International Conference on Computing in High Energy and Nuclear Physics 2012 (CHEP2012) IOP Publishing
Journal of Physics: Conference Series **396** (2012) 052006 doi:10.1088/1742-6596/396/5/052006

# Web enabled data management with DPM & LFC

**Alejandro Alvarez Ayllon, Alexandre Beche, Fabrizio Furano, Martin Hellmich, Oliver Keeble and Ricardo Brito Da Rocha**

European Organisation for Nuclear Research, Geneva 1211, CH

E-mail: dpm-devel@cern.ch

**Abstract.** The Disk Pool Manager (DPM) and LCG File Catalog (LFC) are two grid data management components currently used in production with more than 240 endpoints. Together with a set of grid client tools they give the users a unified view of their data, hiding most details concerning data location and access.

Recently we've put a lot of effort in developing a reliable and high performance HTTP/WebDAV frontend to both our grid catalog and storage components, exposing the existing functionality to users accessing the services via standard clients - e.g. web browsers, curl - present in all operating systems, giving users a simple and straight-forward way of interaction. In addition, as other relevant grid storage components (like dCache) expose their data using the same protocol, for the first time we had the opportunity of attempting a unified view of all grid storage using HTTP.

We describe the HTTP redirection mechanism used to integrate the grid catalog(s) with the multiple storage components, including details on some assumptions made to allow integration with other implementations. We describe the way we hide the details regarding site availability or catalog inconsistencies by switching the standard HTTP client automatically between multiple replicas. We also present measurements of access performance, and the relevant factors regarding replica selection - current throughput and load, geographic proximity, etc.

Finally, we report on some additional work done to have this system as a viable alternative to GridFTP, providing multi-stream transfers and exploiting some additional features of WebDAV to enable third party copies - essential for managing data movements between storage systems - with equivalent performance.

## 1. Introduction

### *1.1. DPM and LFC*

The Disk Pool Manager (DPM) and LCG File Catalog (LFC) are two grid components with more than 240 endpoints available.

The purpose of the LFC is to allow the access and retrieval of any file using just its Logical File Name (LFN), without the need to know where exactly it is replicated[1]. When a user does a request to an LFC, it will return a list of "Storage File Name" (SFN) where this file can be retrieved.

DPM is a lightweight storage system mostly targeted at Tier-2 and medium size sites, providing reliable and scalable storage while keeping configuration and management simple[1]. It maps a SFN to a Transfer URL (TURL), which is the association between a Physical File Name (PFN) and an access protocol such as RFIO, HTTP or GridFTP[2].

### *1.2. HTTP and WebDAV*

The Hypertext Transfer Protocol (HTTP)[3] is a well known and widely implemented standard protocol based on a stateless client-server model. It can be used as a data transfer protocol, where users request a resource from a specific server, which then sends the response back.

Web Distributed Authoring and Versioning (WebDAV)[4] is an extension of the HTTP protocol. It provides a framework to handle metadata through the web, so users can create and retrieve resources, and also modify its associated properties, such as owner, description, permissions and any other information that can be expressed as text or xml.

### *1.3. Objectives*

Currently, when a user wants to retrieve a file, a catalog must be queried for different replicas of that file. Then the client will be redirected to the site hosting that replica, and finally to its actual physical location within that site. Generally, this is done automatically, but specific tools are needed.

HTTP supports this kind of behaviour as the response of a request can be a redirection to a different location than the one originally demanded. The functionality is the same, but as it is an extremely widely implemented standard, data could be accessed from almost any kind of platform, from servers to mobile devices.

WebDAV gives us, in addition, metadata handling. It is also a popular protocol, with clients available for a wide variety of operating systems.

Giving support to these standards we allow users to access their data easily and transparently, from any machine, through any platform, and using clients they already know how to use - for instance, web browsers, as shown in figure 1.

