticipate the technical basis on which our systems will be deployed. This runs from market analysis to establish a likely hardware platform, with availability of various storage technologies, networking components, data-centre architectures and the like, up through opportunities that will be offered by the operating systems of the day, e.g., regarding asynchronous I/O. WLCG has various mechanisms for tracking this evolution.

2.4 Data management directions

This section presents an overview of a number of currently open strategic directions of interest to the community. It is possible that during Run 3 new directions will be opened, or existing ones consolidated or withdrawn.

Data access via distributed caches Data access via caches is an important part of computing models which foresee consolidation of storage into a smaller number of larger sites. Such caches can also serve to accelerate access to opportunistic CPU, for example in the cloud, and can act as a latency hider in the context of random, sparse access to files.

The dominant technology here is clearly XCache, part of the XRootD project, which offers a block-based, read-only caching service. Other storage projects participate as origin servers.

Despite various investigations, the rich selection of third party projects offering HTTP caching have yet to be exploited by the community. While they all have some technical limitations when deployed for this use case, WLCG's preference for the xroot protocol for file access must also be a factor. XCache alone will therefore likely remain the dominant caching technology in the runup to Run 4.

A *Virtual Placement* (VP) prototype for distributed caches, based on on XCache, has been developed and deployed on all US ATLAS sites and two EU ATLAS sites. Due to sheer scale and the fact that analysis workflows almost always involve one or more infrequent derivation steps, the majority of HEP data is read rarely and would be considered "cold" in caching scenarios. This fact, coupled with job scheduling systems that prefer to schedule tasks in a manner that colocates workloads with data, makes naïve deployments of caches at best marginally effective. The virtual dataset replicas satisfy the job system's preference for colocating jobs with data, which allows jobs to start as soon as the first data block arrives. Future development will see VP completely integrated in Rucio thus making it available for all the experiments using it.

Third party copy With the end of support for the Globus implementation of GridFTP the community has made major steps towards a data distribution infrastructure based on both HTTP and the xroot protocol. This work was undertaken within the DOMA TPC WG and cuts across the stack discussed in this document, involving Rucio, FTS, GFAL, Davix, and the storage systems.

Given the maturity of the HTTP TPC implementations there is no obvious source for concern on Run 4 timescales. They will have to be validated at Run 4 data rates and certain optimisations may be required, for example resumption of failed transfers. There is an ongoing activity for Data Challenges to ensure that such requirements are tracked, and will continue throughout Run 3 with increasing intensity. The extension of these implementations to support token-based AAI is also well advanced. Looking further ahead, the emergence of new data transfer protocols could even be supported by retaining HTTP for the control channel while modifying the data channel.

Regarding xroot TPC, significant progress was also made in the TPC WG, although implementation coverage is not the same as HTTP, e.g., StoRM has no plans to support it.

Exploitation of tape Tier 1 centres operate tape storage systems whose principal mission is the archival of LHC data. Unlike many other archival solutions which tend toward the "write once, read never" style of operation, WLCG tape systems are the subject of continual I/O optimisation in the hope of supporting more diverse workflows and relieving capacity pressure on pure disk systems.

A tape system is typically a separate component sitting behind either dCache, EOS or StoRM. It can, in many cases, be further separated into an integration layer and a lower tape management layer, e.g., the StoRM/GEMSS/SpectrumProtect stack illustrates this. Only certain aspects of this stack represent common software activities from the community, in particular CTA and Enstore represent complete low-level tape solutions and ENDIT is used as an integration layer at more than one site.

The Run 4 outlook indicates the possibility of a partial consolidation on CTA as a tape backend, initially behind EOS and subsequently behind dCache. The outcome of various feasibility and prototyping projects in the next few years will clarify to what extent this represents a community direction. If such a direction does emerge, CTA would represent a community asset which could be the subject of cumulative investment by the concerned sites. While CERN has committed to support a CTA community the technology choice will always remain with the sites. Some, typically serving multiple science domains, are unlikely to abandon their existing investment in IBM solutions such as Spectrum Protect or HPSS.

One aspect of tape systems not addressed by the tape backend is the interface offered to the experimental frameworks. Tape systems need to offer some specific functions, such as bring-online or buffer eviction (unpinning). These were offered historically by the SRM protocol, although recently CTA has implemented them using the xroot protocol. Community efforts to create a common, HTTP based interface are underway and progressing well and should protect the community from both protocol proliferation and the obsolescence of SRM.

All concerned projects have reported various plans related to optimisation of tape use. In many cases, this has been triggered by extensive testing as part of the ATLAS “data carousel” mechanism. As these optimisations are often predicated on knowledge of experiment workflows and iterative test/update cycles have proven effective, the maintenance of this pressure throughout Run 3 is likely to represent the best path to reaching Run 4 performance.

It is possible that some of today’s tape systems will be displaced by other implementations of archival storage, such as the disk based system at KISTI. Commercial cloud offerings also exist, accessible through a different set of APIs than those used in WLCG.

QoS and cost savings The QoS concept represents a key method of storage cost containment for Run 4. It is based on the idea of breaking the current, simple DISK/TAPE dichotomy in WLCG storage and of recognising a more diverse set of storage types, e.g., fast SSDs, volatile out-of-warranty disks, object storage or archival disk.

The set of storage types of interest, otherwise known as *QoS Classes*, is something that should be decided at the WLCG level, to allow sites to advertise their capabilities to experiments using a common vocabulary. An agreement on the definition of this list will be a key enabler for QoS in Run 4. A common set of classes would allow this finer classification granularity to propagate throughout the system, even as far as resource pledges and communication with funding agencies. Due to the unfeasibility of expressing QoS savings in terms of currency, alternative methods of expressing savings must be explored.

All existing systems are able to statically configure spaces based on different media types which would represent a particular QoS. The expected market evolution in media, in particular the flattening of the HDD capacity/cost curve and the diminishing cost difference with respect to SSDs, suggest that there will be a natural shift in this area which will need to be accommodated.

Different policies regarding data replication or erasure coding can be used to explore different performance/reliability/cost permutations offered by the system. This can be offered natively by the storage system (e.g. EOS or XRootD) or delegated to a lower layer (e.g. Ceph).

The simplest path to exploiting the QoS concept is to allow storage systems to configure different areas or endpoints with different Qualities of Service. Support would then be required only at the experiment orchestration layer, to recognise and populate these areas appropriately, and beyond, in workflow management which would have to broker compatible jobs.

A more sophisticated model, requiring support at the storage system layer, would be implementation of a data lifecyle which would migrate data between QoS classes automatically, based on some policy. Some systems, for example EOS, already have this capability. Experiment frameworks could be notified when QoS transitions take place.

Perhaps the most advanced possibility is the provision of an interface which would allow the experiment frameworks to trigger QoS transitions, as they do today for DISK and TAPE. This would involve community agreement on an interface, which would likely have to be created by the community, for managing such transitions, and its common implementation across storage systems. As such, this represents a considerably more ambitious approach and currently no storage systems report plans in this direction.

Token support When data distribution emerged as a natural pathfinder use case for the token AAI, the DOMA TPC WG took on the coordination responsibility and oversaw implementation in the relevant shared software activities. Considerable progress was made and all projects report support for tokens, with some issues outstanding. This affected Rucio, FTS, GFAL, Davix, and the storage systems.

There is a large intersection with other common projects such as IAM and other working groups. Finalisation of support depends on interaction between these activities and on various usage models being defined and precise policies being described and adopted, for example regarding the content of the tokens and regarding specific workflows and trust relationships. The existing WLCG mechanisms should to continue to iterate as the use cases reach production.

Storage diversity (HPC / Cloud) The common software activities are not reporting concrete plans related to cloud or HPC. This is in part because the topic covers a wealth of use cases, some of which are already naturally supported. For example, the simple fact of deploying these services on a public cloud has been demonstrated a number of times. Furthermore, no explicit requirements regarding cloud were communicated by the experiments.

For HPC the need to lower the barrier to using these resources is well accepted, alongside the requirements that this will place on networks. It is less clear how the issue affects the projects under consideration in this report. They could, for example, function as edge services, allowing easy exchange with the existing grid, in contrast to the likely evolution of the Globus servers commonly deployed. They could also function internally to the HPC cluster, as described in the XRootD entry. As with cloud resources, the community is still in the process of dialogue with the centres and evolving the requirements. A central "mechanism" is being put in place to liaise with PRACE, EuroHPC and similar entities. In addition, the presence of technical contacts at the various HPC centres has been deemed necessary in the commissioning phases of the ongoing HPC activities.

Data management and the network Certain advances anticipated to bring benefits for Run 4 are predicated on increased coupling between data management systems (inc storage) and the network layer. One can cite possible provisioning responsibilities for Rucio or FTS, and packet-marking by the storage systems. This new collaborative front, and future technical dependency, should be reflected in the structure of WLCG's R&D activities.

2.5 DOMA coordination in WLCG

Much of the R&D work has been driven through the coordination of the DOMA activity within WLCG, as well as independent work by the experiments and facilities. When seen as a whole, the DOMA activities are now entering Phase II, where its approach has been modified accordingly. The major idea is to support the many ongoing activities better through a flexible coordination framework which can deal with the overlapping nature of the work. This means that activities are not followed by a *static* working group but instead are embedded into an overall DOMA R&D timeline towards both Run 3 (commissioning of prototype services for the experiments and facilities) and Run 4 (scalability and performance). In the context of this restructuring a full DOMA activity survey is currently being conducted, which will serve as a *living document* for Run 4 R&D planning.

2.6 Run 3

The increase in scale of ALICE and LHCb during Run 3 will bring many opportunities to observe the scaling of the infrastructure alongside the implementation of various functional enhancements which are expected to underpin the Run 4 computing (e.g. non-GridFTP TPC, tokens). As no projects reported having ringfenced HL-LHC R&D effort, the practical consequence is that development directions of the CSAs are dictated by Run 3 requirements.

