dCache, the Agile Storage Technology

Patrick Fuhrmann[1], Gerd Behrmann[3], Christian Bernardt[1], Dmitry Litvintsev[2], Paul Millar[1], Tigran Mkrtchyan[1], Antje Petersen[1], Albert Rossi[2], Karsten Schwank[1]
[1] DESY, Deutsches Elektronen Synchrotron, Hamburg, Germany
[2] FERMILAB, Fermi National Accelerator Laboratory, Batavia, US
[3] NDGF, Nordic Data Grid Facility, Copenhagen, Denmark

E-mail: patrick.fuhrmann@desy.de

Abstract: For more than 10 years, the dCache storage technology is managing data for high-energy-physics experiments around the world. Starting with the HERA experiments at DESY and the Tevatron experiments at FERMIlab, via the currently active detectors at the LHC, it is now preparing for the challenges of the upcoming linear collider. As a side effect of this increasing flexibility in terms of dCache partners, customers and technology features, dCache becomes increasingly interesting for non-HEP communities as well. This paper gives an inside into the dCache collaboration and deployments and illustrates the effect of design decision on the current dCache feature set.

1 The dCache System Cheat Sheet

dCache is a technology to store and deliver data in a highly scalable way. Heterogeneous storage nodes can be added to the system and nodes can be decommissioned without interrupting production services. The user view of the data is based on a single name space tree, being identical for all supported protocols, and independent of the actual location of the data within or outside of dCache. dCache is attached to a large set of tertiary storage systems in production. It provides a variety of storage management features, as there are 'Hot Spot Detection', 'Multi Tier Storage', migration of data from old to new storage nodes and many more, as described in section 4. dCache supports a large set of standard protocols to transfer and manage data, e.g. NFS4.1/pNFS[1], GridFTP[2], WebDAV/Http[3] and the Storage Resource Manager Protocol, SRM[4] as well as some community specific protocols. dCache systems can be configured to identify users by X509[17], Kerberos[18] and user/password credentials and allows user to be mapped to user and group IDs by calling out to existing infrastructures, e.g. using LDAP.

2 dCache System Deployments

As dCache is an open source software product, and as such doesn't have a central registration for its installations, the dCache team only has a rough estimation on the number and sizes of the installations around the world. Within the World Wide LHC Computing Grid[5], installations and sizes are frequently obtained from the GLUE information systems[6]. For other communities we can only guess from the requests in our bug tracking system. We estimate that, at the time being, around 80 dCache instances are deployed around the world. The biggest are very likely the 7 Tier I centers of WLCG, namely SARA in Amsterdam, IN2P3 in Lyon, FERMIlab near Chicago, BNL near New York City, KIT in Karlsruhe, Triumf in Canada and PIC in Barcelona. In Russia, two dCache Tier I's are ramped up in Dubna and Moscow. In total, about 120 Petabytes of WLCG data is stored in dCache installations of which

the largest single instance holds about 14 Petabytes on disk and 40 Petabytes on tape. Services are typically spread over more than 300 physical nodes. A nice example of a highly distributed single dCache is the installation at the Nordic Data Grid Facility, NDGF. The NDGF dCache spans 4 countries, namely Sweden, Norway, Denmark and Finland, with head nodes located in Copenhagen. Each country contributes with its own disk and tape resources. Towards the outside world, the system appears as a single dCache instance with a single namespace, however, data is stored at the different Nordic countries. The smallest dCache, to our knowledge, is an installed on a Raspberry PI mini computer, serving 1 Terabyte of dCache with NFS4.1 and WebDAV.
Concerning communities, sites using dCache for WLCG tend to use dCache for other communities as well. Most prominent examples are FERMIlab with CDF, their Intensity Frontier groups and a public dCache for smaller groups, SARA in Amsterdam for the LOFAR[7] antenna community, DESY with BELLE, BELLE2[8], IceCube and their Photon Science, CFEL[9] and XFEL[10] experiments and many more.