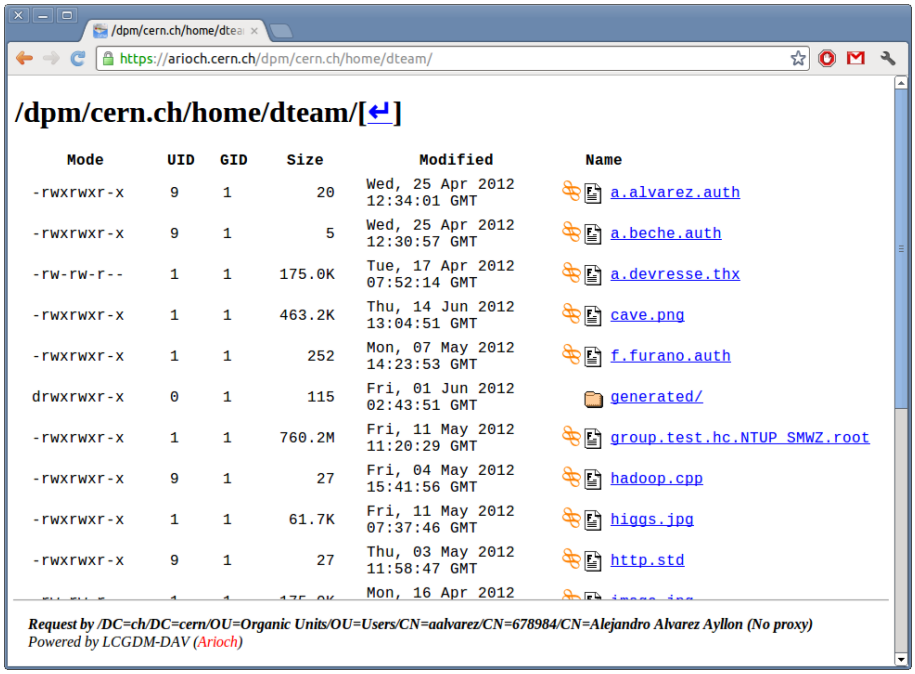


Figure 1: Accessing a DPM server using Google Chrome

## 2. HTTP for data transfer

In order to provide a reasonable alternative, our HTTP implementation is comparable in terms of functionality and performance to GridFTP, which is a transfer protocol that extends the standard FTP protocol and allows third party control of data transfer, multiple streams for a single transfer and partial file transfers, among others[2].

### *2.1. Multistreams*

With the term "multistreams" we refer to virtually splitting a single file in chunks, and downloading or uploading them in parallel. This can improve the general transfer throughput as it will mitigate the limitations of bandwidth per connection. We have run performance tests using different number of streams and we have found that although for Local Area Networks (LAN) the difference is neglibible, there is a big improvement when doing transfers through a Wide Area Network (WAN). These results are consistent with those of GridFTP.