2.7 Risk analysis and mitigation

2.7.1 Technical evolution in WLCG

In pursuing its mission to provide the offline computing platform for the LHC experiments, WLCG hosts numerous forums intended to facilitate communication between experiments, sites and technology providers. These are the foundation of the iterative process which guides technical evolution of the infrastructure and drives a technological lifecycle from the conception of ideas through to implementation, deployment and even, in some cases, retirement. The DOMA activity is a key element in this, guiding data-oriented development.

In the ideal scenario, these bodies would function perfectly, delivering accurate, achievable and timely requirements to healthy software projects capable of producing the implementations in time for deployment and testing before Run 4 starts. The principal risks faced by the community are related to failures of various aspects of this cycle, or the assumptions it is based on.

This section covers the main risk scenarios, their impact and what mitigations are possible. It complements the corresponding section in the project-specific sections.

2.7.2 Requirement management

One set of risks is related to requirement management. One can imagine innumerable ways in which the current processes can fail - requirements may be incompatible with one another or otherwise unachievable, they may be delivered late, they may misunderstood, they may be incomplete or they may even be wrong and not represent real needs. All of these, if not addressed, would result in a compromised infrastructure, ultimately affecting the physics output.

The mitigations here revolve around exploiting WLCG's communication channels to ensure the validity of the requirements that CSAs are working towards. As the process is iterative and change is gradual, these checks also need to be continuous (or at least periodic).

2.7.3 Dependencies

The CSAs, like all software projects, depend on a diverse set of third party technology providers. They also exhibit many internal dependencies, for example note the number of projects relying on the XRootD framework. Such dependencies represent risks in that they may fail to deliver as expected and projects may even cease. A certain number of these dependencies are relevant to the present document.

Databases Projects have reported concerns about the scaling of relational databases under certain possible Run 4 scenarios. The risks depend heavily on the Run 4 computing models, often scaling with number of objects tracked (typically files) or metadata operations expected. The mitigation here is ensuring that the key issues are identified so they can be clarified as early as possible. For example, early clarification of expected file sizes would give more time for scalability issues to be anticipated and addressed.

Ceph Ceph is an open-source software storage platform that is frequently used at sites and can act as a lower tier in the storage management stack, behind many of the storage services discussed here. It provides 3-in-1 interfaces for object-, block- and file-level storage. Its ubiquity means that disruptions to this project would have repercussions throughout the infrastructure.

The mitigation in this case is maintaining the community's influence. While Ceph is owned by Redhat, there is a Ceph foundation which is organised as a directed fund under the Linux Foundation. Both CERN and STFC are associate members of the Ceph foundation. Given Ceph's widespread adoption and the infrastructure's increasing reliance on it, WLCG's integration into the Ceph community is an important asset and helps to maximise our influence.

Tape One hardware dependency is worth highlighting. Tape is crucial to Run 4 storage planning but the market is now highly consolidated around a small number of vendors. The continued health of this market must be monitored as its cessation would jeopardise the data volume and cost targets and invalidate a major assumption underpinning today's workflows. Exploration of alternatives, such as archival disk, would mitigate this risk.

Internal dependencies A number of projects depend on other libraries or services from within the community (for example regarding security) and there are dependencies between the CSAs treated in this document. Cessations of support would divert resources from dependant projects with the severity of the impact varying greatly depending on the scenario considered. The mitigation in this case is the mobilisation of a community response to either take over maintenance or plan a migration. A particularly notable example was the end of support for the open source Globus distribution which resulted in a managed community migration away from GridFTP.

2.7.4 Community and sustainability

The final category of risk relates to the sustainability of the CSAs and the communities that support them.

Various concerns were identified regarding the ability of the projects themselves to scale with user base expansion. The "factor of 10 at flat funding" rule applies equally to projects themselves which will be faced with expanding requirements and communities. CSAs manage requirements generated by communities beyond LHC experiments, sometimes of considerable size, and in some cases experiments expressed concerns over their influence on the CSAs. While serving diverse communities has a very positive effect for the long-term sustainability and importance of the CSAs, sufficient resources need to be devoted to the needs of LHC computing.

Direct experiment contributions are considered fundamental for the CSAs, however available funding for personpower can make this difficult. New mechanisms, incentives, and recognition for contributions to the CSAs need to be investigated, possibly at the WLCG or institute levels, for sustainable development of the LHC experiment's required functionalities.

WLCG uses a variety of different storage solutions and this diversity comes at a cost, even if there is considerable internal collaboration and reuse. A consolidation of effort between these projects is not practical so the major risk is that of incompatibility and the associated complexity of rolling out across the infrastructure any change which affects all projects. This risk can be significantly mitigated by the community's focus on shared interfaces.

Finally, we observe that even if the CSAs produce the required solutions on time, some work remains on the experiment side to integrate and fully exploit new capabilities or scales. This must be factored into any planning.

2.7.5 Networks

Some of the networking models under consideration for Run 4 anticipate not only an increased network awareness in the CSAs but in some cases active participation in capacity provisioning and QoS management. As always, clear and early agreement on the relevant model and consequent requirements on the CSAs will be crucial to success. Further details regarding risks pertaining to successful exploitation of networks are detailed in section 7.3.

3 Rucio

3.1 Activity description

Rucio is a software framework that provides functionality to organise, manage, and access large volumes of scientific data using customisable policies. The data can be spread across globally distributed locations and across heterogeneous data centres, uniting different storage and network technologies as a single federated entity. Rucio offers advanced features such as distributed data recovery or adaptive replication, and is highly scalable, modular, and extensible. Rucio has been originally developed to meet the requirements of the high-energy physics experiment ATLAS, and is continuously extended to support LHC experiments and other diverse scientific communities.

Rucio is based on a distributed system architecture and can be sectioned into four major system layers: Clients, Server, Daemons, and Resources/Middleware. The clients are different user-facing components, such as Command Line Clients, Python APIs, etc. which enable users and external applications to easily interact with Rucio. The server(s) are load balanced httpd servers processing HTTP-REST requests from the clients. Most workloads, such as deletion requests, transfer requests, consistency checking, etc. are however processed asynchronously by the daemon layer. These are horizontally scalable agents which execute the queued workloads asynchronously. The resource/middleware layer includes the relational database, which is used to store the catalogue and is also used for all transactional state-keeping. Several external middleware systems, such as FTS3, which Rucio utilises to execute, optimise and monitor transfer requests, ActiveMQ for message queues, as well as CRIC as a central grid information catalogue.

Rucio offers a variety of options to monitor the system. Internal monitoring, monitoring of data flows as well as report generation is offered as a common stack which can be utilised by any Rucio community. For the monitoring needs of ATLAS and CMS, this monitoring workflow is further augmented by the CERN IT monitoring infrastructure.

3.2 Management and communication

Rucio is an open-source, common, software project and relies on the contributions of developers coming from the wider Rucio community. Principal stakeholders are the LHC experiments ATLAS and CMS, as well as the CERN associated experiment AMS. Additionally scientific communities such as Belle II, DUNE, LIGO/VIRGO, SKA, Xenon1T and others are also stakeholders. Organisations hosting Rucio deployments for their local use-cases (CERN, BNL, Fermilab, RAL), and scientific projects such as ESCAPE are principal stakeholders.

Communication is direct via the Rucio Slack channels, mailing lists, weekly public Rucio meetings where development as well as operational issues are discussed, as well as a community workshop hosted once a year. Additionally, for each feature release (three times per year) a development roadmap is discussed and agreed on between the development teams and the different community stakeholders defining the short/medium term goals of the project.

The project management is done by the Rucio project lead coordinating the development priorities and developments of the Rucio community.

3.3 Team status and outlook

Rucio is fully organised as an open-source software project, integrating contributions from the wider Rucio developers community. However, these contributions are mostly focused on smaller feature developments to cover needs/use cases from these communities, the actual ongoing evolution, optimisation and especially maintenance of the system is almost entirely done by the ATLAS Rucio team (the core team). This is also reflected in the component leadership, where different developers take formal responsibility for the maintenance and evolution of the different functional components of Rucio. With few exceptions, this also relies on ATLAS developers.

3.4 Development plans

Scalability While Rucio servers and daemons are designed to scale horizontally, database scalability remains a major focus for future developments. The integration of features and workflows offered by the utilised database systems (Oracle, PostgresQL, etc.), such as temporary tables, JSON-type columns and indexing, to optimise the resource utilisation of the database queries, will be an ongoing activity over the coming years.

Quality of Service The development of Quality of Service, to offer experiments a way to orchestrate the resource needs of their computational workflows to best match the storage resources provided by sites, will be a major development, involving the entire associated software stack (Rucio, FTS, storage). Due to the involvement of many systems and the wide range of use-cases of QoS, appropriate testing will have to be foreseen.

Authentication and Authorisation Infrastructure The evolution of the already existing OIDC token support within Rucio. While Rucio includes OIDC support, further development will be needed based on the outcome of the integration testing of the DOMA TPC working group. Adaption and compliance to the WLCG JWT token profile also has to be implemented. Full support for OIDC/OAuth2 tokens is also a major requirement by non-LHC Rucio communities.

Non-LHC developments Several developments are planned and carried out by non-LHC communities. While some of these development needs are shared by several communities, some are specific to a subset of sciences, such as the astronomy sector. While there are no conceptual conflicts with these developments, the complexity increase does put additional load on the development teams.

Metadata The expansion of metadata functionality aims to enable communities to manage their entire metadata catalogues within Rucio. This includes the handling of more complex and compound metadata queries.

Permission model The concept of data embargoes, where data access is limited to a small set of users, for a specific amount of time, is prevalent in some scientific communities. The aim is to support and integrate this functionality both on Rucio as well as on storage level.

Integration Further integration of additional transfer systems next to FTS and Globus and to evolve the support of hosting multiple VOs on one Rucio instance (Multi-VO) are also developments foreseen for the coming years. Supporting the integration work for additional workflow management systems next to PanDA and DIRAC might also require additional developments, e.g., providing new APIs.