3 Agility by partner and project selection

To our opinion, the success of dCache is, to a large extend, based on the selection of partners and international projects. Generally, dCache partners have a particular interest in a well functional and widely deployed system, as they are not only contributing to the dCache code, but dCache is their strategic technology to serve their customers. dCache started as a collaboration between FERMIlab and DESY. Later, the Nordic Data Grid Facility joined and most recently dCache.org is collaborating with the 'University of Applied Sciences' (HTW) in Berlin to get students involved in the design and development process of dCache and to make unlimited storage space available to them; storage space not influenced by financial interests, which, as we recently learned, often results in the violation of fundamental privacy rights. From 2010 to 2013, dCache was involved in the European Middleware Initiative, EMI[11], as one of the 4 major European Middle-wares, beside UNICORE[12], ARC[13] and gLite[14]. This participation helped dCache significantly in streamlining its quality and certification procedures and to improve the collaboration with groups of similar interests. Beside that, it made dCache more easily available to a larger customers base. In order to even improve that aspect, dCache is negotiating an SLA with the European Grid Infrastructure EGI[15]. On the German national level, dCache is part of the Large Scale Data Management and Analysis project, LSDMA[16], targeting for a German wide simple data management in terms of unified access and authentication protocols.

4 Agility by Design

Besides selecting competent partners and projects to provide an agile technology, the dCache team puts significant efforts in designing a basic framework, sufficiently flexible to accommodate current and future customer needs. A complete description of the design would exceed the scope of this document, however this paper is touching on some examples of dCache basic design principles and their consequences in terms of features.

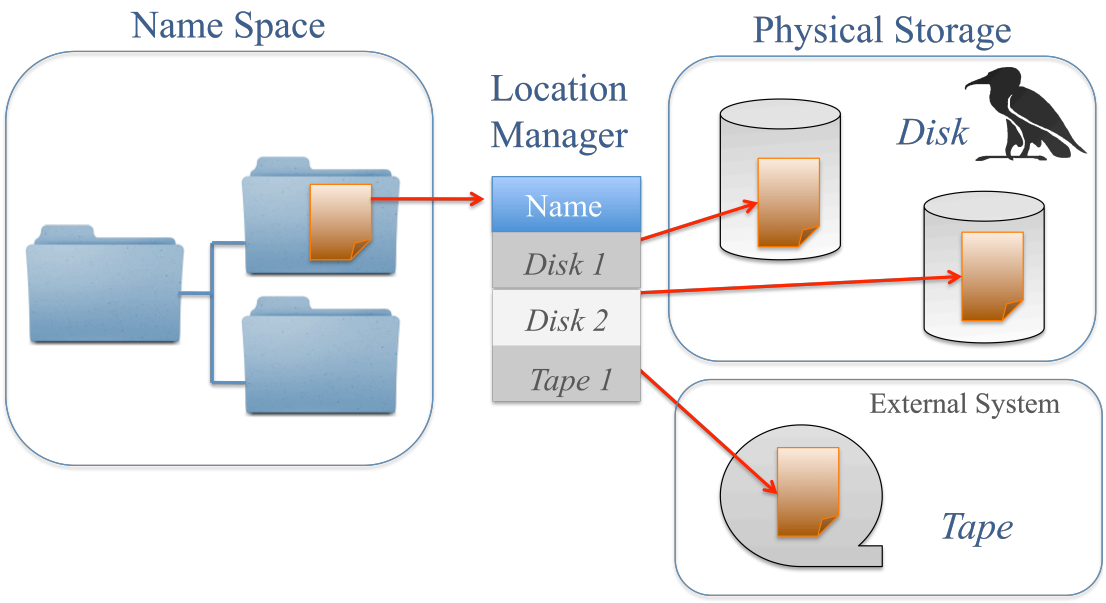

Figure 4-1

4.1 Smart Data Location Management

In order to horizontally scale in terms of size and performance, dCache allows distributing data among a large number of physical storage servers. However, the user accessible file system is represented under a single rooted files system tree. Consequently, in addition to the normal meta-data information, dCache stores the locations of file replica, which are either locations within dCache or external to dCache. External locations are mostly URI's, pointing to tertiary storage systems, e.g. tape systems. Figure 4-1 shows the relation between the file system entry and the associated locations on dCache pools and outside of dCache. All internal dCache services have access to that table and modify the content, if files are moved around. Based on that basic design principle, the following features are currently available in dCache:

4.1.1 Support of external file locations.

All, or a subset of dCache data pools can be configured to move data from disk to one or more attached tertiary storage systems. Depending on the policy of the data pools, files, available on tape, are removed from disk, whenever disk storage space is running short. File removal decisions are based on a 'least recently used' algorithm. If a file, only having an external copy, is requested, the data is fetched from that external source to a dCache disk, before it is delivered to the requesting client. Both, the store and the restore operation are handled transparently to dCache end users. Scripts perform the interaction between dCache and the external storage media. At the time being, dCache instances are connected to HPSS™, TSM™, DMF™, OSM™ and the Enstore tape systems. Example scripts exist for connecting dCache with cloud services, like Amazon S3 or others.

4.1.2 File replication on hot spot detection

Whenever a file is requested, and there is more than one copy of that file available on dCache pools, the system calculates the most appropriate replica for that request. The decision is two-folded. First, the dCache configuration system is determining those file locations, which are allowed to serve the requesting client; taking into account the IP number of the client and the requested transfer protocol, eg. NFS4.1/pNFS, gridFTP, WEbDAV, etc. If more than one file copy fulfills the configured rules, the

copy on a pool with the lowest load is picked. In case, the load of all allowed pools exceeds a certain threshold, dCache can decide to create a copy of that file on a less used data pool first, before delivering the file to the requesting party. This feature is known as 'file replication on hot spot detection' and equally spreads the load of the data servers on highly used systems. The system takes care that file copies are removed if load is dropping.

4.1.3 dCache file resiliency management

High availability storage systems need to guarantee that data can be delivered uninterruptedly even in case a subset of storage nodes is not available due to scheduled or unplanned downtimes. dCache can provide such a service by declaring a set of pools as 'resilient storage'. In that case, all files on those pools will get one or more copies on different nodes. dCache takes care that at each point in time, the number of copies of each individual file is between the minimum and maximum number configured. With that, (minimumNumber-1) storage nodes can be down without affecting overall system and data availability.

4.1.4 The Data Migration Service

With growing storage infrastructures, regular operations require to add new storage nodes or to decommission old ones. The dCache migration module allows handling those operations without system downtimes.
The problem of adding new pools is that those pools are normally empty. However, empty pools are very attractive for new incoming data and as such can possibly become a bottleneck. Although dCache has some mechanisms to mitigate that effect, the migration module allows leveraging the filling grade between old and new pools by moving data between them. This kind of data migration is done without interrupting the normal production service. On top, the migration module allows to shuffle data between pools manually or automatically based on tags assigned to file system directory tags.

4.1.5 Multi Tier Storage

Another feature, making use of handling multiple internal and external file copies, is the ability to optimize data access by storing data on media, most appropriate for a particular access profile. One example has already been mentioned earlier. Data, not used very often, might be stored on tape only, not occupying disk storage. However, if requested, dCache makes that data automatically available on disk again. That feature can be extended to support different access profiles. As regular spinning disks have advantages in streaming data while modern Solid State Disks are significantly better for random read access, dCache can be configured to create a copy of a file if it is requested from a particular set of compute nodes or by a protocol, which in general indicates random access, e.g. NFS. On the other hand GridFTP access is a very good indication that streaming access is requested, so that spinning disks would be the preferred storage media for that request.

