CERN AFS phaseout: status & plans

Jan Iven[1,*] and *Alberto* Pace[1]

[1]CERN - European Organization for Nuclear Research, 1211 Geneva, CH

Abstract. In 2016, CERN decided to phase out the legacy OpenAFS storage service due to concerns for the upstream project's longevity, and the potential impact of disorderly service stop on CERN's computing services. Early 2019, the OpenAFS risks of the project collapsing have been reassessed and several early concerns have been allayed. In this paper we recap the work done so far, highlight some of the issues encountered, and present current state and planning.

1 Introduction

AFS [1] is a shared secure filesystem with global namespace, and OpenAFS [2] is its major implementation. CERN decided in 2016 to phase out its AFS service, over a timeframe of 5 years, after successfully running it for over 20 years. That decision was taken due to a high perceived risk of a fast upstream project collapse at an inconvenient moment, i.e with CERN still heavily relying on AFS in the middle of a Large Hadron Collider (LHC) run. For full details on the background and original motivation of this phaseout decision see [3]. At the time, no single drop-in replacement had been identified. Instead, based on the various use cases, the data was to migrate to a set of different services (or, in some case, to be deleted). In case where this was not straightforward, these other services would have to be adapted, and the migration might also require user-side changes to their workflows.

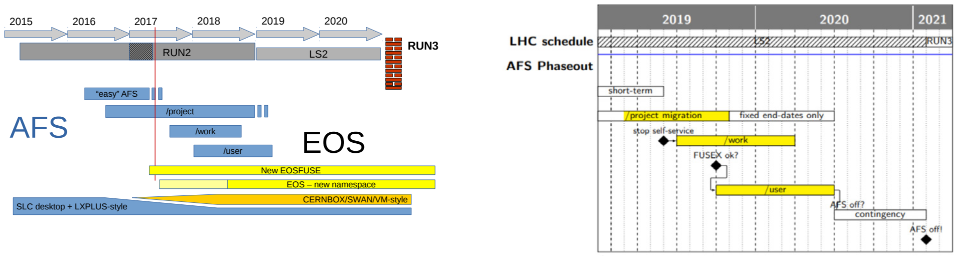
The CERN AFS space was split into different areas: personal storage space (separate homedirectories and workspaces) and shared storage for projects or experiments (with delegated quota and volume administration). There was a wide variety of use cases on AFS. Some of the major ones were personal documents/scripts/configuration, software development and distribution, batch system storage (logs and input data) and physics data (in particular for smaller and older experiments). Also, many websites were hosted there. The major migration options initially foreseen for the AFS phaseout were CASTOR for archiving stale (but important) data, CVMFS for (large-scale) software distribution, EOS [4] for data that needed to be kept accessible, and finally deletion, for clearly-identified obsolete content.

In the following, we will compare the original phaseout planning with reality, highlight some of the issues encountered and the in-depth investigations triggered by the phaseout, summarize the current state of AFS at CERN and provide a brief outlook.

1.1 Original planning

The initial phaseout plan had proposed to migrate away most non-critical use cases already during LHC RUN2 (i.e before end of 2018), and address any leftovers and run-critical use

[*]e-mail: jan.iven@cern.ch



(a) in 2Q2017[3]

(b) in 1Q2019

Figure 1: Evolution of the phaseout plans.
Note e.g the shift for workspace migration from mid-2017 (1a) to mid-2019 (1b)

cases during the subsequent LHC "long shutdown" (LS2) in 2019. This was admittedly somewhat aggressive, but provided ample contingency during the LS2, and still foresaw fully stopping the AFS service at CERN before LHC RUN3, i.e before 2021 - see Fig.1a.

This was to be achieved through a coordinated CERN-wide effort, with formal representation of all affected CERN user groups and experiments. At the time, no single ready drop-in replacement for AFS had been identified, but many "live data" use cases (non-archival, non-software deployment) were expected to migrate to EOS, which had already a filesystem interface. The phaseout plan had identified users' home directories as well as the batch computing at CERN as being particularly challenging – it foresaw an eventual need for 'compute' to be less reliant on a shared filesystem, but hoped for a natural technology-driven evolution towards 'sandboxing', 'cloud'-like access and new frontend interfaces such as web-based notebooks.