3.5 Risk analysis and mitigation

Personpower Due to the on-boarding of new communities and their workflows, the resulting complexity increase of the software and the inflation of support needs an additional strain is put on the core development team, which is already in a difficult personnel situation. This needs to be addressed by finding suitable funding models and structures, to fund and integrate developers, from different communities, in the core development and support of the project. Based on the current known development plans, the project will require an additional 2 expert FTE over the next years. This number has to be revised once more concrete requirements are finalised by the experiments.

Database scalability Rucio heavily relies on relational database systems for the catalogue and internal state-keeping. Both ATLAS and CMS use dedicated Oracle databases at CERN. It needs to be ensured that the relational databases, and their provisioning, can support the Rucio workloads (especially table sizes and transactional rates) at Run 4 scale. Consideration needs to be given to the number of files stored, their access frequency as well as access patterns, which are the primary drivers in terms of scalability. Due to the support of multiple database systems, possible mitigation strategies include the utilisation of the backend best supporting the workload. Moving some of the workload, such as non-transactional data, to non-relational databases is also an option. If database scalability is not reached, Rucio might not be able to interact with third parties (WFMS, production system, users) at the required frequency, limiting the scale of these systems, as well as not be able to delete or orchestrate transfers at the required scale.

FTS Scalability As Rucio orchestrates all data movement through FTS, the scalability of the software is essential for Rucio. While partitioning of the workload to different FTS instances is, and could be, done, this mode of operation creates additional issues in terms of network optimisation and data inconsistencies. If FTS scalability is not reached, Rucio might not be able to orchestrate transfers at the required scale.

4 File Transfer Service (FTS)

4.1 Activity description

The File Transfer Service (FTS) is a low-level data management service, responsible for scheduling reliable bulk transfer of files from one site to another, while allowing control of the network resource usage. FTS provides simplicity by allowing easy user interaction for submitting transfers, via CLI clients, REST API or a web-based portal (WebFTS). It also provides a real-time monitoring platform that is rich in content and a Web Admin interface for service configuration, such as access rights and transfer limits on individual storage elements or transfer links. The service is currently responsible for distributing the majority of the LHC data across the entire WLCG infrastructure and it is fully integrated with experiment frameworks such as Rucio and DIRAC.

The service's reliability is assured by various features such as data integrity checks, in the form of checksum comparisons, a robust transfer retry mechanism, multi-protocol support (WebDAV/HTTPS, GridFTP, xroot, SRM), support for various storage systems (EOS, DPM, Object Storage, STORM, dCache, CASTOR and CTA), support for tape storage technologies via bring-online and archive monitoring operations, diversity on the ways that clients can access the service and generally a flexible and scalable design. One of the biggest advantages of FTS is its ability to run in a *no-configuration* mode: the Scheduler and Optimiser components will maximise the number of parallel transfers for a given transfer link, the goal being to maximise throughput without saturating the storage endpoints. Fine grained scheduling rules can be defined via intelligent priorities, achieved via Activity shares and VO shares. Lastly, FTS can publish transfer messages to an ActiveMQ instance. This allows monitoring systems to have a very detailed view about FTS activity throughout the grid.

FTS has had an active involvement with the DOMA Working Groups in the past, which has given way to the following developments, some of which were funded or contributed by external projects:

- HTTP-TPC support in Gfal2 and Davix
- CDMI-QoS support in FTS
- Storage Element-issued token support in FTS and Gfal2
- OIDC token support in FTS and Gfal2

The service continues its active involvement with the DOMA Working Groups.

4.2 Management and communication

FTS is an Open-Source project, distributed under the Apache 2.0 License. The main development effort comes from the FTS team (CERN IT-Storage group). Throughout past and present times, the project has received external contributions, either sporadic or in the form of longer-term partnerships. A strong emphasis is given to the Open-Source nature of the project and a call for future contributions is presented.

Announcements are done via an FTS3 mailing list, where new releases, upcoming changes and deployment plans are advertised. The mailing list can be used to arrange more in-depth forums to discuss and contribute to the future direction of the project.

Communication with the FTS team is mostly done via the FTS developers mailing or the FTS Mattermost public channel. For support enquiries, either the FTS support mailing list should used, or ticket systems such as GGUS or CERN ServiceNow. Weekly developer meetings are held within the team to discuss fixes, features and deployment plans. When appropriate, a recurrent meeting setup is done involving an external contributor.

The main stakeholders of the project are the experiment frameworks such as Rucio, DIRAC and ASO, small-and-medium experiments (e.g.: NA62, CAST) and scientific communities such as EGI, all of which use FTS extensively.

4.3 Team status and outlook

The Storage Group in CERN-IT is home to a number of the common software projects discussed in this document. It hosts the EOS, CTA, FTS, GFAL and Davix projects, is home to the CERN contributions to the XRootD project (XrdCl in particular), and supports a wide range of related services, most notably Ceph and CERNBox.

The group benefits from rich exchange between these projects, not only at the technical level, but also with personnel regularly spending time on multiple projects. This concentration of expertise is a community asset which is effective in training young engineers, supporting the grid-wide adoption of its projects, and sharing a pool of effort across the projects as requirements dictate.

The current pool of 6 developer FTEs, spanning a large range of contract types and experience, is 1 FTE for FTS, 4 FTE for EOS, 1 FTE for GFAL and Davix, and 1 FTE for XRootD. This team is augmented by an operations team with DevOps-style contributions encouraged across the two activities. With a flat effort projection this team is likely to experience increased pressure as its install base expands and as the components are exercised in ever-more demanding and diverse contexts.

4.4 Development plans

Database scalability The main area of focus for FTS Run 4 preparation has to do with the database scalability. Multiple options are being explored, namely, a different database deployment model, different database providers (currently FTS uses MySQL), a different queue-system entirely. This option is explored for the QoS daemon, which is responsible with long-standing operations (such as staging, archive monitoring and QoS transition).

Involvement in DOMA TPC, QoS and AAI Working Groups The FTS team will continue its involvement in the relevant working groups in order to finalise the TPC and token auth implementation and to track developments in QoS (for which it may play a role as a scheduler).

Network developments FTS has been identified as the possible the site of some of the richer interactions envisaged between orchestrators and the network layer. FTS may be called upon to provision network capacity in anticipation of transfer campaigns and may have a role to play in traffic shaping and packet marking. While nothing is planned in this area the team will track these topics.

Protocols New protocols may emerge to better exploit the networks of the future, and APIs may be extended or created to enrich interactions with storage. FTS has a well-designed, modular approach to protocol support and will be in a good position to support such developments.

4.5 Risk analysis and mitigation

Database scalability FTS's scheduler is currently based on the MySQL family of relational databases. Some Run 4 scenarios present a concern with scaling with the number of transfers and R&D may have to be undertaken to radically re-imagine the scheduling component, possibly using non-relational persistency.

Team Run 4 may represent an enlargement of FTS's role and significance, alongside possible new duties for example in network management. Adequate support in this case may require a reinforcement of the team.

5 Data Management Clients

5.1 Activity description

Exploitation of WLCG's storage systems is of course reliant on a set of client libraries and applications. In addition to providing protocol access to the storage endpoints, the clients play an integral role in the performance of the system and are often the subject of extensive optimisation for example regarding latency hiding or efficient use of Erasure Coded data.

The principal access protocols expected for Run 4, HTTP and xroot, have native clients which are built into an enriched client stack through the Davix and GFAL projects. Alongside this there is an associated population of clients offering alternative implementations (e.g. from the ARC project) and extended functions (e.g. token management).

GFAL (Grid File Access Library) is a C library with python bindings providing a protocol abstraction layer to grid storage. GFAL is an important shared component of the WLCG infrastructure, being the client library used by FTS for all its storage interactions, in addition to its extensive use in Rucio and DIRAC.

GFAL's goal is to offer a simple, common File API for file operations in a distributed environment. GFAL abstracts all the commonly used file access protocols in the Grid & Cloud environment. Protocol support is handled through plugins which typically link to native client libraries. HTTP support is implemented through Davix.

Davix is an HTTP-client library, adapted to the grid environment. The library is quite stable, only undergoing the occasional fix and/or feature release. It is used not only through GFAL but also forms the basis of ROOT's HTTP support and is used internally as a client in a number of systems.

The xroot client, managed within the XRootD project as the XrdCl component, is a crucial source of the i/o optimisations on which high-performance access, particularly over the wide area, depends.

With XrdCl being part of CERN's contribution to the XRootD project, the major client components cited above are all managed within the same team at CERN.

5.2 Management and communication

GFAL is an Open-Source project, distributed under the Apache 2.0 License. The project is managed by the same FTS team. The main source of communication is via the DMC developers mailing thread and ServiceNow or GGUS ticketing systems. The main stakeholders of GFAL are FTS, experiment frameworks and the DOMA TPC, QoS and AAI working groups.

Davix is an Open-Source project, distributed under the GNU Lesser General Public License. The main source of communication is via the Davix developers mailing thread and ServiceNow ticketing system. The main stakeholders of Davix are GFAL, DOMA TPC working group and the ROOT framework.

5.3 Team status and outlook

GFAL and Davix are maintained by the Storage Group at CERN, please see section 4.3 for further details.

5.4 Development plans

As data management clients, GFAL and Davix are called upon to track most developments of relevance in this document. The recent advances in TPC and Token AAI both required support in GFAL and Davix and this represents an ongoing commitment. Any new interactions, for example new tape or QoS interfaces, will need the same. GFAL and Davix are both stable projects, fully supported in a reactive mode.

5.5 Risk analysis and mitigation

The main risks for GFAL and Davix are related to the evolution of the client libraries on which they themselves depend. For example, GFAL's support for SRM relies on an ageing gsoap based client (maintained by the GFAL team) and Davix currently offers support for both libneon and curl, a situation which would ideally be resolved with a full migration to libcurl. None of these considerations represent major risks to the sustainability of the projects.

6 Storage technologies

6.1 XRootD and XCache

6.1.1 Activity description

XRootD is a highly flexible framework that provides primarily xroot[s] and http[s] protocol access to distributed resources; typically, but not exclusively, storage resources. The framework consists of multiple customisable and replaceable components (a.k.a. plug-ins) that can be composed to address specific resource access needs. The core framework supplies clustering (LAN and WAN), networking, plug-in, scheduling, and TLS services.