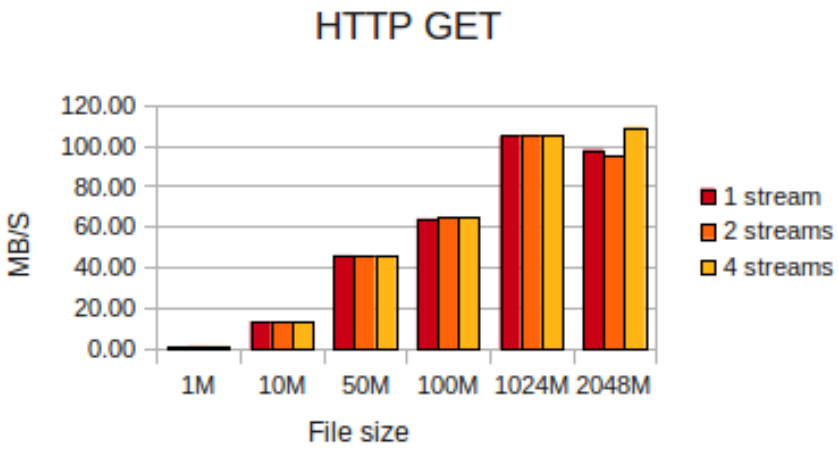


Figure 2: Multiple streams on a LAN

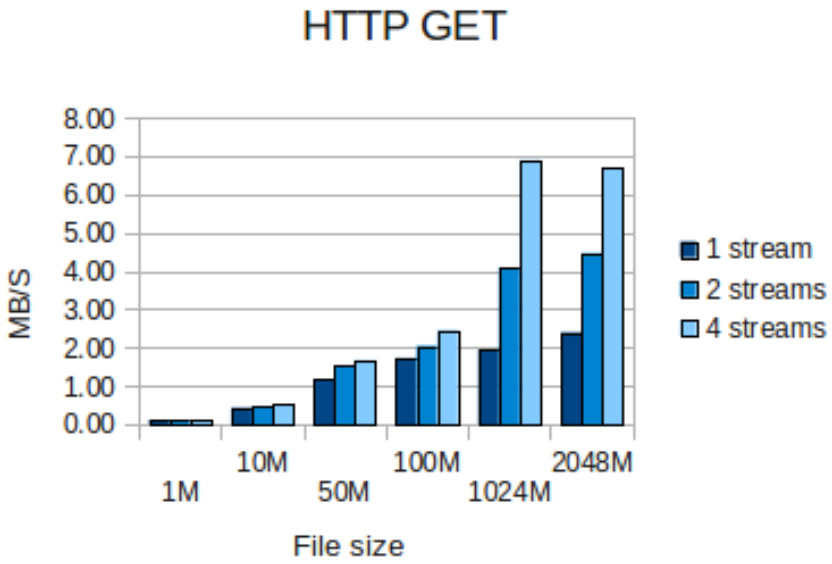


Figure 3: Multiple streams on a WAN

It is clear from these plots that multiple streams barely impact the transfer throughput on a Local Area Network (figure 2), while on a Wide Area Network the performance improvement is significant (figure 3).

*2.2. Third party copies*

Copying data from one server to another is one typical use case, which can be done either downloading and then uploading again the file to a different server, or delegating the copy to the source or destination server, which is more efficient. This is what is called "third party copy", supported by GridFTP.

HTTP by itself doesn't give this functionality, but DAV defines a COPY method that "creates a duplicate of the source resource, identified by the Request-URI, in the destination resource, identified by the URI in the Destination header"[4]. This restricts us, however, to *push* scenarios, where the source server orchestrates the process. This use case is displayed in figure 4.

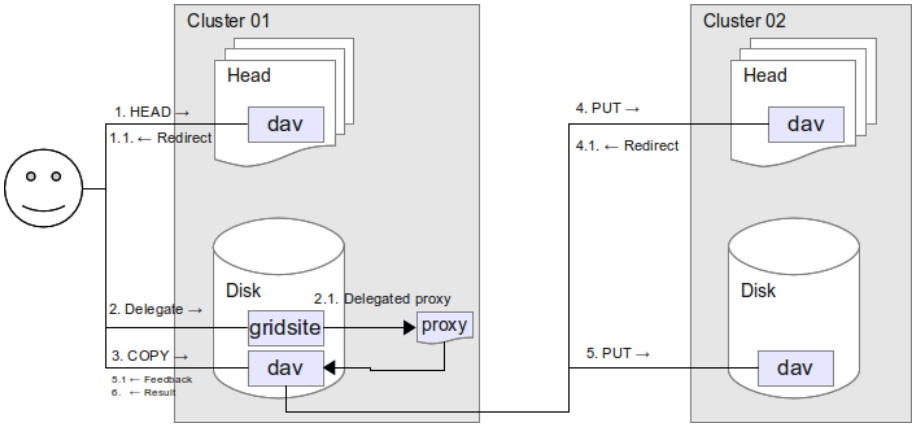


Figure 4: Third party copy

In this scenario, the source server needs to act on behalf of the user requesting the action. For this, the user will delegate temporarily his/her credentials to the service, negotiating a *proxy certificate* which will then be used by the source server to authenticate itself as the user against the destination server.

This means that any kind of http server can act as the destination, as long as it accepts this kind of certificates. However, both the client and the origin need to be aware of the additional DAV functionality.

*2.3. Partial file transfers*

This is already part of HTTP, which defines the "Range" and "Content-Range" headers for this purpose in the HTTP/1.1 standard[3]. In fact, this mechanism is used for the multiple streams transfers. A reference implementation for multiple streams and third party copy is provided as part of this project (Subsection 5.4).

## 3. WebDAV for metadata

WebDAV extends the HTTP protocol allowing clients to perform web authoring operations[4]. That is, it provides a framework to handle namespaces, so users can perform operations such as create, rename, modify or remove on files and directories through the Web.

For this, it defines new methods for manipulating properties and collections (i.e. directories), and the possibility of passing parameters through XML on a request body, instead of HTTP headers or queries, so they can be easily extended.

It also uses XML to encode the answers, such as the list of properties of a file, or the content of a folder.