4.2 The Authentication Management System

A second example for a flexible design in dCache is the way dCache manages authentication and user mapping. As each scientific community and sometimes even different sites within the same community realm use different types credentials to identify users, a static authentication system in dCache wouldn't be of any use. Therefore dCache provides an authentication framework, allowing to plug-in modules

for the different steps of the authentication and user mapping process. As a matter of fact, the

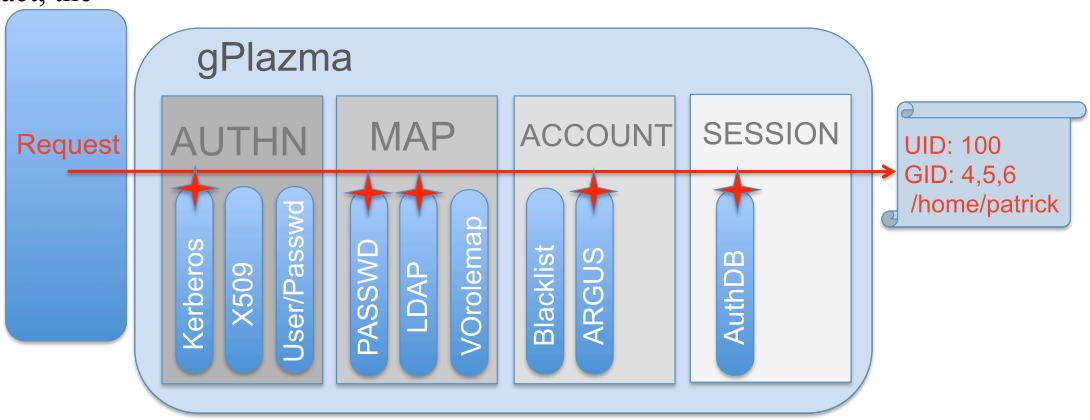

Figure 4-2

System distinguishes 4 steps in the authentication process, the login, the user mapping plus user account-and session operations (Figure 4-2). Site administrators can configure different modules for each step. The standard dCache bundle provides "ready to use" plug-ins for common cases and publishes the module interface specification for customers desiring to create their own customized modules. Each module within one of the steps can decide whether the provided information is sufficient to proceed with the module chain, to let the request fail or to jump to the next major step. At the end of the complete chain, it is expected that one valid user ID and at least one valid group ID is present. For the 'login' step we offer a X509[17], Kerberos[18] and a Username/password module, while for the mapping step we can callout to LDAP[19], a password file or check the Grid VORolemap file. In the user account step we check for blacklisted users in a configuration file or we can callout to the EMI-ARGUS[20] infrastructure service. As already mentioned, sites are free to configure a combination of those modules or to provide their own ones.

4.3 Generic trend to customizable frameworks

Over time, the dCache team replaced a significant part of the active static code by frameworks, where the provided implementation of a particular service can be replaced by different implementations. This, for example, allowed dCache to migrate from its former name space provider (PNFS) to the current, SQL based one (Chimera), to plug-in the CMS simple file catalogue name mapping service into the xRootd[21] doors and/or to let customers choose between two file distribution algorithms for incoming data. [22]

5 Work in progress

5.1 Cloud Storage and Identity Management.

With our recent collaboration with the computer science department of the University of Applied Sciences, Berlin, we have the great opportunity to involve future scientists into the active dCache design process as early as possible. In the currently running common projects, we are evaluating mechanisms to share data between scientists and modern ways, to publish data out of the cloud. With that we attempt to merge the historic way of scientists to handle data with the way social networks are evolving.

This includes rethinking of authorization but as well opens the horizon in terms of authentication and federated identity management. In the past, storage users had to be registered at the site, providing the actual storage service. Nowadays customers are requesting to use their social network credentials to fetch data from all possible sources, they have access to, including scientific data repositories. Consequently, dCache is extending its authentication framework in the direction of SAML and OpenID. Another project in collaboration with the HTW is the implementation of the first standard and well-specified cloud storage interface, the Cloud Data Management Interface (CDMI)[23] defined by SNIA. It is a very powerful way of handling data plus meta-data. However, the acceptance of that standard by industry remains to be seen. To evaluate the usefulness of a CDMI implementation, we are collaborating with the UNICORE team, as they plan to implement a CDMI client.