2 Progress made

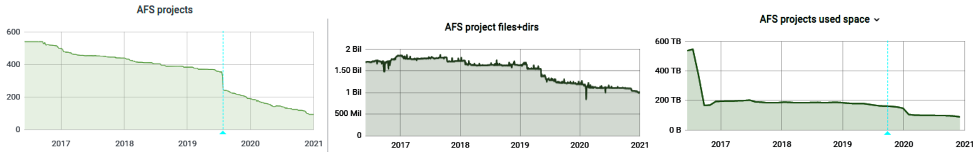
2.1 Project areas

The archival of unused AFS data was largely non-controversial, and archival speed was determined by available human effort. Many project areas were still being used to some degree, and often contained websites which needed to stay available. Identifying old contacts and potential new owners, assessing current usage and curating content before archival were quite slow, but steady progress was being made, as can be seen in Fig.2a[1]. While many AFS project administrators indeed diligently reviewed and migrated their AFS project data, some only reacted once their data had been archived (which had to be reverted in a few cases). Of course, a number of project areas - including the largest ones - were (and are) actively used, and in some cases could only be partially archived or removed. This is reflected in Fig.2b - while the decline in file count is less impressive than for project areas, the next Fig.2c shows the effect of (huge) data files being archived.

2.2 Software distribution

Likewise, the shift of experiment-wide software distribution from AFS into CVMFS had already started when an AFS phaseout had been discussed, so was "agreed upon in principle", and was largely driven by the user community itself. However, their focus was to make the newest software releases also available on CVMFS, and did not necessarily include the cleanup of the previously-used AFS space. This then led to parts of the user community

[1] the sudden drop in project count in July 2019 is due to dropping already-empty areas from a cache



(a) number of AFS project areas (b) file+directory in projects (c) projects, used space

Figure 2: AFS project area phaseout over time

still relying on AFS, even if the majority of an experiment had switched. Also here, the main challenge was separating ongoing use of AFS by "inertia" from actual hard (and non-covered) software-related use cases. For one major software distribution area (for the LCG software), the timeline for actual removal had to be extended several times to allow for ongoing publications to continue the use of legacy software releases only available from (and intimately tied to) AFS.

Nevertheless, for some use cases, AFS actually turned out to be the preferred solution for software. In particular, the distribution of commercial (engineering or simulation) software with "site licences" mapped well onto AFS' optional IP-based access control, which neither CVMFS nor EOS supported.

2.3 Investigations

Part of the AFS migration were efforts to better understand the actual use of AFS – after so many years as a service, and over generations of users, it was not clear even to the data owners what exactly was going on. Some of the use cases were raised upfront by the nominated phaseout representatives, others came from investigating current content or access patterns from within the service itself, others only got raised when a particular area was made inaccessible. Some examples of the issues looked at are presented below.

2.3.1 External dependencies into CERN AFS

A particular worry for a phaseout was the potential impact on other sites - the HEP community has grown together over the years, and relied in some cases on each others' services. In particular, CERN was seen as informal (i.e not governed by MoU) storage service provider to the HEP community. Efforts were made to make the HEP community aware of the AFS phaseout at CERN and to contact external institutes upfront (via WLCG communication channels and the HEPIX conferences). In order to raise awareness (and get in touch with soon-to-be-affected institutes), CERN performed two "external disconnection" tests - a single-day test in 2017, and a full week in 2019. The first test very quickly identified several institutes that had configured their machines to use CERN's AFS as homedirectory. The second test found no major external dependencies anymore.

2.3.2 File types found on AFS workspaces

In early 2019, the only AFS area at CERN that still showed growth were personal AFS workspace - also evident in Fig.6a below. An investigation showed that ROOT data files made for 47% of the data volume found there, in particular by users from the two largest LHC experiments at CERN. This was somewhat surprising, since EOS was acknowledged to be best suited for such data, and individual users easily could get 1 TB quota (i.e 10× the AFS quota) there.

As a consequence and to counter this trend, the AFS service - in collaboration with experiment computing experts - started to ask users from these experiments to justify their request for AFS workspaces, and encouraged local batch users to place at least their "big" data files on EOS. This exercise unearthed some outdated documentation, tutorials and "community lore" that still assumed AFS, but also showed that some use cases (e.g recompiling parts of the experiment framework software) indeed still required AFS. Nevertheless, the growth of AFS workspaces has since indeed flattened out.