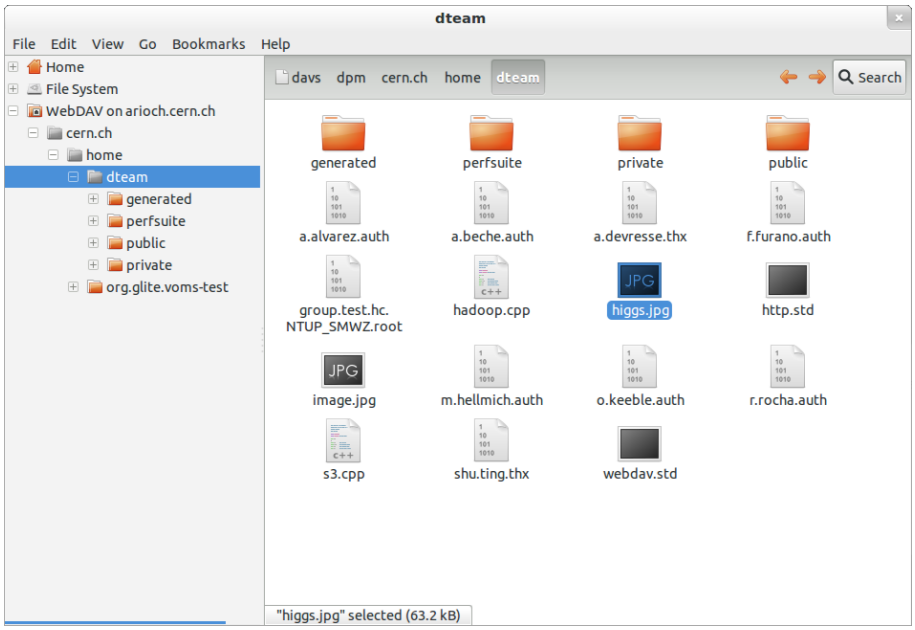


Figure 5: DPM accessed through DAV on Nautilus

WebDAV is supported by a wide variety of clients for several operating systems (see subsection 5.2). For instance, Nautilus on GNU/Linux (see figure 5).

## 4. Global Access Service

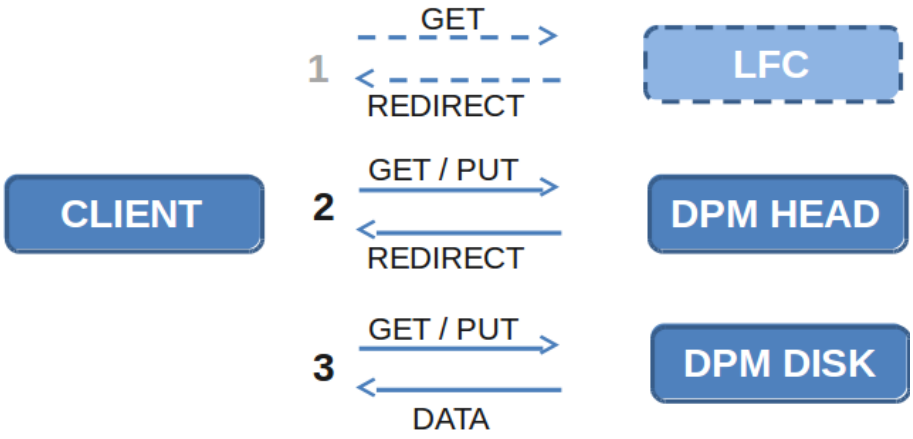


Figure 6: Redirection scheme. The same protocol is used from the catalog to the disk.

In the architecture of DPM/LFC, one entity handles the namespace with the metadata, and several different ones store the files with the actual data. This means that any request must go through a server hosting the namespace first, and then redirect to the disk node storing the physical file.

Additionally, one more level of redirection may be needed, since catalogs (LFC) can also be browsed through HTTP/DAV. These two basic use cases are represented in figure 6.

However, there is no guarantee that the selected replica is actually available. This is specially true for catalogs, which usually store multiple replicas for one single logical entry. In this case, the user usually will want to fall-back to any other available replica.

Using HTTP redirections it is possible to redirect the request to a first replica, but passing an additional argument indicating where other copies can be found in case of failure, so if the first attempt fails, the client will be automatically forwarded to another site. This is represented in figure 7.

