CERN's AFS replacement project

To cite this article: J Iven et al 2017 J. Phys.: Conf. Ser. 898 062040

View the article online for updates and enhancements.
CERN’s AFS replacement project

J Iven, M Lamanna and A Pace
CERN - European Organization for Nuclear Research, 1211 Geneva, CH
E-mail: jan.iven@cern.ch

Abstract. OpenAFS is the legacy solution for a variety of use cases at CERN, most notably home-directory services. OpenAFS has been used as the primary shared file-system for Linux (and other) clients for more than 20 years, but despite an excellent track record, the project’s age and architectural limitations are becoming more evident. We are now working to offer an alternative solution based on existing CERN storage services. The new solution will offer evolved functionality, and is expected to eventually benefit from operational synergies. In this paper we will present CERN’s usage and an analysis of our technical choices: we will focus on the alternatives chosen for the various use cases (among them EOS, CERNBox and CASTOR); on implementing the migration process over the coming years; and the challenges and opportunities of the migration.

1. Overview
CERN has been using the Andrew File System (AFS) for over 20 years, and has built much of the local computing infrastructure around it. This system needs now to be phased out. While no single drop-in replacement has been found, the majority of use cases will be migrated to CERN’s EOS storage system. The migration will be done together with the CERN user community in a complex multi-year project, but also presents a chance to revisit many of the old computing workflows and adapt them to modern tools.

The paper starts with a brief overview of AFS and its usage at CERN. We explain the reasons for our decision to migrate away from it, highlight the expected challenges, and present the migration process. Afterwards the systems that will replace AFS are introduced. We finish with a discussion on the risks and opportunities created by the migration, in particular the chance to set the direction of computing at CERN for the years to come.

2. Introduction to AFS
AFS is a distributed global shared filesystem originally developed at Carnegie Mellon University in 1983. After having been spun off into Transarc (1989) and later purchased by IBM in 1994, the current software has been open-sourced in 2000 under the “IBM Public license” and is maintained since by the OpenAFS project – ([1], [2]).

AFS has a scalable architecture - the file system tree (data and metadata) is split into “volumes”, which then can be distributed over different machines, with optional read-only replication. It also features built-in security based around Kerberos, a truly global namespace as well as advanced client-side caching. While not a full POSIX-compliant filesystem, AFS implements enough of the semantics such that most applications work unchanged.
3. (Open)AFS @ CERN

AFS and later OpenAFS have been used at CERN since 1993. While the service is small in comparison to other storage services at CERN (see [3] for an overview), it is in the top-10 of AFS cells worldwide, by data and capacity [4]. Some key figures that characterize the CERN AFS service:

- 35k users (5k active/day),
- 450 TB of data,
- 3.8x10^9 files/directories,
- 3.5x10^9 accesses/day

More details and information on the CERN AFS service evolution over time can be found in [5]. The AFS space at CERN is split by allocation (personal vs shared) into different areas:

- 35k personal home directories (2-10 GB each),
- 8.5k personal workspaces (10-100 GB) - split from the above to contain overloads from batch computing,
- 500 shared project areas (1 GB-10 TB quota for the whole project, split into several volumes, for a total of 37k volumes),
- group shell environments (typically below 1 MB - small and few, but highly replicated since used by every login)

Originally these areas used to be hosted on different quality of hardware. Nowadays a common hardware type is used, although home directories and some critical projects are still on redundant power and UPS.

AFS space at CERN is primarily used for:

- Per-user content (analysis, scripts, documents, etc),
- Software development (project work, engineering, experiments),
- Batch input/programs/output area,
- Software distribution (ready-to-run binaries and libraries, as well as tarballs),
- Websites (3.6k personal, 600 project sites)

In addition, CERN’s AFS space has also been used for experiment data, to hold temporary files, and as a quick “backup” solution for other services. Occasionally some external sites collaborating with CERN uses it to provide home directories to their users. It is also used to support complex workflow state machines, and - as all storage systems beyond a certain age - it does contain lots of obsolete data.

4. Why replace?

Despite the impressively long lifespan and scalable architecture, AFS has several long-standing issues, e.g.

- the split of data+metadata into “volumes” as unit of placement allowed for easy scale-up, but at the same time limits the load within a single volume to what a single server can handle.
- The volume is also the unit of replication (for load sharing and higher availability), but only once marked as read-only. However, users often would like concurrent writes to within the same directory tree, or a mix of read- and write-activity.1

1 Proposals for better-scaling read-write volumes have been made but have not found their way into OpenAFS: read-write replication [6] [7], object storage [8] [9]
• the AFS-3 protocol is tied to a variant of the obsolete DES encryption standard, as well as to IPv4-only addresses. Efforts have been made to address both issue via protocol updates [10], these are essentially blocked. Both issues have non-technical repercussions.²

• the AFS-internal RPC protocol (“RX”) has limited window size and does not work efficiently over “long fat pipes”. This is already very noticeable between the two CERN computers centers (22ms apart): transactions that take minutes on the LAN take hours there, to the point that many volumes are deliberately not being replicated to the second computer center.