2.3.3 Left-behind files in AFS

In 2020, the question was raised whether users perhaps had de-facto moved to EOS, i.e copied their data and used it from EOS, and might just have "forgotten" the data copy left behind on AFS (as there was no real incentive to remove this). However, a study found that there was little overlap in filenames on the two systems (about 1%, including duplication from "common" filenames), and also found no clear indication of an overall shift of activity by the users (as reflected in file modification times). Instead, the majority of users were using just one of the systems, even when they had accounts on both, and most others use both systems in parallel - their long-term activity pattern is the same on both systems.

3 Issues encountered

3.1 EOS-FUSEX

It was already clear at the beginning of the migration that while the initial EOS filesystem interface provided a convenient ad-hoc access to large physics datafiles, it was not up to the variety of usages that the AFS users subjected it to. After some initial attempts of iterative improvement of the existing code, a decision was taken in 2016 to fully rewrite this component, with new server-side interfaces and the addition of a client-side caching layer [5]. Latency reduction and a more POSIX-compliant interface (in line with what AFS provided) were the main design goals.

The rewritten filesystem indeed solved many of the issues reported by the users. However, for some usecases, in particular version control and compilation with many small (and temporary) files, EOS-FUSEX was much faster than before but still was on average 2-3 times slower than AFS. This is partly due to the EOS architecture (data goes to two machines, and metadata still elsewhere), and this performance discrepancy has largely been accepted for general use, as it brings a significantly increased scalability. Services that are highly affected (such as continuous integration, build services) typically have moved to local builds (since AFS also is already markedly slower than a local SSD).

While originally foreseen for roll-out still in 2017, the inherent complexity meant that a first 'closed beta' version was made available in May 2018, and first production data was made accessible via the new filesystem in Oct that year. Testing of nontrivial AFS use cases against the new code could then resume, but based on data location, either the 'old' or 'new' EOS fileystems were used, so some testers repeatedly encountered issues they had already reported.

Another unforeseen issue was that since 2014 CERN had started to use a secondary data center in Hungary, and part of the disk capacity was installed there – this arrangement ended in 2019, and parts of the storage hardware for EOS had to be brought back to CERN. Whereas for physics data the additional latency (22 ms one-way) for an access from CERN to Hungary (or vice-versa) was not a major issue, and for reading EOS preferred a 'close' file replica, this latency was very noticeable for writing, and risked overshadowing the latency improvements gained from FUSEX. In order to prevent confounding FUSEX and remote-access latency issues, it was decided to only enable the new protocol on EOS instances that were again entirely hosted on the CERN campus, after all remote hardware had been drained, shipped back to CERN and reinstalled. This procedure further delayed the FUSEX roll-out, and only in 2020 were all EOS instances finally accessible via FUSEX.

3.2 EOS namespace

The EOS namespace as used for the first (physics data) usecases had been in-memory (with transaction logging to stable storage). While this provided for excellent latency at runtime, loading the data from disk on restarts made for noticeable outages. The available memory also made for an obvious limit on the number of entries (files+directories) a single EOS instance could hold, and at sizes of around $250 \cdot 10^6$ entries even large-memory machines (512 GB) began to approach their limits. At the same time as the FUSEX rewrite above, a new disk-resident EOS namespace ("QuarkDB"), based on RocksDB, was designed and implemented [5], [6].

This was a race against time – fast uptake by the users of CERNBox, a collaborative tool based on EOS, lead to an ever-increasing number of entries in the underlying "old" EOSUSER instance ($600 \cdot 10^6$ in mid-2018, by then with 1 TB of RAM) - see Fig.3 (from [7]).

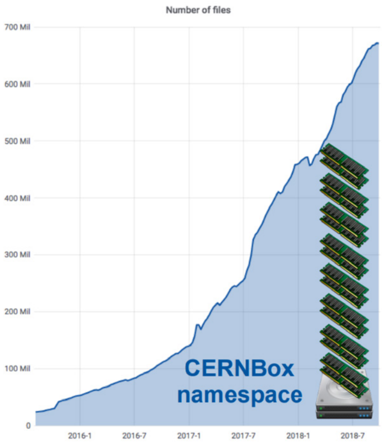


Figure 3: CERNBox namespace growth

The AFS migration would have potentially added up to $4 \cdot 10^9$ more entries - far beyond the capacity of the old infrastructure to absorb. This essentially put a stop on any large-scale data movement from AFS into EOS, until the new namespace was in place.