Numerous production quality plug-ins have been developed over the years to address community needs such as xroot[s] and http[s] protocols, popular authentication and authorisation mechanisms (Kerberos, Macaroons, SciTokens, x509), interfaces to various kinds of storage (Ceph, DPM, EOS, GPFS, HDFS, Lustre, Unix), commonly used checksum algorithms, tape access (CTA, GEMSS and HPSS), third party copy (TPC), proxy services, and full file (FRM) or block (XCache) data caching modes, among many others. The community is actively using the framework to provide robust data access solutions across multiple distributed services such as AAA, PRP, Qserv, SE, Slate, StashCache, XCache to name a few.

The framework is open source and freely available via github, EPEL and OSG repositories.

6.1.2 Management and communication

The XRootD Collaboration is the umbrella entity responsible for the XRootD and XCache projects. The collaboration "entity" exists to provide visible and essential community support for planning, architectural consistency, feature discussion, problem resolution, code reviews, contribution integration, and software releases. As this requires continuity, the collaboration is officially established under the auspices of CERN, SLAC, and UCSD to enable such continuity. Otherwise, the management structure is collaboratively distributed and consensus driven while being subservient to experimental needs.

Priorities are established by collaborating with the experiments, OSG, and WLCG, while taking into account available resources. Progress is continually monitored in weekly meetings and direct outreach.

Community communications are centred on two listserv mailings (xrootd-l and xrootd-dev) and the XRootD github ticketing system. Additionally, key personnel attend weekly meetings with critical path stakeholders and meetings where XRootD related discussions occur. The project also relies on its contributors to engage their communities to prevent surprises and provide critical feedback. Finally, annual workshops are held to promote community understanding of commonalities, priorities and needs.

6.1.3 Team status and outlook

As a community supported project XRootD relies on the contributions of developers and users across the wider XRootD community with the definition of the team constantly changing as people join and leave the project. Development effort varies between 1 and 4 FTEs. The closest to an accepted definition is the core team of key people that provide project continuity with development effort at about 1 FTE. Unfortunately, at the start of the year the project lost the http lead due to a job reassignment in their home institution. That role is now being fulfilled, to some degree, by OSG personnel while the project on boards a new contributor that will be the primary lead for http activities.

6.1.4 Development plans

Plans are determined by community needs and as of this writing future development has not been finalised other than packet flow marking for 3Q21. Currently, focus is placed on fixing edge-case software issues that arise when it is used in novel ways . The majority of emerging needs can be addressed by composing existing plug-ins (e.g. cloud storage). Emphasis is being given to development that improves performance and automates data integrity, especially in light of Run 4 needs. This approach will likely continue in the run-up to HL-LHC.

The project has an established and predictable time-line as documented in the Software Release and Support policy: https://xrootd.slac.stanford.edu/policies/SWRSP.htm and development is scheduled, within that policy, to correspond to OSG and WLCG timelines.

6.1.5 Risk analysis and mitigation

As with all community supported open source projects, engagement and relevance is at the heart of survival. Given that the XRootD framework is used in increasingly different communities and contexts, the risk would appear small but it's hardly a given. Wide deployment across many communities is a blessing and a curse. At some point engagement will exceed the available resources and overall satisfaction will decline. Well before that happens, funding sources for key personnel may consider expending effort for such support as inequitable and not aligned with their priorities. Both are major risks. While the obvious mitigation is to change the funding model, in practice, this is extremely difficult to accomplish.

The major gap in the XRootD framework is HPC applicability. While XRootD has been deployed in HPC centres (e.g. NERSC and GSI) its full utility, especially XCache, cannot be exploited without RDMA support. Currently, XCache uses a markedly limited hybrid approach to address this problem. The lack of RDMA support severely limits the use of XRootD for I/O intensive HPC workloads. This issue has been brought to the attention of some HPC-engaged experiments. Thus far, US ATLAS has shown interest to the extent of requesting a cost/time estimate.

Quality of service (QoS) is currently implemented by named cgroup which is similar to SRM space tokens but is more extensible. This may not to be where the community is heading. On the other hand, as it is not yet clear what the actual requirements are, this may or may not be a major gap.

The two instances above simply indicate that experiments are not yet ready to commit to specific feature development with potentially long-term payoffs. While completely understandable, the risk is when a decision is made the resources will either not be available or the timeline too short. The mitigation that the project is putting into place is to make the architecture more readily amenable to significant technological change by strengthening its component based approach to allow for faster development at a lower cost.

By explicit design, required core dependencies are limited to packages normally distributed or installed with Linux. Dependencies on third-party packages (e.g. Davix, Macaroons, libRados, Rucio, SciTokens, VOMS, etc) are limited to specific plug-ins. Such dependencies need to only be installed if a site wants to use a particular plug-in. This approach effectively shifts dependency support to the community that wants to use a particular feature.

6.2 EOS

6.2.1 Activity description

EOS is a large scale storage system developed at CERN currently providing 400 PB of capacity to both physics experiments and regular users of the CERN infrastructure. Beyond its T0 role, EOS is the basis for the ALICE O2 storage and has seen significant adoption outside of CERN. Since its first deployment in 2010, EOS has evolved and adapted to the challenges posed by ever increasing requirements for storage capacity. EOS is implemented as plug-ins to the XRootD framework. Files are stored using layouts as replicated or erasure-encoded files and organised in a hierarchical namespace using QuarkDB as a persistency backend. The frontend (MGM service) provides cached access to the namespace and other metadata. Storage nodes run one or several FST services to provide access to data stored on a locally mounted filesystem or remote storage. EOS provides access to files via the xroot and HTTP(S) protocols and via a FUSE filesystem. Third-party transfers (TPC) using the xroot protocol have been in production for over ten years while TPC using HTTPS was deployed in production in spring 2021.

6.2.2 Management and communication

EOS is developed within the IT Storage group (IT-ST) at CERN and is a fundamental project for Tier-0 and generic CERN services. The development team has seen a healthy fluctuation of developers with currently 4 FTEs and has had 12 major and 29 minor software contributors since 2010. EOS has a growing community of dozens of external sites running EOS instances. The EOS community meets at a yearly workshop (175 participants in 2021). External EOS support is provided via the CERN Service Now system and a community forum, where people can get into direct contact with developers and operators. EOS development uses a continuous integration system including nightly builds and an Open Source bug tracking system. EOS follows a frequent release policy with small change sets on a weekly or bi-weekly bases. In 2021 EOS is moving from major release version 4 to 5 due to a major version release of the underlying software framework (XRootD), which now provides data confidentiality (encryption on the wire).

6.2.3 Team status and outlook

EOS is maintained by the Storage Group at CERN, please see section 4.3 for further details.

6.2.4 Development plans

General DOMA activities In the context of DOMA activities the EOS development team has integrated the following for Run 3:

- Macaroon and WLCG Token Support
- Third-party copy support using HTTP(S) protocol (TPC/HTTPS)
- Generic QoS interface
- Front-end for the CERN Tape Archive

WLCG token and TPC/HTTPS technologies serve as replacement technologies for X509 certificate based authentication and gridFTP TPC protocol. They are provided by external XRootD plug-ins, which are neither developed nor maintained by the EOS development team. The long-term support and evolution of these plug-ins has to be guaranteed in the community. The token plug-in has been integrated into the EOS release process to keep dependency problems at minimum, the TPC plug-in is part of the XRootD release. The token plug-in is moreover highly security relevant and both plug-ins expose EOS services to performance issues and memory leaks originating from those plug-ins. The expectation for EOS is that these plug-ins will allow ATLAS, CMS and LHCB to run in a similar mode as the ALICE experiment has been running for 10 years with existing TPC/xroot and ALICE token support. The need for gridFTP gateways in front of EOS instances can be dropped since TPC transfers originate and target directly storage servers (as they do for TPC/xroot). In summary these changes allow for an operational simplification for three LHC experiments.

The current usage of WLCG tokens uses a minimal feature set laid out in the WLCG token description document. Part of the token model is in conflict with the authorisation mechanism implemented in EOS. A WLCG token can only be used as an additional authorisation check inside the EOS storage system, but it cannot overwrite the access control definitions configured inside the storage system. Similar to grid map files defining the mapping of certificates to DNs token require a policy for translation of the token bearer identifier to an EOS virtual identity, which might differ in the case of file access or file creation. The future usage of token scopes has to be evaluated and will require harmonisation with internal access control policies. While X509 authentication is connection based in EOS/XRootD, token authorisation requires a per request validation. With respect to Run 4 we don't expect any relevant performance impact induced by the transition from X509 to token based authorisation. The same statement holds for TPC/HTTPS vs TPC/xroot assuming they have an equivalent level of maturity and stability

EOS is the native front-end of the CTA service. Details and implications are discussed in the CTA service section.

QoS and Run 4 R&D As part of a DOMA coordinated activity and the XDC project a generic QoS interface has been added to EOS based on the CDMI standard. While the interface itself might be irrelevant or obsolete for Run 4, the concept of QoS won't.

EOS development and R&D is very active in the field of QoS. The EOS team is evaluating the performance of the xroot and HTTPS protocols in the context of high-performance media such as NMVe based storage devices and also erasure coding (EC) based on native EOS EC, XRootD XrdEc and CephFS EC using low-cost disk-based hardware. These technologies might play an important role for modern high-performance analysis frameworks like ROOT/RNTuple, which benefit from parallel IO with low latency and ultra-high per file streaming performance. EC can likely also replace expensive SSD storage in the CTA project. Another aspect of the R&D is client-side scalability in multi-threaded environments: how clients like XRootD/XrdCl or HTTPS/Davix have to be modified to deliver maximum performance using 100GE+ technology. This field requires a close collaboration with framework developers for modifications inside experiment frameworks. On another topic tests show that a simple dynamic asynchronous read-ahead mechanism in clients performs very well in ROOT based analysis use cases and should probably be added to remote clients. The EOS team has Davix and XRootD/XrdCl development inside the storage group at CERN and is in close contact to initiate/discuss these type of developments. The XRootD server implementation can also profit from a modernisation of its asynchronous IO implementation. Within the time scale of Run 4 this is is becoming a realistic scenario since such a development is tied to a minimal LINUX 5 kernel version. A first round of discussion has taken place and should be continued.