Overall OpenAFS appears to be in slow decline and losing mindshare.

This is confirmed by various metrics such as a slowing release cycle, decreasing commits/month and number of programmers (figure 1). While a reduced change rate might be taken as a sign of project maturity, one would expect the number of messages on the support mailing list to stay steady (or increase) if new sites were to install the software, but the opposite

² IPv6 affects US government funding; DES lost its status as FIPS standard which affects domain-specific regulatory compliance
is the case. The AFS & Kerberos workshops used to be a yearly event (and for a time there even were two per year), but have become infrequent.³

The announced release dates for new feature versions are slipping,⁴ although the OpenAFS maintainers still provide timely security updates and occasional bugfix releases.

Any major code cleanups or rewrites are hampered by the need to stay compatible with the original Transarc AFS version (for trademark reasons). In 2012 the development effort split between open-source OpenAFS and a commercial fork (AuriStor), ultimately reducing the activity on OpenAFS itself.

We also observe increasing difficulties on the client side for modern operating systems (such as MacOS X, Win10, Linux4.x). As the OpenAFS client is implemented as an out-of-tree kernel module,⁵ it requires constant work in order to adapt, including to ever-tighter security restrictions for kernel-level code. The increasing lag between the release of an operating system update and the corresponding OpenAFS support indicate lack of effort and hence reduced interest by the user community.⁶

In conclusion, the overall investment (in terms of available effort or generated support contracts) from the user community appears insufficient to sustain the project in the long run.

For these reasons, CERN started to look for alternatives and in 2015 decided to phase out OpenAFS.

Since the OpenAFS project is still functional, such a transition is not under immediate time pressure, and a multi-year transition period has been decided upon. The CERN accelerator schedule (which sets priorities for CERN activities) foresees a maintenance and upgrade period in 2019-2020 ([12]). During this upgrade, even former critical AFS use cases can be safely migrated. From LHC Run-3 onward (foreseen to start in 2020) CERN should no longer depend on AFS.

5. Challenges

Due to the long operation period of AFS at CERN, the local computing infrastructure is very much built around it: users (and experiments) are used to a relatively seamless software development working environment from the local desktop, to a central interactive Linux cluster (“LXPLUS”), to the CERN batch service (“LXBATCH”), using a common operating system (SLC/CentOS). With some users opting for incompatible operating systems on their desktop or laptop, the LXPLUS cluster remains the reference platform for both local batch farms and the worldwide grid computing (WLCG), and it relies itself on AFS as shared filesystem.

Implicit challenges also come from the qualities of the AFS product itself - AFS is excellent for many use cases, in particular for remote access to small files. In addition, it is freely available and open source, which means no onerous license restrictions on deployment.

No obvious single drop-in replacement for AFS has so far emerged. As such the migration is not a voluntary one to improve general performance or user experience, but rather imposed by external constraints, into replacement systems that have just to be “good enough” (although with the hope of user-visible benefits in certain areas).

³ In 2017, the website[2] still points to the 2014 and 2015 events.
⁴ Most code committed to the “master” branch in 2012 has not found a way yet to a production release in 2017 - this would eventually become OpenAFS-1.8.
⁵ At least on Linux this appears to be due to the license chosen by IBM.
⁶ There are no OpenAFS binary builds anymore for Windows since 2014; the last two MacOS releases each took several months until they were supported; support for 4.X Linux needed dropping a major performance optimization.
6. Phaseout process
The phaseout is a complex multi-year process, and communication and use case discovery play a major role in the project.

![Phaseout timeline](image)

Figure 2: Phaseout timeline (in 2017)

To ensure proper communication with the various user communities, CERN departments and experiments were contacted early and invited to identify “AFS phaseout coordinators”. The coordinators were also tasked with discovering and classifying (specific vs common) use cases in their respective areas, and to act as consultants towards their own users.

The actual migration of the AFS data is driven by the users: they first need to assess whether the data needs to be kept at all and select the appropriate target storage system, copy the data and translate the AFS ACLs to the new system. Then they need to verify that the data in the new location is actually usable, and finally delete from AFS. Due to the potential effect on crucial activities, a behind-the-back migration of the data is generally not possible.