Mid-2018 the deployment of the new code started, through an EOS-internal migration to a new service layout, and in January 2019 most personal user data had been moved [8]. EOS project data was still being moved later in 2019, and as such non-trivial AFS project migrations were further delayed. The newly-structured service then received even more data from an (initially unforeseen) corner: as part of an effort to contain licence costs [9], user data has been migrated out of Microsoft DFS since 2018 [10], and (together with the already-existing CERNBox sync-and-share) these non-Linux use cases now shaped the EOS service and policies, and further increased the need for operational stability.

3.3 $HOME

A particularly challenging use case for AFS is the homedirectory on central CERN IT services. Not only did this have high requirements for availability and reliability, as a shared filesystem with strong authentication this also required tight integration with the machine authentication stack (also a continuous issue for AFS itself). Early attempts to integrate the initial EOS-FUSE were not addressing all possible login scenarios, but EOS-FUSEX eventually solved these.

However, a homedirectory needs to have certain files (configuration data, secrets, caches etc) in well-defined places in order for them to be useful, and these are typically found at the top of the home directory tree - which on personal EOS space was the pre-configured directory for the CERNBox synchronization client. Some of these well-defined places may contain a few 100 k fast-changing files (e.g browser caches), and copying over this data from

AFS into EOS would have stopped the sync client from working. Filtering out content during the copy might have lost configuration data important to the user, and copying to a different (non-synchronized) location would have made these files not found by the application. This issue was spotted early on, but already then there were too many CERNBox users (1 k) to allow for an easy change of the synchronisation setup - and since then CERNBox usage has increased 30-fold.

This issue prevented an easy "one-step" mostly-transparent migration for user homedirectories from AFS to EOS. In addition, the self-service application that would allow users to easily switch homedirectories from one system to the other started to be rewritten in 2018, and until the new version was out, no new features were added to the old code. While some beta testers managed to change their homedirectory from AFS to EOS and reported no major issues, the size of this pilot was not large enough to be representative for all CERN users.

3.4 Workspaces and batch access

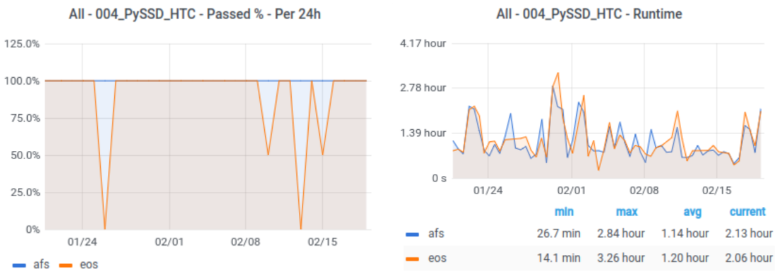
As mentioned before, CERN users can optionally request a "workspace", i.e a second personal AFS storage area, distinct from their homedirectory. The reason for this split was that the local batch system (LXBATCH) is prone to overloading the storage subsystem, and a "blocked" (overloaded) homedirectory in the past could prevent the user from working even on the local desktop machine, let alone on the shared interactive cluster (LXPLUS). As such, this workspace was supposed to be used mainly for LXBATCH. While it is in principle possible to run CERN batch jobs that do not rely on shared storage ("sandboxing"), most users prefer the convenience of having job submission file, job logs, input and output on a single area, and their internal tools and scripts often assume this. Attempts to use natural change points in the past (major operating system release, or the switch from LSF to HTCondor) to "nudge" users to use sandboxes were rejected in the name of making such transitions as transparent as possible.

The batch system also internally use shared storage, and has some stringent implicit assumptions on latency (internal files created on one machine need to be "visible" to other machines as soon as the batch system looks for them), and can itself be badly affected by slowdowns (one of the few "central submission" processes getting stuck in kernel space will affect many users and jobs).

EOS got tested early on (in 2016) as a potential replacement for the batch shared file system, but the initial EOS-FUSE implementation did not yet work here. A series of later tests highlighted other issues also in FUSEX, in particular for the stability of the batch service itself. As it was highly desirable to allow batch jobs to access data files on EOS (where most of the physics data was stored), a compromise was found: batch machine would have access to EOS, but would not allow job submission files or logs to be stored there - as a consequence, local batch users still *need* AFS.