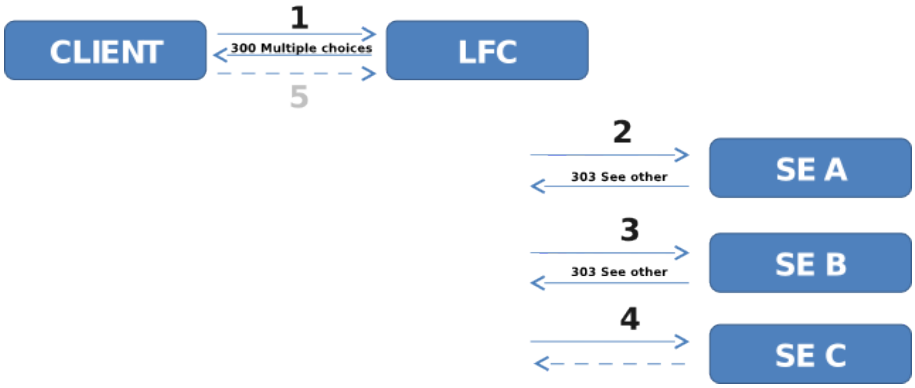


Figure 7: "Global Access Service" redirection scheme

The current implementation is server-driven. That is, the client just follows the redirections when the catalog, and then the storage elements, answer with a 30X code. This can be problematic if the server does not implement this logic, or if it is not reachable. Fortunately, dCache[5] will include support for this scheme in their HTTP/DAV implementation, so LFC, DPM and dCache will be able to interoperate following this redirection mechanism.

HTTP provides a status code that fits this use case - 300 Multiple Choices - but it does not specify how to feed the client with the actual choices. We will investigate this possibility further.

## 5. Client support

In this section we will briefly present the current support status of HTTP/DAV for external third party clients - subsection 5.1 for plain HTTP, subsection 5.2 for DAV, and subsection 5.3 for existing Grid tool.

Finally, in the subsection 5.4 we will introduce a reference implementation that we have developed for the third party copy use case.

*5.1. HTTP clients*

A DAV server can be accessed safely using plain HTTP, which allows the use of regular web browsers - Mozilla Firefox, Google Chrome, Opera,... - to explore a DAV endpoint and retrieve files. However, web browsers don't usually allow uploading files directly - without using a web form - so for that purpose curl can be used. Curl is a command line interface able to interact with several protocols including http/s. It can also be used for scripted access.

We compare the support of some key features of two of the most widespread tools used to access a web server in table 1.

| **Client** | **OS** | **GUI** | **CLI** | **X509** | **X509 Proxies** | **Follow redirects** | **PUT** |
|---|---|---|---|---|---|---|---|
| curl | Any | No | Yes | Yes | Yes | Yes | Yes |
| Web browser | Any | Yes | No | Yes | Only Internet Explorer | Yes | No |

Table 1: HTTP clients comparison

**OS** Operating System

**GUI** Graphical User Interface

**CLI** Command Line Interface

**X509** Support for client certificates

**X509 Proxies** Support for proxy certificates

**Follow redirect** Follows redirections when needed, as described in section 4

**PUT** Writing support

*5.2. DAV clients*

For performing namespace operations, such as creating directories, renaming or fetching properties, WebDAV must be used. Table 2 shows a comparison between the most popular WebDAV clients.

| **Client** | **OS** | **GUI** | **CLI** | **X509** | **X509 Proxies** | **Follow redirects** |
|---|---|---|---|---|---|---|
| Cadaver | Unix-like | No | Yes | Yes | No | No |
| Davlib | Mac OS X | Yes | No | No | No | Yes |
| Shared folder | Windows | Yes | No | Yes | Yes | Not on Win7[1] |
| DavFS2 | Unix-like | N/A[2] | N/A[2] | Yes | No | No |
| gvfs >= 1.12.1 | Gnome 3 | Yes | No | No | No | Not on PUT |
| Dolphin | KDE | Yes | No | No | No | Yes |
| Cyberduck | Win & MacOSX | Yes | No | No | No | Not on PUT[3] |
| libneon | Unix-like | N/A[4] | N/A[4] | Yes | No | No |

Table 2: WebDAV clients comparison

[1] Previous versions were able to redirect on GET.

[2] Filesystem driver.

[3] There is an open request for PUT redirection on confirmation from the user.

[4] Library.