While external QoS management by frameworks like Rucio could be an option, policy driven EOS internal QoS management seems a more realistic, scalable and promising scenario for Run 4. The EOS development project is actively working on testing and evaluating manual and policy driven QoS conversions in production systems already before Run 3. An example of such a policy is to host files which are targeted for analysis use cases initially on SSD based storage system for a given time period until their usage cools off (time based expiration) and move them to a disk based archive QoS layout. In large storage sites such as EOS at CERN a policy driven QoS management is likely to be more efficient than a client driven passive caching approach like XCache or orchestration by Rucio.

The EOS team is (in the context of the ALICE O2 project and some generic R&D) evaluating the usage of very wide erasure coding layouts as RS(24,20), RS(22,20), RS(12,10) and others, which store data with reduced storage overhead for durability purposes but still provide interactive high-performance streaming access to the data at any moment. It has to be judged based on operational experience, which kind of EC layout and traffic amplification is still manageable in production.

Wide erasure coding layouts can be seen as an alternative to a model where experiments park or archive data on tape media. An example of such an installation is the KISTI EOS instance which serves as a Tier-1 tape replacement. It is part of the R&D exercise to identify which of the existing EC implementations will be the best match for Run 4 (EOS/XRootD/CephFS native).

To summarise: all of the mentioned technologies provide similar concepts in the field of QoS with slightly varying details concerning client scalability and network bandwidth overheads.

The last project to mention in this context is dynamic erasure encoding in EOS, whose main idea is to write data initially with high redundancy and to reduce redundancy on the fly when space is needed. Such a model profits from the fact that resources are not immediately fully used. The mathematical properties of erasure coding allow to drop a fraction of parities without inhibiting the ability to reconstruct data in case of disk failures with remaining parities as long as the minimal required amount of information is available for reconstruction. Such a QoS modification does not require data to be rewritten.

Scalability for Run 4 While it is straight-forward to increase the average file size in EOS to Run 4 scale (because new technologies allow to scale-up relatively easily IO per file), it requires a critical evaluation of file transaction rates, if file sizes are not expected to grow accordingly. Due to the hierarchical nature of the EOS namespace, file transaction rates are not linearly scalable. It is possible to scale-out file transaction rates for read access to WORM data, while it is difficult to achieve this for file write access without splitting namespaces. Therefore the experiment estimates of the overall number of files and file transaction rates are extremely relevant for the preparation of EOS for Run 4. As a consequence the initially mentioned plug-ins for token authorisation and TPC have to be tested for file transaction scalability as well as all stacked services sitting on top or behind EOS (Rucio, FTS, CTA).

Outlook Within the timescale of Run 4 even larger architectural changes within the EOS project are possible. These can be discussed and planned, once more clear requirements for data and meta-data access are available.

6.2.5 Risk analysis and mitigation

EOS relies on the XRootD project as a client server framework. The framework provides excellent functionality for authentication. authorisation and transport over the xroot and HTTPS protocols and there is no interest in replacing XRootD as a framework. If in the long-term the need should arise to drop XRootD as a framework due to missing community support, there are several possible paths to follow. Already today EOS provides a GRPC interface for meta-data access, all the internal communication could easily be moved to GRPC and any other protocol usable for high bandwidth transport between storage servers. The natural choice to offer to clients as a remote protocol would be HTTP(S), which is already available in physics frameworks and provided by the Davix library. Technically the XRootD server could be replaced by an open source C++ HTTP(S) server. Some more effort would be required to port plug-ins to a new server framework like the TPC/HTTPS plug-in. The biggest effort however would be to enforce changes in the user community by dropping a protocol and CLI. As very simple tests reveal, there is no better implementation of the xroot or HTTPS protocols available for file access available as an Open Source project with equivalent functionality.

XRootD has been established in WLCG for 20 years and has received major support and contributions from the CERN IT storage group as a collaboration member. The parallel HTTP(S) universe introduced by DOMA components de facto added a significant long-term commitment and man power investment for the EOS project. It is fair to say that it didn't help to simplify the EOS storage service, because changes in the WLCG community are slow and blurry and components are dropped only after years of decommissioning (e.g. SRM, now gridFTP).

The support of the native HTTP(S) protocol plug-in in XRootD is currently not well defined in the community and therefore to be considered for future planning. At the same time the EOS project has a natural interest in the evolution and continuation of the generic HTTP(S) protocol support.

6.3 dCache

6.3.1 Activity description

dCache is an experiment agnostic distributed storage system developed as a joint effort between Deutsches Elektron-Synchrotron (DESY), Fermi National Accelerator Laboratory and Nordic e-Infrastucture Collaboration. The project was started in 2000 to provide a highly scalable solution for handling the data volumes coming from the HERA and Tevatron experiments. Today dCache is used by nine Tier-1s and many Tier-2 centres around the world.

One of the core design aspects of the dCache storage system is an aggregation of a large number of data servers into a single-rooted distributed storage system that scales horizontally with the number of data nodes. dCache provides a POSIX compliant namespace and supports various access protocols and authentication and authorisation schemes that can be combined together to allow different scientific communities to use a shared infrastructure. In addition dCache can be connected to a tertiary tape storage system and transparently (to users) manage the data on tapes in a write-back/read-through manner.

dCache supports the following access protocols and authentications:

- DCAP, with GSI, Kerbros and password authentication
- FTP(S), with GSI, Kerberos, password authentication
- HTTP(S), with X509, Kerberos, password, OAuth2, SciToken and Macaroons authentication
- NFSv4.1/pNFS, with auth-sys and rpcsec-gss (kerberos) authentication
- xroot, with GSI and SciToken authentication

The HTTP, NFS and xroot protocols provide in-transit data protection through either the TLS or Kerberos layers.

6.3.2 Management and communication

The dCache software is developed as a joint effort between DESY, Fermilab and NeIC. All collaboration members use dCache as the primary system to store and manage the scientific data of WLCG and non-LHC communities. Moreover, many national and international projects in which stakeholder sites participate include dCache related activities. The dCache project goals are defined by collaboration members to satisfy each site's needs and are synchronised and prioritised at weekly developers meetings. The priority of a given task is typically adjusted according the scientific community's requirements, for example binding a delivery of a desired functionality to LHC operation schedule.

The dCache project has multiple communication channels to its user communities. This includes the issue tracking system, weekly telephone calls with Tier-1s and annual user workshops to disseminate future development directions and collect user requirements. The on-site communities usually have direct communication with developer teams. dCache participates actively in various DOMA sub-projects, addressing and demonstrating progress on the issues identified.

dCache applies a time-based release strategy, with regular feature releases every three months, and long-term releases every year, which are supported for two years. This gives sites the flexibility to choose the best migration path that suits their operational needs.

6.3.3 Team status and outlook

Despite the fact that dCache started in the year 2000 it relies on up-to-date technology. This is achieved by an ever ongoing modernisation of the code base. The goal of this activity is to adopt widely used technique to facilitate rapid progress by new team members. The development team consists of experienced software developers with deep knowledge in storage systems who mentor newcomers. The dCache software was used as an open-source project by a hands-on programming course organised by the University of Applied Sciences in Berlin. Every code change goes through a peer code review process and must have a unit test and pass regression and integration tests. This process safeguards high code quality.

As of today, the dCache developer teams consists of six persons at DESY, two persons at Fermilab and one person at NeIC. Their combined contributed effort is about 7 FTE. In the past 20 years more than 50 individuals have contributed to the code base.

6.3.4 Development plans

Enhanced tape scheduling Originally dCache's tertiary tape system connectivity interface was optimised for efficient writing to tapes while file recalls were expected to be relatively rare. The recent ATLAS Data Carousel activities with large-scale recalls have disproved this assumption. Unordered requests to tape libraries resulted in a high number of tape mounts and re-position requests, thus overall low tape access efficiency.

Generally, a tape system clusters requests by tape and does reordering of file requests based on their physical or logical location, potentially taking advantage of Recommended-Access-Order (RAO) capabilities. However, to allow such clustering and make use of available physical tape drives the dCache server should collect multiple incoming recall requests before forwarding them to back-end HSM system.

To address request grouping a prototype scheduler has been implemented and integrated into the SRM interface, which is currently used by FTS to perform a "Bring Online" operation. The scheduler decisions are made based on the tape inventory, which is tape library specific information. In simulations, the prototype scheduler achieved the expected improved tape recall efficiency, however, in-field tests are still required. Moreover, WLCG moves away from the use of SRM, thus the tape recall scheduler should be integrated into another dCache service available to all transfer and storage management protocols.

REST-API As mentioned above, WLCG data management had been using SRM heavily. Today, SRM usage is limited to interaction with a tape system only. Moreover, the SRM interface is based on Simple Object Access Protocol, or SOAP, for short - a technology that was popular in the early 2000s.

Since 2015 the dCache team has been working on an alternative high level storage management solution, which is exposed as "Representational state transfer", aka REST-API interface. A small subset of this API is identified as a common tape system interface by other storage providers at WLCG. Together with CTA, FTS and StoRM developers we are working on a common specification of REST-API that will be used instead of SRM bring online.

Integration with CTA The CERN Tape Archive (CTA) is an open-source storage management system developed by CERN to manage LHC experiment data on tape. Although today CTA's primary target is CERN Tier-0, many sites that run dCache consider it as an alternative to commercial HSM systems used at the sites.

dCache has a flexible tape interface which allows connectivity to any tape system. There are two ways that a file can by migrated to tape. Ether dCache calls a tape system specific copy command or through interaction via an in-dCache tape system specific driver. The latter has shown (by TRIUMF and KIT Tier-1s), to provide better resource utilisation and efficiency. Together with the CERN Tape Archive team we are working on seamless integration of CTA in dCache.