Neither the LXBATCH service nor the EOS service look forward to change this situation - both are concerned about the impact on their stability from the other service. Batch users are also not actively pushing for a change, as this would imply removing many (hardcoded) assumptions on AFS.

In the mean time, a series of complex testcases have been created by the CERN engineering community to simulate representative workflows, and have been whitelisted to also run batch jobs, with job submission and logs on EOS (and AFS). These test have been running since mid-2019. While they have highlighted numerous issues, they also allow to quantify and compare reliability and performance of both AFS and EOS over time - see Fig.4 for an example, and ultimately will establish enough confidence that (from a user's perspective) this activity could switch from AFS to EOS.



(a) success rates over time

(b) runtimes

Figure 4: example of a complex testcase on both AFS and EOS

3.5 Incompatible policies

In some cases, the difficulties of migrating usecases from AFS to EOS were not due to eventual technical shortcomings, but arose from deliberate policies that had been established after the AFS service, and for which certain exceptions had been "grandfathered" in for AFS. As an example, those areas of EOS that hold personal or project data (and are accessible via CERNBox) have mandatory strong access control - while users can share individual files or directories (even anonymously), they cannot perform sweeping access rule modifications[2]. CERNBox is designed to (also) hold confidential data, this policy is meant to prevent inadvertent data disclosure. CERN AFS on the other hand does permits the user to enable worldwide anonymous access[3], and by default even creates a pre-configured "public/" subdirectory. AFS also allowed the use of IP-based access control, with predefined sets to cover the CERN (IPv4) networks. Both anonymous and "CERN-only" access were used with non-sensitive data, for sharing with large communities or for access by or through other services that did not themselves have access to user credentials. For use with EOS, such services would need to be running with a service identity.

A similar situation occurred on the dedicated accelerator and experiment sub-networks - these had been established in order to shield vulnerable devices (where security fixes could not always be applied in a timely fashion, e.g due to running experiments) from the general-purpose networks (with their regularly-infected desktops). They also were meant to prevent (non-agreed and possibly undesirable) runtime dependencies between experiments and IT services. As such, cross-connectivity between IT services and these networks had been reduced to a minimum – but AFS was indeed available there, and in some cases had been incorporated into the computing setup (clusters with shared homedirectories, software distribution etc). Attempts to correct this situation for EOS (and force access through gateway services, or CERNBox sync&share) lead to these use cases not working for EOS.

3.6 Start-and-stop

A non-technical problem for the migration came from the way the project was structured, and how it communicated. While the small community of nominated AFS phaseout experts had a good idea on what was already working on the replacement systems, and which usecases were still being blocked from being migrated, the same was not true for the general AFS users at CERN. Information about the various attempts to stop and restarts percolated through the CERN user community at different speeds (".. I thought we had stopped?" was

[2] `chmod -R a+rwx` is a surprisingly common user reaction to "permission denied" errors
[3] CERN AFS regularly enforces access rules on certain parts of the namespace