The meaning of the columns is the same as described in subsection 5.1. Both tables are an adaptation and expansion of the presentation "WebDAV in dCache"[6]

*5.3. Grid tools support*

*ATLAS PanDA pilot* The Production and Distributed Analysis System (PanDA) is a key component in the ATLAS distributed computing[7]. Thanks to ASGC (Academia Sinica Grid Computing) there is a new PanDA mover (retrieves the data needed for an analysis) based on HTTP, which has been used to perform stress tests with Hammercloud[8] on our WebDAV implementation.

*dCache* is a storage manager that provides a file name space and a data pool manager[5]. It is used mostly at Tier1 and some Tier2 sites. They will include support for the global access mechanism.

*FTS3* The File Transfer Service is a service for transferring files between storage elements on the Grid. Currently FTS2 only supports SRM and GridFTP, but FTS3 will include HTTP as a transfer protocol.

*GFAL 2.0*[9] is a data access interface for grid storage systems which allows direct data interaction and transfers functionalities. It provides a single API for all the common protocols: GridFTP, SRM, RFIO, file, FTP, and, in the near future, HTTP/DAV.

*ROOT*[10] is a data analysis framework used by several experiments, including ALICE, ATLAS, CMS and LHCb. It already supported random I/O through HTTP, but a patch was submitted to include support for HTTPS and client certificates. These features have been available since version 5.32, released in November of 2011.

*5.4. htcopy*

We provide a reference client that supports multistreams and all the logic behind third party copies, including the credential delegation. Although it is intended just to provide a reference implementation, it is fully functional, and it supports downloading, uploading and third party copies.

This sample client is based on libcurl[11] - the underlying library behind curl - and it is able to reuse SSL sessions, a contribution made by our team to the libcurl source code. SSL session reuse improves the overall performance when more than one connection is established between two endpoints since it allows heavy initial hand-shake to be performed only once.

## 6. Performance results

As mentioned, our HTTP/WebDAV implementation aims to provide a good alternative to both GridFTP and Disk Pool Namespace (DPNS) / LCG File Catalog (LFC) specific protocols. To prove we are actually providing it, performance tests have been run in local and wide area networks.

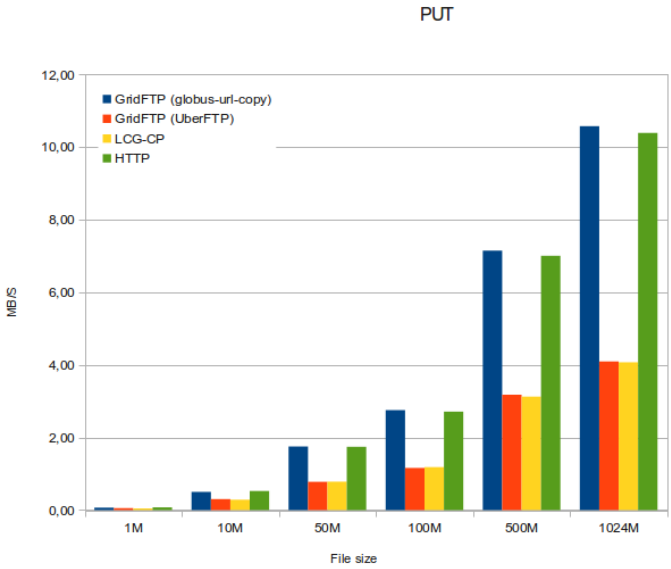


Figure 8: Uploading a file

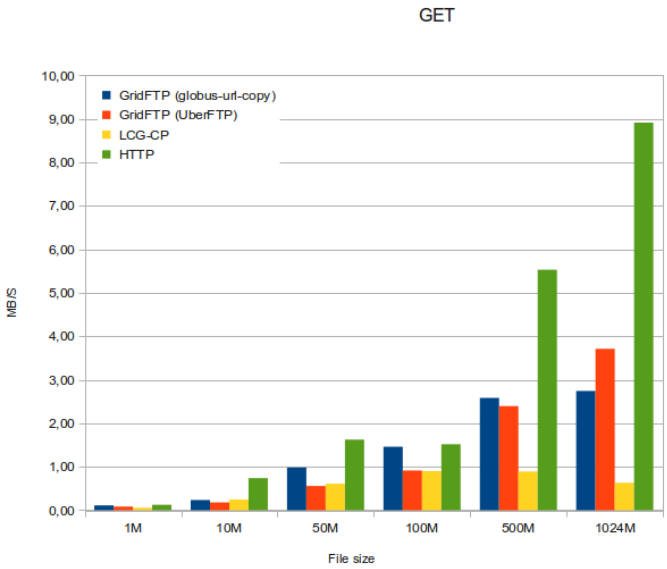


Figure 9: Downloading a file

*6.1. Transfer performance*

For the transfer tests, we have executed the transfer of different file sizes from CERN to Academia Sinica Grid Computing in Taiwan. The transfer was done through the Optical Private Network, which provides a primary link with a bandwidth of 10 Gigabits per second.

