AFS/LSF Phase Out personal experience

R. De Maria

Use cases for AFS and LSF

OpenAFS is used in ABP for:

1. Working directory when running in Lxplus and Scientific Linux distributions.
2. Storage space for simulations campaigns with LSF and Boinc (LHC@Home).
3. Sharing (anonymously and with access control) development and production files (data and applications) within CERN (GPN and TN) and outside CERN.
4. Hosting websites.

LSF is used in ABP for:

1. Simulation campaigns by teams and individuals. Decades of simulation and analysis submission jobs accumulated and time consuming to update.

Example Typical WorkFlow

user1:

ssh lxplus

cd /afs/cern.ch/user/u/user1/project

vi code.c

make

cp code /afs/cern.ch/user/u/user1/public/project

user2:

ssh lxplus

cd /afs/cern.ch/user/u/user1/project

cp code ~/myproject

cd myproject

bsub –q8nm code data

Very lightweight workflow to share work and computing resources.
How will this change after the PhaseOut?

AFS and LSF Phase Out: Process

My understanding is:

- AFS will be discontinued in 2019, but it will increasingly difficult to work on it now (loosing client support, IT support steer on alternatives)
- Alternatives to AFS for some use cases: EOS, CVMFS, DBonDemand.
- No alternative for remaining OpenAFS use cases for the time being.
- LSF will be discontinued in 18 months.
- Alternative to LSF: HTCondor. Meant to replace all use cases.

AFS and LSF Phase Out: Issues

1. Not a single product to replace OpenAFS (EOS, CVMFS, DBonDemand for the time being). One has to learn and test multiple solutions. -> very expensive for the user.
2. No alternative product for the most used scenarios (working directory, batch jobs, many small files). One has to rethink completely software and frameworks . -> Even more expensive for the users.
3. Poor or no documentation on the usage, features or known issues for the alternatives (EOS, CVMFS, HTCondor). Impossible to know in advance if application X will work unless dedicated testing. -> again at the cost of the users.
4. No clear timeline and roadmap ("EOS can evolve as CERN product, but is not clear how", "Feature X will not|might be implemented"). Contradicting messages on the future of the services. E.g.will LXPLUS and BATCH node have a shared file system (at all, supporting legacy applications, access control)? -> Too many options to follow, again the cost of the users.

Some IT support is alleviating the cost (e.g. Laurence for porting a fraction of SixDesk).

The Phase Out process requires substantial user resources that were not allocated at least by ABP.

AFS and LSF Phase Out: Technical issues

EOS features discovered in few months of personal testing:

- Large space for user and project.
- Fast read access.
- Limited maximum number of files.
- No anonymous read access.
- Fuse mounted EOS available in lxplus and SWAN but has many issues with legacy application (VIM, Jupyter, make, SQLite, …) expecting a well behaved file system.
- Vey slow in creating and modifying existing files.
- EOS command line tool cannot be easily compiled (or compiled at all) in Linux distribution. Problems even in Scientific Linux.
- Today no support in TN (under discussion as far as I heard).

Clearly not even close for a replacement of OpenAFS, but very good for serving big data files and we are already using it for it.

AFS and LSF Phase Out: Conclusion

For my personal experience:

- AFS and LFS Phase Out are not transparent for the users.
- following-up the phase–out process (study new products, test them, interact with IT developers, understand implications) requires substantial time.
- The possible outcomes, as of now, imply presently working software to be rethought and re-developed and not simply ported. This will require substantial resources that are not presently available.
- My core activities has been slowed down to follow the phase-out process already.