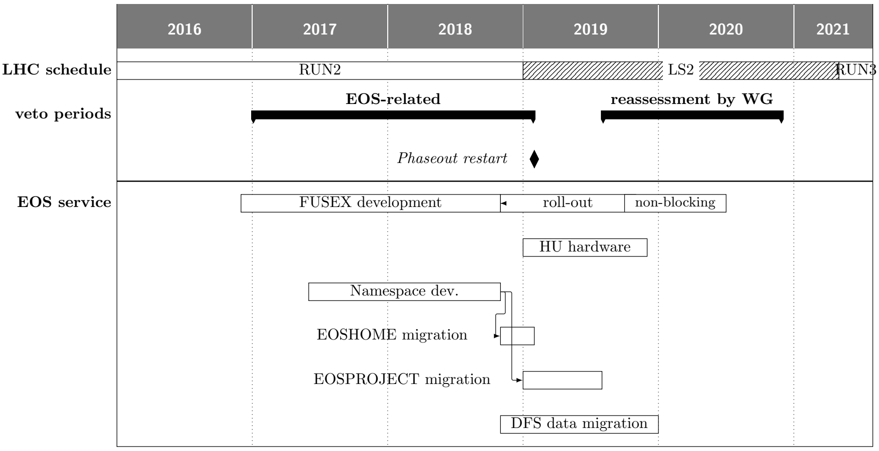


Figure 5: migration veto periods

quite commonly heard). User experience of early attempts (e.g during the EOS FUSEX beta period) also kept resurfacing, even when the underlying technical issues had long been addressed. As the roll-out of new EOS versions onto client machines and EOS instances came in stages, issues that were ’fixed’ for the development team could still be observed by parts of user community. All of the above mingled with (acknowledged) architectural EOS issues, such as higher latency on ’small file’ usecases.

In hindsight, the way the project was structured along experiments and departments was meant to qualify and assess individual use cases, but did not make for an efficient information flow to the end user. The long duration of the exercise, including periods without much activity, also contributed to a certain "AFS news" fatigue.

4 Evolution of the plan

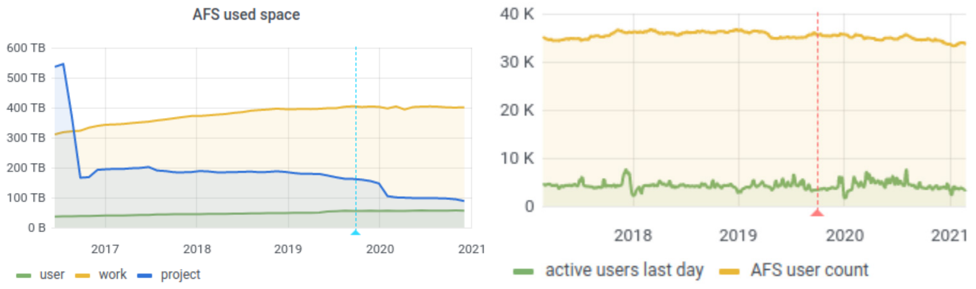
The various issues encountered during the migration had led to delays in the "live" filesystem usecases – much of the time, incoming AFS data and/or usecases related to it had been vetoed on the destination - see Fig.5.

Nevertheless, in January 2019 the AFS migration exercise was officially restarted. The schedule had started to run out of contingency, with only 2 years left until RUN3, and the most critical use cases (personal homedirectories and workspaces) still firmly on AFS. Eventually, the LHC schedule also had been pushed back - a decision in Dec 2019 set the start of RUN3 to May 2021 (and later, due to the impact of COVID-19, to 2022) - but this still would not permit a full phaseout before RUN3.

4.1 Migration reassessment

In 2019 it had become quite clear that despite our fears a few years earlier, the upstream OpenAFS project still wasn’t dead, and also regulatory compliance issues such as weak cryptography or IPv4-only had not had a major impact. While development activity had slowed down compared to 2012, it had stabilized at lower level. The project released several security and bug fixes[4], added support for new Linux kernels/distributions, and also released major version 1.8[5] before starting on the next development branch. Lastly, a rewritten Linux client

[4] including a fix for a major time-dependent issue in a single day in January 2021
[5] albeit much of the code had been contributed in 2012



(a) AFS used space

(b) AFS users

Figure 6: current AFS use

("kafs") had made it into the Linux kernel tree (without the legacy IBM public licence issue), was mostly functional and was starting to be shipped by Linux distributions. At the same time, upstream development had not gone for major architectural features or protocol-level changes, so existing operational limitations (e.g on concurrency or volume sizes) and potential compliance issues (crypto, IPv6) were still to be dealt with[6].

Nevertheless, IT started to become confident to be able to operate AFS as service during LHC RUN3, including on the next major CERN Linux operating system version.

In parallel, without a clear and imminent threat to OpenAFS stability the CERN-wide effort to migrate the remaining use cases was questioned by the user community. In particular, the LXPLUS=LXBATCH computing model relied on shared storage both internally and to provided convenient access to user data, and a migration to an alternative "sandboxing" batch model (where user data would be explicitly transferred in and out) was felt to be both expensive to implement, and not in the interest of the users.

Lastly, the issues already encountered in the migration so far made a successful complete phaseout before RUN3 unlikely.