We have tested pure GridFTP transfer - no namespace operations - using globus-url-copy and UberFTP in active mode, GridFTP through lcg-cp - which registers the file in the namespace first - and HTTP, which also registers the file in the namespace before the transfer starts. In all cases 8 parallel streams have been used.

We can see HTTP is a good candidate for transfers, performing as well, or even better, than GridFTP. No optimisation - such as buffer or block size - for any of these combinations have been done.

Performing these tests we have also found unexpected discrepancies between GridFTP clients, as can be seen in figures 8 and 9. More work is needed for identifying their source, but our first investigations point to differences in the default values, such as the GridFTP version used.

*6.2. Metadata performance*

In addition, we have also executed some performance tests related to metadata handling: create, stat and delete. The same test set has been run through direct access to the DPNS daemon and through our WebDAV module, so we can compare. The results are presented in figure 10.

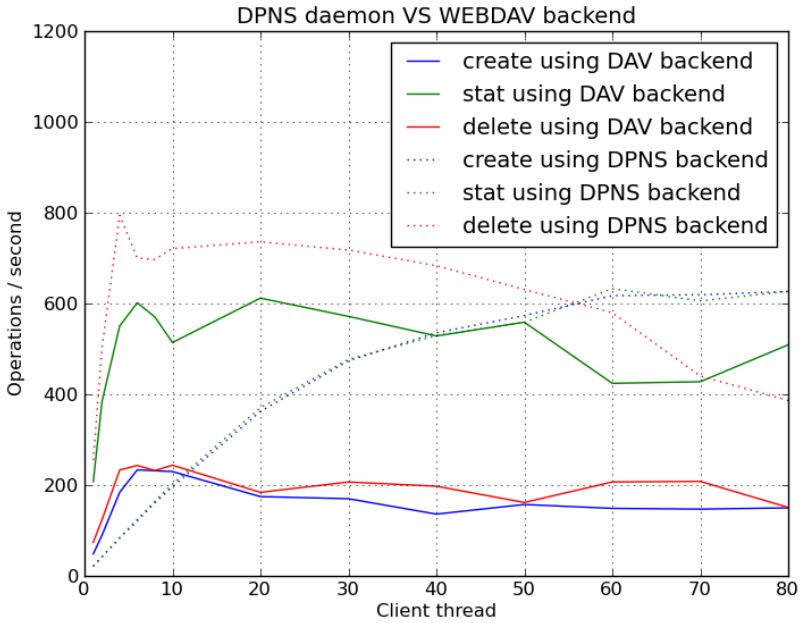


Figure 10: Metadata performance

We can see that the new implementation is even faster than traditional DPNS for the stat operation for a low number of clients. Up to a point it starts being slightly slower, but still is comparable to the performance of the DPNS daemon.

Additional tests executed running the client in two different physical machines show similar figures, independently of having only one, or both, machines running the test suite at the same time. This seems to indicate that the performance drop is due to the higher complexity of the deserialisation on the client side, rather than a problem related to the server implementation.

More and deeper tests need to be done to confirm these preliminary results.

We are also working on an extension that will use Memcached[12] in order to increase performance and reduce load on the server side.

Description of the machine where the services were running:

**CPU** Intel(R) Xeon(R) CPU L5520 @ 2.27GHz

**Cores** 4

**Memory** 11.72 GB

**DB Connection pool size** 99

**Network** Gigabit Ethernet

*6.3. Spdy*

Spdy[13] is an application-layer protocol designed to reduce latency when requesting web pages from a server. It is developed as part of the Chromium project, and it is already supported by Chromium, Chrome and Firefox on the client side, and by some Google services and Twitter on the server side. An Apache module is also available for web site usage. There is development work ongoing for curl support.