5.2 Dynamic Federation

In the context of WLCG, but not restricted to that infrastructure, we are collaborating with the CERN Data Management (DM) team to setup a worldwide WebDAV/HTTP federation system. The initial goal is to make all WLCG storage available through a single WebDAV/http endpoint, redirecting requests to the final source of the data. The software of the 'Dynamic Federation System' is provided by CERN DM. dCache is involved in the deployment and the interoperability testing of that interesting infrastructure.

6 Summary

This paper gives a high level overview of activities undertaken by the dCache collaboration to provide a future-proof data management and storage system. Our belief is that by the way we incorporate partner institutions and projects, we are fit for upcoming challenges in the context of WLCG and beyond. In terms of the technology itself, over the last years, we migrated dCache from a rather static storage system to a set of frameworks, providing 'out of the box' solutions for recent data challenges but at the same time being flexible enough to be extended when required.

7 Bibliography

[1] Alex McDonald NFS4.1 "NFS4.1 using pNFS" SNIA Webcast; http://snia.org/sites/default/files/Part4-Using_pNFS%20Feb._2013.pdf

[2] I. Mandrichenko *et al.* "GridFTP V2 Protocol Description" OGF Document GFD-R-P.047.

[3] Y. Coland *at al.* "HTTP Extensions for Distributed Authoring", IETF RFC 2518

[4] Arie Shoshanie *et al.* "The Storage Resource Manager Interface Specification, Version 2.2" LBNL Document, http://sdm.lbl.gov/srm-wg/doc/SRM.v2.2.html

[5] The Worldwide LHC Computing Grid (WLCG) web portal: http://lcg.web.cern.ch/LCG

[6] Laurence Field *et al.* "GLUE Specification V 2.0" OGF Document GFD-R-P.147

[7] Marco de Vos *et al.* "The LOFAR Telescope: System Architecture and Signal Processing" Proceedings of the IEEE, Vol. 97, No 8, August 2009

[8] David M. Asner "BELLE 2 Experiment Network and Computing" Cornel University Library, Report arXiv:1308.0672 [physics.ins-det]

[9] CFEL Portal: www.cfel.de
[10] XFEL Portal: www.xfel.eu
[11] Alberto Di Meglio *et al.* "Towards next generations of software for distributed infrastructures: the European Middleware Initiative" IEEE 8th International Conference on e-Science
[12] UNICORE Portal: www.unicore.org
[13] Oxana Smirnova "ARC Middleware and its deployment in the distributed Tier 1 center by NDGF" Proceedings of Grid 2008, Dubna, RU
[14] E. Laure *et al.* "Programming the Grid with gLite" Computational Methods in Science and Technology, 12(1), 33-45(2006)
[15] EGI Web Portal www.egi.eu
[16] Large Scale Data Management and Analytics, LSDMA Web Portal: www.helmholtz-lsdma.de
[17] D. Cooper et al. "Internet X.509 Public Key Infrastructure Certificate …" IETF RFC 5280.
[18] C. Neuman *et al*. "The Kerberos Network Authentication Service V5" IETF RFC 4120
[19] OpenLDAP Foundation "Lightweight Directory Access Protocol (LDAP)" IETF RFC 4510
[20] The ARGUS Web Portal www.switch.ch/en/grid/argus
[21] The xRootD Web Portal xrootd.slac.stanford.edu/
[22] Gerd Behrmann *et al.* "Weighted Available Space Selection" Presentation during the dCache workshop, 2012
[23] SNIA Document "The Cloud Storage Management Interface, Version 1.0.2" SNIC Technical Position, June 2012