The phaseout progress itself is tracked via JIRA tickets both for all major branches of the CERN AFS tree, as well as for identified common use cases.

A series of external disconnection tests is foreseen to flush out cases where CERN’s AFS is relied upon by external institutes that cannot be identified from logs (due to low access rates) or contacted via community communication channels (such as HEPiX or WLCG).

7. Replacements
The main storage systems proposed for data migrations out of AFS are EOS (with CERNBox) and CVMFS:

EOS is the main CERN disk system, it is heavily used for LHC data and currently at over 100 PB of capacity [13]. EOS is accessible via many protocols (HTTP, ROOT, GridFTP, SRM, SMB...), including as a filesystem (FUSE mount, e.g on LXPLUS, LXBATCH).

We expect EOS to be the main migration target from AFS for user and project files. We propose that LXPLUS (interactive access) and LXBATCH services will also use EOS as a mounted home directory. Nevertheless we expect desktops and laptops to largely opt for a local home directory and CERNBox (see below).

Synthetic tests and early user feedback showed that EOS-FUSE performed worse than AFS in several areas (in particular for small files). Currently (June 2017) it is being rewritten to address such issues. Similarly, the EOS namespace would not be able to hold all CERN AFS files, and is therefore being extended [14].
EOS currently is not backed up to tape, but provides resiliency against unwanted changes and deletions thanks to file versioning and an “trashcan”-style deferred deletion. It copes much better than AFS with concurrent access to a single tree, and last but not least allows to expand the per-user storage into the TB range. The EOS ACL system is at least as expressive as AFS ACLs, and is integrated with CERN’s “E-Group” [15].

As a recent service addition implemented on top of EOS, CERNBox is a customized ownCloud service that provides advanced file sharing (via a web interface) [16]. The CERNBox client also allows disconnected access to a subset of files (via asynchronous synchronization). The service has so far been very popular at CERN, including for many non-AFS users.

We expect shared software to move from AFS to CVMFS (the CERN virtual machine filesystem [17]). Nowadays primarily used to deliver read-only content to massively-parallel access (such as experiment binaries and libraries to Grid worker nodes), this has been already adopted by the LHC experiments for software delivery before the AFS phaseout came up. We also plan to use CASTOR (the CERN tape archive) in the migration, since it provides reliable and cost-efficient storage for rarely-accessed data. [18]

AFS data will be classified by the user into several categories, based on expected future use. This classification then gives the most suitable migration target:

- shared data, high chance of usage in next months: migrate (as individual files) to EOS.
  - specific for AFS-based websites: a new EOS-based web service has been set up in 2016.
  - specific for code (and CVS repositories): migrate to central CERN version control.
  - specific for executables and libraries accessed from batch nodes: migrate to CVMFS.
- (mostly) non-shared, high chance of usage in next months: keep on local disk, optionally synchronize and share via CERNBox. In particular user files (Office documents, GUI configuration etc) would be in this category.
- unused now, low chance of future usage: store (as archive file) in CASTOR.
- some special use cases will be moved onto other CERN IT storage services, such as cluster filesystems, e.g. non-kerberized NFS, CephFS or HDFS or (with application changes) might best be suited to a database.
- lastly, we also expect some cleanup of unused and unwanted AFS data.

8. Risks vs Opportunities
A major upheaval of the computing environment clearly is risky – from short-term impact on productivity, to accidentally-deleted data, to undetected data loss or corruption.

It also carries direct costs such as time spent reviewing and migrating, adapting tools, and - since the migration process is driven by the AFS users - much of these costs will be borne by them.

Other risks related to the AFS phaseout (and corresponding mitigations) are:

- Failure to analyze existing use cases: a mismatch between old AFS capabilities and the new replacement system(s) that cannot be overcome by sufficiently fast changes (either to the workflow, or to the new system) will stall the project and may even slow down phaseout in technically unaffected areas. Hence the early discovery of such cases is crucial to the project, as well as careful selection of the migration target.
- missed window of opportunity: CERN has decided on a multi-year migration relying on users to do most of the work, but users might shy away from the risk of disruption, or treat...
the issue at (too) low priority, which might lead to AFS being still used in critical roles when workflows get frozen for LHC Run 3.

However, the realistic timescale and coupling to major CERN milestones will allow for careful inspection of the old data and use cases, to perform exhaustive tests in the new system, and to schedule the migration to coincide with a “naturally-occurring” software or system update so as to minimize additional effort and downtime. In the mean time, regularly providing (and asking for) status updates will ensure that the phaseout is not “forgotten”. Eventually the unavailability of AFS clients for modern operating systems will impose a natural endpoint.