To reassess the overall situation, a technical working group was started in autumn 2019 [11] - and until their verdict, AFS migration was back on hold (with the exception of ongoing archival, which was not contested). Their recommendations were delivered in late 2020, and

- "did not identify an urgent need to change the current home directory",
- recommended that "CERN IT should continue to support AFS at an appropriate level"

This was in line with statements made by the service since mid-2019 ("use AFS where required"). However, they also "agreed [..] to reduce the dependency on and utilization of AFS" and stated that "reducing the usage of AFS is crucial". These long-term points were due to the assumption that while OpenAFS was still coping nicely with iterative changes at the level of the operating systems, it would in the long run be unable to handle foreseeable IT-industry growth rates in compute and storage. Interestingly, again no "obvious" AFS drop-in successor was identified.

4.2 Current status and outlook

At the end of 2020, we still have 560 TB of data and 34 k users on AFS (Fig.6), both numbers have been fairly constant over the last years. Most experiments actively try to make sure that

[6] none of this should be taken as criticism of the OpenAFS project, we are very grateful for their software and efforts

the majority of users do not have to rely on AFS for daily work, except for the accepted use cases on homedirectories and local batch access.

The current situation of AFS at CERN is stable, and the dependencies into AFS have been identified. For experiment/services still remaining on AFS, those responsible have mostly made informed decisions - migration has been added to long-term work plans, and the risks accepted in the mean time. For users, a convenience component due to global filesystem-level access to their personal data remains.

Home directories, personal workspaces for use with local LXBATCH and (some) project areas will still be on AFS for LHC RUN3. At the same time the capacity of the AFS service is not expected to increase much beyond the current state, and the allocation per user is expected to stay stable as well, or to decrease. The effort to reduce experiment-level dependencies on AFS will go on, and there is ongoing adiabatic cleanup of project areas.

As the CERN local computing capacity continues to naturally grow, we expect a gradual increase in AFS overloads due to parallel batch jobs, each affecting a small subset of co-located users. It is unclear whether mitigations can be found for these overloads. Once these become more noticeable, we expect a further shift away from AFS for interactive use. Eventually either a dedicated batch-only cluster filesystem may have to be put in place, else AFS will slowly assume this role by pushing out non-batch usecases. After RUN3, the situation will have to be assessed again, but we believe that the use case for AFS will by then be much weaker.

References

[1] Wikipedia, *Andrew file system — wikipedia, the free encyclopedia* (2021), [Online; accessed 15-Jan-2021], `https://en.wikipedia.org/w/index.php?title=Andrew_File_System`

[2] *OpenAFS project web site*, [Online; accessed 15-Jan-2021], `http://openafs.org/`

[3] J. Iven, M. Lamanna, A. Pace, **898**, 062040 (2017)

[4] A. Peters, L. Janyst, Journal of Physics: Conference Series **331**, 052015 (2011)

[5] E.A. Sindrilaru, A.J. Peters, G.M. Adde, D. Duellmann, Journal of Physics: Conference Series **898**, 062032 (2017)

[6] Bitzes, Georgios, Sindrilaru, Elvin Alin, Joachim Peters, Andreas, EPJ Web Conf. **214**, 04019 (2019)

[7] X. Espinal Curull, *IT-User meeting 25: Brief updates on IT services* (2018), `https://indico.cern.ch/event/736088/contributions/3036289`

[8] L. Mascetti, M. Lamanna, E. Karavakis, J. Moscicki, H.G. Labrador, A.J. Peters, *Migration of user and project spaces with EOS/CERNBox: experience on scaling and large-scale operations*, poster at CHEP2019 (2019), `https://indico.cern.ch/event/773049/contributions/3474465/`

[9] *CERN MALT project*, [Online; accessed 04-Feb-2021], `https://cern.ch/malt/`

[10] G. Lo Presti, S. Bukowiec, L. Mascetti, H.G. Labrador, V.N. Bippus, A. Smyrnakis, M. Kwiatek, J. Moscicki, *CERNBox as the hyper-converged storage space at CERN: integrating DFS use-cases and home directories*, poster at CHEP2019 (2019), `https://indico.cern.ch/event/773049/contributions/3474470/`

[11] X. Espinal, J. Moscicki, A.J. Peters, A. Wiebalck, D. Van der Ster, Tech. Rep. CERN-IT-Note-2021-001, CERN, Geneva (2021), `https://cds.cern.ch/record/2750122`