We have executed some tests with the *Page Benchmarker* plug-in for Chrome to see if this makes a difference, but the data shows the increase in the speed is very small, probably due to the simplicity of the web view, which does not benefit much from multiplexation over the same connection. Furthermore, client certificates cannot be used yet through Spdy, so only anonymous access is possible.

For these reasons the usage of mod_spdy is discouraged for now, but we will follow the project, as it has potential for improving the performance of parallel requests.

## 7. Availability

The necessary RPMs for installing a DAV front-end for DPM/LFC are available in the EMI repositories, but they are also part of the Fedora and Extra Packages for Enterprise Linux (EPEL) repositories, where we aim to have an agile and fast release cycle. We have also kept installation and configuration simple and easy, especially since our implementation builds on well known components - such as Apache 2 - but we keep some additional documentation anyway in our Trac page[14].

## 8. Future work

*8.1. DMLite*

Present work in DPM/LFC includes a refactoring effort providing a pluggable and extendable architecture[15]. WebDAV will be one of the first components profiting from this new development.

*8.2. Metalink*

We are currently investigating the possibility of providing Metalink[16] links through our WebDAV implementation. Metalink would provide a better client-driven fall-back mechanism, that, additionally, would improve download performance as different parts of a file can be downloaded from different sources.

Client support is available for all major platforms. Metalink downloads are already being provided by OpenOffice, SUSE, Ubuntu, Fedora, Linux Mint and others.

*8.3. Profiling and optimisation*
More work is needed in order to profile typical use cases. This will allow us to improve the response time and overall performance, specially for metadata operations.

## 9. Conclusions
In this paper we have presented our current HTTP/WebDAV front-end for DPM and LFC, and we have shown performance data that proves that it is a serious and reliable alternative as an access and transfer protocol. In addition, as a well established standard, it can be used with multiple third party clients, some of them already familiar to most computer users, making access the data stored on the Grid easier.

Our current implementation is ready for production, and it is already available in EPEL 5 and 6, where we aim to have frequent updates with fixes and new functionalities, while keeping a high level of interaction with our users.

## Acknowledgement
This work was partially funded by the EMI project under European Commission Grant Agreement INFSO-RI-261611

## References
[1] Abadie L, Badino P, Baud J P *et al* 2007 Grid-Enabled Standards-based Data Management *24th IEEE Conference on Mass Storage Systems and Technologies*, pp. 60-71
[2] Allcock W 2003 GridFTP: Protocol extensions to FTP for the Grid
[3] Fielding R, Gettys J, Mogul J, Frystyk H, Masinter L, Leach P and Berners-Lee T 1999 RFC 2616: Hypertext transfer protocol – HTTP/1.1
[4] Goland Y, Whitehead E, Faizi A, Carter S and Jensen D 1999 RFC 2518: HTTP Extensions for distributed authoring – WEBDAV
[5] Ernst M, Fuhrmann P, Gasthuber M, Mkrtchyan T, Waldman C 2001 dCache, a distributed storage data caching system *8th IEEE/ACM International Conference on Grid Computing*, pp. 243-250
[6] Tsigenov O and Baranova T 2011 WebDAV in dCache
[7] Maeno T, De K, Wenaus T, Nilsson P *et al* 2010 Overview of ATLAS PanDA Workload Management *J. Phys.: Conf. Ser. 331 072024*
[8] Van der Ster D, Elmsheuser J, Ubeda M and Paladin M 2011 HammerCloud: A Stress Testing System for Distributed Analysis *J. Phys.: Conf. Ser. 331 072036*
[9] Devresse A 2011 GFAL Evolutions and gfalFS *3rd EMI All-Hands Meeting*
[10] The ROOT project: `http://root.cern.ch/`
[11] libcurl: `http://curl.haxx.se/libcurl/`
[12] Memcached: `http://memcached.org/`
[13] The Chromium Projects, SPDY: An experimental protocol for a faster web
[14] LCGDM-DAV Trac page: `https://svnweb.cern.ch/trac/lcgdm/wiki/Dpm/WebDAV`
[15] Alvarez A, Beche A, Furano F, Hellmich M, Keeble O and Rocha R 2012 DPM: Future Proof Storage *International Conference on Computing in High Energy and Nuclear Physics*
[16] Bryan A, Tsujikawa T, McNab N and Poeml P 2012 The Metalink Download Description Format