POSIX access for non-HEP communities Almost all sites that run dCache provide storage to non-HEP communities as well. The Cherenkov Telescope Array, Lofar, IceCube, Life science, Genome research, to name just a few. All these communities have their own requirements on how data is accessed. With the growing use of Jupyter Notebooks in combination with Apache SPARK or DASK as well as heavy use of HPC resources, low latency data access is crucial for the interactive analysis. Today, more than 50% of data at DESY accessed over NFS.

The dCache team was a pioneer in providing an NFSv4.1/pNFS interface to a distributed storage. We are not only continuously improving dCache's NFS implementation, but also regularly participating in NFS-community testing events to ensure compatibility with other implementations of servers and clients.

Through re-export of NFS-mounted dCache via a SAMBA server, data access can be provided to Microsoft Windows based clients that use commercial software, like MathLab, Origin or Mathematica. dCache server integration with Microsoft Active Directory via standard LDAP interface preserves the data ownership, group members ship and the access rights.

6.3.5 Risk analysis and mitigation

Risks dCache development effort is not pledged effort by WLCG and instead depends on DESY, Fermilab and NeIC support, and thus depends on their data management strategies and priorities. With a growing number of sites using dCache and the limited size of developer and support teams, overall user satisfaction may decline. Despite the DOMA coordination effort, often-changing experiment requirements and use-cases put an additional load on the developers and might lead to missed expectations on the users' side. Though large sites have good expertise in dCache deployment and even contribute to the project's code-base or documentation, such efforts are not sufficiently recognised by the HEP community and are rare.

The dCache software is written in the Java programming language. Though this gives us an access to a large knowledge base and well-tested libraries, the hardware and OS selection for the dCache nodes is limited to availability of a Java virtual machine for a given platform.

Missing functionality dCache is developed to cover a broad but nevertheless limited number of use cases. As typical HEP analysis doesn't require an HPC file system, dCache is designed to support so-called High throughput computing. This means that it scales well when multiple parallel jobs access different files and has poor performance when it comes to delivering a single file to a large number of clients. Moreover, the support for hardware capabilities is limited to the functionality exposed by the JVM, thus direct use of RDMA or NVMe is either impossible, or requires additional effort (with Java17, the next LTS release, a newly added support for Foreign Function Access will allow calling any native function within Java code).

Dependencies dCache comes with a number of external dependencies. The developers continuously revisit the use of external libraries and replace them with up-to-date versions or alternatives if required. However, some of those dependencies are WLCG/HEP community specific, have a low adoption rate, or are effectively abandoned. This is especially critical for security related dependencies that handle GSI, VOMS or other grid related aspects.

6.4 CTA

6.4.1 Activity description

The CERN Tape Archive ("CTA") is the archival storage system for the custodial copy of the data produced by CERN's experimental program. It is deployed as the tape back-end to EOS disk storage. Together, EOS+CTA replace the legacy CASTOR storage system. Its architecture is designed to allow it to meet the scalability requirements of Run 3 and the HL-LHC era.

CTA consists of three main software components: the Frontend (mediates communication between EOS and CTA), the Tape Server (controls tape-related hardware: libraries, drives and cassettes) and the Catalogue (database of file and tape metadata). The scheduling logic is embedded in the Tape Server which communicates with the Frontend via a queueing system. File namespace metadata and operations are delegated to EOS.

6.4.2 Management and communication

The CTA software is developed by the CERN IT Storage group, alongside EOS and FTS. The principal stakeholders are the CERN experiments, including the four LHC experiments and around a dozen smaller collaborations. CTA is also preserving data from numerous inactive or legacy experiments going back to the LEP era. Communication is directly with the data management teams or representatives for each experiment, via the CERN Service Portal or informal channels.

There are also a number of external collaborations. The CTA public instance archives data from external collaborations (ILC, DUNE) and data preservation use cases (BaBar). CTA will be deployed at RAL (UK Tier-1) and other sites external to CERN (e.g. AARNET and IHEP). The dCache developers are exploring the feasibility of using CTA as a tape back-end to dCache. CERN has entered a collaboration with Fermilab to evaluate the suitability of CTA for their use-cases.

6.4.3 Team status and outlook

CTA is maintained by the Storage Group at CERN, please see section 4.3 for further details.

6.4.4 Plans

Migration from CASTOR to CTA

CTA entered production in June 2020 and the four LHC experiments were migrated from the earlier CASTOR system to CTA by February 2021. The migration of CERN's other active experiments was completed in June 2021. A variety of other internal usecases and the data of inactive or historical experiments will be migrated before the start of Run 3.

The RAL Tier-1 currently runs CASTOR and plans a similar migration before the start of Run 3.

Preparation for LHC Run 3

CTA is feature complete for its main function of data archiving and retrieval for Run 3. In order to verify this and to establish that performance and reliability are adequate, the four LHC CTA instances have participated in in full-chain data challenges (DAQ → EOS → CTA). These have demonstrated around 40GB/s into CTA's archival buffer and further tests are envisaged before the start of the run once hardware procurement is complete.

Some features are being finalised on a Run 3 horizon:

- Repack: CTA can repack individual tapes, but changes are needed to the scheduling logic to manage an extensive repack campaign
- Scheduler: several necessary improvements have been identified for more efficient operations
- The final Run 3 deployment platform will require an upgrade from XRootD version 4 to 5, and perhaps other adaptations depending on decisions regarding future Linux platforms.

Finally, there is a community effort to develop a new REST API to allow stage-and-transfer from tape without using SRM. The primary use case is for HTTP transfers, which lack a staging mechanism in the protocol. The stakeholders for this work are the IT Storage group, the dCache developers and CNAF (Tier-1 site using StoRM).

Run 4 scalability and optimisation CTA was designed as CERN's future platform for archival storage, intended to serve the organisation for Run 3 and for the HL-LHC era. While validation of the system at Run 4 scale remains a few years off, the architecture is not considered to need review for HL-LHC. CTA's deployment at CERN, based on a fast SSD buffer, is intended to exploit the ever-increasing bandwidth that tape systems can deliver.

Certain aspects of system scalability will nevertheless need to be tested as Run 4 approaches. Scalability with number of drives and metadata scalability (with number of tapes, files and file operations) will need to be assessed. The issue of increasing file sizes will certainly affect the system and may need a rethinking of various aspects of the tape server and scheduling logic. In general, scheduling optimisations, will be possible, reflecting better understanding of experiment workflows (e.g. archiving tactics which promote read efficiency), or to adapt to technology changes (e.g. larger tapes which take over 24hrs to read), or to ensure efficient mass repack operations.

Platform and stack The CTA service at CERN runs with an Oracle database for its catalogue. In the medium term, it is envisaged that CTA will move to an open source database and support for both Postgres and MySQL is present in the code. A production deployment will require setting up and tuning the database for high availability and fast performance.

Evolution of tape usage and archive technology CTA will evolve to meet new requirements, including usecases based on backups, data preservation and digital archives. The team is well connected to the tape industry and is confident of its ability to track the roadmap for tape technology. More disruptive changes, such as new tape interfaces (e.g. S3-like APIs) or novel archive technology will be evaluated as required.

Data management themes Many of the themes under discussion in this document, such as access via caches, TPC, token support and packet marking, are delegated by CTA to its disk management system (EOS, and potentially dCache).

As a tape system CTA represents a particular QoS; low cost, high latency, high durability. At a fixed budget, a tape system can support "hotter" access by investing in drives rather than media and reflecting the necessary changes throughout the system. This is relevant to supporting use cases such as the Data Carousel.

CTA supports multiple archive replicas and geographical separation. While this could in principle be exploited to place replicas offsite in the cloud, this is not currently planned.

Community As there are an increasing number of non-CERN users of CTA, the establishment of a community is underway. This will bring additional requirements on the project, while of course also bringing resources and opportunities. These considerations include support for new disk systems such as dCache and support for other tape formats (particularly those used for existing data at other institutes which are considering CTA adoption). CTA will also support external deployments through a binary distribution free of CERN-specific dependencies.

6.4.5 Risk analysis and mitigation

Tape expertise is rare and the systems are complex. This means that staff turnover can be a concern and new recruits require considerable investment. This risk is offset by the shared talent pool in the Storage Group at CERN and CTA's numerous collaborations with other projects.

CTA's software dependencies are critical also for other projects (EOS, FTS) and the wider WLCG community. The main dependencies are XRootD, Google Protocol Buffers, Ceph RADOS and a relational database, currently Oracle. None of these projects are viewed as likely to be disruptive, but the move to an open source database will require careful examination of long-term scalability demands.

The tape market is highly consolidated and there is only one company investing in tape head technology and only two companies producing media. While this market situation represents a risk, the IBM tape technology roadmap seems fairly secure for at least the next ten years with analysts not anticipating any flattening of the areal density projections. The CERN tape operations team strategy is to use a mix of library, drive and media types inasmuch as this is possible. We have decommissioned our Oracle hardware and invested in two new Spectralogic libraries, with a mix of IBM and LTO drives and media.

The growth of the CTA community will place an extra burden on the central team. The mitigation is to strive to lower the barrier to contributions in order to enable collaboration models which augment the total effort within the project.

6.5 Echo

6.5.1 Activity description

Echo is the name given to the RAL Tier-1 disk storage. It is made up of a Ceph backend, with gateway machines providing access via GridFTP, Webdav and the XrootD protocol for external users. Each Worker Node on the RAL farm runs an XCache in a docker container that allows it to directly access the backend storage. Ceph manages the underlying disk storage and provides simple installation and operation, massive scalability and a comprehensive set of features for maintaining data resilience. The XRootD frontends provide the same set of features as describe in the XRootD and XCache subsection. The libradosStriper and XrdCeph components, which are maintained by RAL provide the interface between Ceph and XRootD. The accounting information and file list dumps are provided by a set of short python scripts which query the Ceph backend before uploading the relevant output file.