The migration comes with an important benefit: it is a major opportunity to innovate the way the CERN community does computing, and a natural point to introduce disruptive technologies.

Together with our user community, we treat the migration not as just the immediate move of data from system A to system B, but rather focus on the workflows around it – with limited extra effort perhaps these workflows can be improved (more efficient, better scaling, less manpower-intensive, ...). Some examples of use cases that may benefit from such an analysis:

- For batch computing, “sandboxing” (where input and output are specified upfront, and can be copied via explicit operations) means much better scaling and efficiency, and allows to break-out to (opportunistic) cloud compute capacity (i.e. resources that do not “look like LXBATCH”).
- A clean split between the "desktop" and the "compute environment" allows to use fast-changing modern operating systems on one, while providing a stable production platform on the other.
- Improved support for mobile devices: the disconnected operation (on a subset of data) via CERNBox allows to use continue work while not connected to a network.
- For physics analysis, web-based “workbooks” (as provided by the SWAN service [19] from EOS-FUSE) offer an attractive integrated interface for documents and computing, as well as easy sharing (via CERNBox), beyond the traditional “shell + file system” approach. Together with open-access datasets, these allow full interactive reproduction of results, and are set to become a powerful tool for analysis, teaching and outreach.
- tools like virtual machine images, containers, continuous integration services and out-of-tree builds are all innovative approaches to improve the computing experience for the CERN users, while reducing the need for a shared filesystem.

In conclusion, we believe that the AFS replacement project - while undoubtedly imposing a cost on the CERN user community - has strategic value in shaping the way CERN computing will evolve for the next 15 years.
References

[1] Wikipedia 2017 Andrew file system — wikipedia, the free encyclopedia [Online; accessed 29-May-2017] URL https://en.wikipedia.org/w/index.php?title=Andrew_File_System

[2] OpenAFS project web site [Online; accessed 29-May-2017] URL http://openafs.org/

[3] Mascetti L et al. 2015 Journal of Physics: Conference Series 664 042035 URL http://stacks.iop.org/1742-6596/664/i=4/a=042035

[4] Altman J 2015 AFS deployment rankings, private communication

[5] van der Ster D, Moscicki J T and Wiebalck A 2014 Journal of Physics: Conference Series 513 042046 URL http://stacks.iop.org/1742-6596/513/i=4/a=042046

[6] Satyanarayanan M, Kistler L, Kumar P, Okasaki M, Siegel E and Steere D 1990 IEEE Transactions on Computers 39(4)

[7] 2008 OpenAFS project proposals [Online; accessed 29-May-2017] URL http://www.openafs.org/uiuc_capstone.html

[8] Reuter H 2008 AFS + object storage, talk at OpenAFS workshop 2008 URL http://workshop.openafs.org/afsbpw08/talks/thu_3/OpenAFS+ObjectStorage.pdf

[9] Reuter H 2014 AFS/OSD current state and future, talk at European AFS and Kerberos Conference 2014 URL https://indico.cern.ch/event/271400/contributions/1610363/attachments/487822/674360/AFS-OSD-state-and-future.pdf

[10] AFS3 standardization group [Online; accessed 29-May-2017] URL http://afs3-stds.central.org/

[11] OpenAFS project on BLACKDUCK Open Hub [Online; accessed 29-May-2017] URL https://www.openhub.net/p/openafs

[12] Longer term LHC schedule [Online; accessed 29-May-2017] URL http://cern.ch/lhc-commissioning/schedule/LHC-long-term.htm

[13] Peters A and Janyst L 2011 Journal of Physics: Conference Series 331 052015 URL http://stacks.iop.org/1742-6596/331/i=5/a=052015

[14] Peters A, sindrilaru E, Adde G and Lensing P 2016 Journal of Physics: Conference Series

[15] CERN E-Groups [Online; accessed 29-May-2017] URL https://egroups.cern.ch/

[16] Mascetti L et al. 2015 Journal of Physics: Conference Series 664 062037 URL http://stacks.iop.org/1742-6596/664/i=6/a=062037

[17] Blomer J, Aguado-Sánchez C, Buncic P and Harutyunyan A 2011 Journal of Physics: Conference Series 331 042003 URL http://stacks.iop.org/1742-6596/331/i=4/a=042003

[18] Presti G L et al. 2007 24th IEEE Conference on Mass Storage Systems and Technologies (MSST 2007), 24-27 September 2007, San Diego, California, USA pp 275–280 URL http://doi.ieeecomputersociety.org/10.1109/MSST.2007.7

[19] Piparo D et al. 2016 Future Generation Computer Systems (Preprint https://cds.cern.ch/record/2158559/files/CERN-OPEN-2016-005.pdf)