6.5.2 Team status and outlook

Echo is managed by the dataservice team at RAL. This team is made up of five people and supports multiple Ceph Clusters. There is also extensive knowledge of XrootD at RAL as we also run CTA/EOS and before that Castor. There is 1.5FTE development effort dedicated to maintaining the libradosStriper and XrdCeph plugins. Usage of Echo (and other Ceph clusters) by non-LHC VOs is growing rapidly, providing additional effort to support the service.

6.5.3 Development plans

RAL maintains the libradosStriper and XrdCeph plugins which provides an interface between Ceph and XRootD. The GridFTP plugin, which also relies on the libradosStriper plugin was written and maintained by RAL. Once the migration from Globus is sufficiently complete support for this protocol will be dropped. This will allow us to integrate the libradoStriper plugin with the XrdCeph plugin. Work is also underway to improve the way that the plugins handle Vector reads.

The primary access method for Echo of the LHC Experiments during Run 3 is expected to be via XrootD. Therefore to first order Echo can be considered an XRootD site and will provide the same features and development timeline as XRootD. This includes data access via distributed caches, TPC and Token Support.

6.5.4 Risk analysis and mitigation

The Ceph backend has widespread use across the globe and is maintained by Redhat. There is a Ceph foundation, which governs the development of the project. For XrootD the risk analysis has already been described. The libradosStriper and XrdCeph components are relative small well defined pieces of code that should be maintainable with the effort available at the Tier-1. If for some reason this is not possible, it would be possible to provide a CephFS storage endpoint and provide XrootD servers on top of that. Both the dCache and EOS developers have investigated putting their storage on top of Ceph, so that is another possibility.

The Ceph backend is an object store, rather than a file system. This means there is no directory structure and various commands are not supported. This was chosen as it simplifies the design and can scale better. It was possible to make this decision because the LHC experiments had all demonstrated that they didn't actually need a file system to run jobs. In practise however, people make assumptions about how the underlying storage works and there have been numerous incidents/bugs accessing data due to this. Over time as VO get used to working with object stores this risk decreases.

6.6 StoRM

6.6.1 Activity description

StoRM is a lightweight storage resource manager (SRM) solution developed at INFN, which powers the Italian Tier-1 data centre at INFN CNAF, as well as more than 20 other sites. StoRM implements the SRM version 2.2 data management specification and is typically deployed on top of a cluster file system like the IBM Spectrum Scale storage solution (aka GPFS).

StoRM has a layered architecture split between two main components, the StoRM frontend and backend services. The StoRM frontend service implements the SRM interface exposed to client applications and frameworks. The StoRM backend service implements the actual storage management logic by interacting directly with the underlying file system.

Data transfer is provided by the GridFTP, HTTP/WebDAV and XRootD services accessing directly the file system underlying the StoRM deployment. StoRM WebDAV, besides HTTP data transfer functionality, also provides a WebDAV-based data management interface and support for HTTP third-party copy.

StoRM is interfaced with the IBM Spectrum Protect via GEMSS, a component also developed at INFN, to provide optimised data archiving and tape recall functionality.

6.6.2 Management and communication

StoRM is an open source project, distributed under the Apache 2.0 License, and developed by the Software Development group (SD) at INFN CNAF. The principal stakeholder is the CNAF T1 data centre, whose storage services rely on StoRM to support the data management and access activities of several experiments (including the LHC ones).

Another important stakeholder is the EGI federation, which relies on StoRM for its Online Storage Service offering.

Communication with the CNAF stakeholder happens by natural collaboration between the CNAF Storage group and SD development group. Support requests and bug reports are submitted to the development team either through the official EGI support channels (GGUS), via the StoRM JIRA issue tracker or via Github (where the StoRM codebase lives).

The StoRM team actively participates in international collaboration boards (e.g., WLCG DOMA, EGI URT), to gather requirements and feedback, discuss evolution plans and report on development progress and support activities.

6.6.3 Team status and outlook

The Software Development group at INFN CNAF looks after the development and evolution of StoRM and supports the operations of the StoRM instances at the CNAF T1 data centre. The team also provide third-level support for instances deployed in other EGI/WLCG sites.

At the time of this writing, the Software Development group at CNAF is composed of six people, of which four have permanent positions at INFN. The SD development team is also responsible for the development and maintenance of other important components for the WLCG stack (INDIGO IAM, VOMS, Argus), with an estimate of 1 FTE being fully dedicated to StoRM support, maintenance and evolution.

6.6.4 Development plans

TPC StoRM provides support for third-party copy via two services: the StoRM Globus GridFTP and the StoRM WebDAV service. The StoRM WebDAV service implements the HTTP third-party copy protocol as defined by the DOMA TPC Working Group. There are no plans to provide direct support for the xroot third-party copy protocol.

Token support The StoRM WebDAV service supports token-based authentication and authorization in compliance with WLCG JWT profile and implements support for storage-issued tokens (aka macaroons).

The new WLCG tape management REST API being designed in collaboration with dCache, CTA and FTS developers will provide native support for token-based authentication and authorization.

Exploitation of tape StoRM integrates with tape storage using GEMSS, a component also developed and maintained by INFN. The StoRM/GEMSS combo as deployed at INFN CNAF has shown excellent performance in tape exploitation challenges like the ATLAS Data carousel.

StoRM is participating in the efforts leading to a new WLCG REST API for tape which will represent a modern alternative to the SRM frontend for tape recall management.

QoS and cost-savings StoRM supports basic QoS-management via a plugin for the INDIGO DataCloud Cloud Data Management Interface (CDMI) server implementation. The StoRM development team has interest in evolving the CDMI implementation in line with requirements emerging for WLCG as soon as those requirements are be defined.

6.6.5 Risk analysis and mitigation

Globus toolkit StoRM depends on the Globus toolkit in the StoRM frontend and GridFTP services. The StoRM WebDAV service, which has no dependencies on Globus libraries, provides support for HTTP third-party copy and is proposed as a replacement for GridFTP.

The StoRM SRM frontend depends on Globus for GSI authentication. A Globus-free SRM frontend which will rely on the VOMS libraries will be provided before the official Globus toolkit EOL (ie. March 2022, at the time of writing).

GPFS/SpectrumProtect costs StoRM has been successfully deployed on other POSIX file-systems (Lustre, CephFS). For integration with tape, IBM Spectrum Scale, GPFS and GEMSS are considered a mature, scalable and stable solution. There are currently no plans to migrate to an alternative tape management backend.

7 Network technologies

7.1 Activity description

WLCG relies on the networks as one of the critical parts of its infrastructure both within the participating laboratories and sites as well as globally to interconnect the sites, data centres and experiments'

					2027 Network Gbps				
	Percentage				*Minimal*			*Minimal*	*Flexible*
T1	**ATLAS**	**CMS**	**ALICE**	**LHCb**	**ATLAS + CMS**	**ALICE**	**LHCb**	**LHC**	**LHC**
CA-TRIUMF	10	0	0	0	200	0	0	200	400
DE-KIT	12	10	21	17	450	80	70	600	1200
ES-PIC	4	5	0	4	180	0	20	200	400
FR-CCIN2P3	13	10	14	15	450	60	60	570	1140
KR-KISTI-GSDC	9	15	26	24	480	110	100	690	1380
IT-INFN-CNAF	0	0	12	0	0	50	0	50	100
NDGF	6	0	8	0	110	30	0	140	280
NL-T1	7	0	3	8	140	10	30	180	360
NRC-KI-T1	3	0	13	5	50	50	20	120	240
UK-T1-RAL	15	10	3	27	490	10	110	610	1220
RU-JINR-T1	0	10	0	0	200	0	0	200	400
US-T1-BNL	23	0	0	0	450	0	0	450	900
US-FNAL-CMS	0	40	0	0	800	0	0	800	1600
Atlantic Link					*1250*	*0*	*0*	*1250*	*2500*
Sum	**100**	**100**	**100**	**100**	**4000**	**400**	**410**	**4810**	**9620**

Table 1: Network bandwidth estimates for T1s, based on rounded percentages of the pledges of tape.

instrumentation. Networks currently used by WLCG and DOMA are provisioned and operated in close collaboration with International and National Research & Education Network providers (REN) such as GEANT, ESNet, Internet2, etc. There are two dedicated virtual networks interconnecting the bulk of the existing resources, the LHC Optical Private Network (LHCOPN) connecting T0 and T1 sites and the LHC Open Network Environment (LHCONE) connecting T1 and T2 sites. REN support for the LHC program has been key to its success; a crucial aspect of the network evolution for HL-LHC will be a forward-looking technical engagement with all the major RENs, as changes in the WLCG network flows can have significant impact on all users of the RE Internet.

In the recent years, RENs have been able to continually expand their capacities to over-provision their networks relative to the experiments' needs and were thus able to cope with the recent rapid growth of traffic between sites, both in terms of achievable peak transfer rates as well as in total amount of data transferred. Some of the major RENs, such as ESnet, have seen traffic grow by a factor of 10 every four years. WLCG experiments have adapted their computing models to benefit from this trend by introducing a more interconnected system, moving away from strict tier-based hierarchies towards full mesh topologies, while at the same time increasing their overall traffic by 40-60% every year. Last year (2020), ESnet conducted a user requirements review, which has been followed by the WLCG and extended to the entire WLCG infrastructure in order to provide estimates of the network needs for the HL-LHC. The current numbers are summarised in Table 7.1. It shows both the "minimal" scenario in which network is considered a scarce resource, as well as "flexible" scenario, which expects similar network conditions to those of LHC Run 2 and 3. These estimates are in line with the DOMA requirements as well as with the estimates previously reported by the LHC experiments.

LHCONE network traffic currently accounts for around 40% of the entire REN capacity and with the estimated growth rates it has potential to dominate the REN traffic in time for HL-LHC. With other major experiments coming online in both HEP and non-HEP domains, such as SKA, LSST, CTA, etc. it is conceivable that LHC network traffic will come under pressure. At the same time, RENs will be facing multiple challenges in trying to ensure the estimated growth rates are delivered including funding, complexity, vendor equipment timelines and deliveries, speed and evolution of the market as well as available space and power.

New content delivery models, such as data lakes, as well as recent advances in network virtualisation technologies and programmable networks will have broad impact on the design and provisioning of the data centre networks. Network virtualisation and programmable networks could reduce many of the needs for over-provisioning and address the challenges above, but both areas are very actively evolving, which requires coordination and focused effort to understand how they can be adopted, deployed and operated and how the inter-play between LAN and WAN will be organised in the future.

In the context of the DOMA project there also additional challenges related to networks:

- The network will become a critical resource that needs to be provisioned, monitored and exploited in a way to ensure sufficient capacities are available. Potential delays associated with the network capacity provisioning will need to be factored in, so that the connectivity of the WLCG sites will be ready on time and at the appropriate level of capacity for the data lake model to work.

- T1s and major T2s will need to deploy network capacity for delivering data to remote sites, in addition to the capacity required to deliver data to their local applications. In addition, their operational effort to maintain the new content delivery model should not outweigh the effort saved by eliminating storage and network infrastructure in the small to medium sized WLCG sites.

7.2 Development plans

Several network R&D activities complement and support the DOMA project and the data challenges program. They are carried on in parallel to the existing DOMA activities and their outcomes should be integrated progressively in the network services so that they can be commissioned at scale in future data challenges.

Existing network-related activities can be split into two particular areas: **network provisioning**, which is focused on testing and provisioning new network capacities, and **network capabilities**, which focuses on extending current network capabilities in the areas of network monitoring, network transfer efficiency, network orchestration and network-aware applications.

7.2.1 Network provisioning

These activities are focusing on making sure the capacity of the network will be able to meet the demand, not only by adding expensive links, but also by making sure all the existing capacity is used in an efficient way and not left idle. This is important not only on the long distance WAN networks, but also in the on-site data-centre networks and their interconnections, to make sure there aren't bottle necks in any point of the path from the source to the destination.

In these years preceding LHC Run 4, WLCG will challenge the network capacity and its ability to deliver what is expected, by pushing the complete system (network, storage, applications) to its limits. All the components will have to be made aware one of the others and of their status and, most importantly, made be able to dialogue together, to optimise transfers by avoiding errors and congestion.

A first step is the mere capacity planning that both WLCG sites and RENs have to do and implement, to make sure the infrastructure is ready to meet the minimal capacity required: datacentre networks, interconnections with LHCONE and LHCOPN, LHCONE and LHCOPN themselves, including access to opportunistic resources as HPC centres and commercial cloud services.

Network capacity is a static resource: fibres, optics, transmission devices, routers, everything has to be pre-installed in order to be used. What can be dynamic is the way the bandwidth is assigned to the users: once the minimal capacity is in place, dynamic circuits that make use of the spare bandwidth can be created on demand. These on demand circuits can be used to temporarily increase the bandwidth between two sites that are transferring a lot of data. A project like NOTED is aiming to detect large FTS data transfers in order to trigger actions that could increase the available bandwidth between the source and the destination of a large transfer, just for the duration of the transfer. Initiatives on DTNs (Data Transfer Nodes) aim for similar results, by requesting large interconnecting links only where and when they are needed. To make dynamic circuits between any possible pair of sites, RENs are working together to build systems that can provision point-to-point circuits on multi domain networks. There are also initiatives that are exploring on demand use of the private networks of the worldwide cloud and content providers.

Datacentre networks also have to make sure the minimal capacity needed is in place; that network devices are able to deliver all the flows, especially the long distance ones, without packet loss due to buffer overflows; that overlay networks can be built dynamically by automation; that security is not neglected or sacrificed for the sake of performance. Adoption of software defined, cloud native networking will be essential.

7.2.2 Network capabilities

Activities and projects focusing on *network capabilities* complement *network provisioning* by improving visibility into our networks, increasing transfer efficiencies and permitting the design and development of advanced stateful network management systems and network-aware applications. Advances in this area will be crucial in mitigating capacity deficits while at the same time contributing to our ability to fully utilise the existing bandwidth we have today. There are multiple activities and projects in each area, some of which have been recently established while others have been on-going for several years now.

Reliable and complete network usage monitoring is one the primary objectives as detailed network traffic monitoring still remains a challenge. Understanding the HEP traffic in detail is critical for understanding how our complex systems are actually using the network. There are various aspects which are covered by different activities and projects:

- **Network and flow packet marking** is a recent project performed in collaboration with RENs telemetry activities, which aims to provide network usage monitoring of the experiment activities along the network paths. This will help correlate the existing transfer monitoring with the network usage seen by the RENs and will pave the way towards more advanced network telemetry and tracing.
- **Site and REN traffic monitoring** is another area, which has been recently initiated in conjunction with the upcoming data challenges and aims to provide REN and site inbound/outbound traffic available to the experiments.
- **Debugging performance issues** The WLCG Network Throughput Working Group, in collaboration with OSG and WLCG sites, operates a comprehensive network of monitoring agents based on perfSONAR. This activity has been critical in triangulating complex network problems, decreasing time to resolution while at the same time providing advanced services such as network analytics and alerting.

Network transfer efficiency has been constantly improving as new technologies are becoming available. The overall goal is to fully utilise available capacities while at the same time improving the way networks are shared between multiple experiments and/or science domains. Traffic shaping is a recent initiative performed in collaboration with RENs that aims at shaping the network flows to better match the end-to-end usable throughput. The same mechanism can also be used to throttle the traffic, introduce network level QoS and avoid unintentional bursts in network usage. There are also important external developments that will impact transfer efficiency, such as new congestion control algorithms and various network stack optimisations, which are being evaluated on a regular basis by ESnet and other RENs.

The next generation network orchestration systems designed as part of the GNA-G, AutoGOLE/SENSE working groups aim to provide a stateful network management system that could address many of the challenges mentioned above. SENSE and ROBIN are examples of projects implementing such a novel architecture, which uses software-defined, intelligent orchestration and scheduling to manage both storage and network resources. The existing testbed currently offers on-demand provisioning of end-to-end network circuits with guaranteed QoS using a high-performance data transfer engine and high-end Data Transfer Nodes (DTNs).

7.3 Risks analysis and mitigation

The success of the networking plans outlined in the previous section will depend on finding and assigning effort to complete the needed work. Historically, this has been a significant challenge because networking has not been directly supported by the WLCG experiments at anything approaching the level of effort of work in the computing and storage areas. Instead, the experiments have primarily relied upon best effort work by a small number of physicists and computer scientists in conjunction with various network research projects and the activities of the various research and education networks involved in supporting science globally. Because of this, there are a set of risks, gaps and dependencies that need to be addressed.

In terms of **risk**, WLCG has a few high-level items to track:

- The WLCG experiments will have insufficient available bandwidth to effectively utilise the set of globally distributed resources available. One way this might arise is due to a number of new science domains of LHC scale coming online by the time of HL-LHC.
- Network capacity costs could be passed on to the experiments, resulting in an unplanned cost with significant implications for the planned computing model.
- Sufficient bandwidth could be available, but not effectively usable by the experiments due to configuration, architecture or application design deficiencies.
- New services available in the R&E networks are unable to be used by the experiments because the experiment's software has not planned for integration of those capabilities.

As we look ahead to HL-LHC, we see there are gaps in identified effort, expertise and service integration for network capabilities. The experiments are already swamped in dealing with existing plans to evolve their software, computing and storage infrastructures to work at HL-LHC scale. Networking, while fundamental to their globally distributed infrastructure, has been one of the most reliable and capable components, and so it is difficult to justify assigning new effort while so much has to be done. The worry is that with many potential global-scale, data intensive science domains coming online, we need to make sure we have the tools and infrastructure in place to optimise our ability to use the networks we have.

Lastly, we note that in the current mode of operation, we have significant dependencies that are outside of WLCG control. Most obvious are the research and education networks that we critically depend upon. Much of the capacity and capability we foresee comes from these networks and their development and prototyping. We believe WLCG needs to continue the very successful collaboration created over the last 20 years with the RENs, the various network research projects with a HEP focus, and the WLCG site networks.

8 Related projects

The Common Software Activities discussed in this document are of course part of a wider set of shared projects on which WLCG's data-oriented activities depend. Some are themselves storage related, such as CVMFS. Others are AAI systems like IAM and VOMS which will manage access to DOMA resources. There are information systems such as CRIC involved in discovery and accounting. Projects such as perfSONAR aid in the monitoring and operation of data distribution.

The editorial approach has been *not* to include details of such projects unless there was a specific cause for concern which should be raised.

9 Summary and conclusions

This document presents in considerable detail the current state and plans of all the Common Software Activities under review, highlighting the major themes dominating planning discussions within the CSAs, experiments and sites, as well as a consideration of the risks.

WLCG and DOMA have already identified a number of Run 4 oriented themes requiring attention, as outlined in Section 2.4, and have established groups to iterate consensus and follow developments from discussion through prototyping to production. Notably, discussions with communities outside of WLCG are ongoing towards interoperable environments. Moreover, with data taking at the LHC approaching once more, the attention of the CSAs is focused on a number of pressing issues for Run 3 readiness, which will ultimately serve as the basis for the Run 4 platform. This review therefore underlines how seriously Run 4 planning is taken already. Concerning this effort, the critical situation of personpower across the projects cannot be understated.

With the first year of Run 4 production scheduled in 2028, details on many important points, both conceptual and technical, are nevertheless unresolved and significant decisions still remain open. We have emphasised the importance of the mechanisms which elevate particular topics to the community level in order to reach consensus. No doubt the subsequent guided iterations will pass through a number of intermediate phases before the final resolution is reached. An understanding of our Run 4 readiness will therefore be sought through consideration of the ongoing *planning* as much as through consideration of the strategic *plan* itself.

We conclude by quoting a sentence from the original guidance received with the mandate for this review: *The review is intended to establish that there is a credible plan and that all parties are "on the same page"*. We have documented the processes comprising what we have referred to as the "guided iterative" approach in reaching a Run 4 computing platform. It is precisely these mechanisms which are keeping all stakeholders on that same metaphorical page.