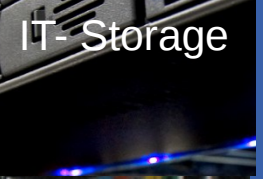


CERN's AFS Replacement Project

- Motivation, process, alternatives
- Impact & opportunities

AFS usage at CERN

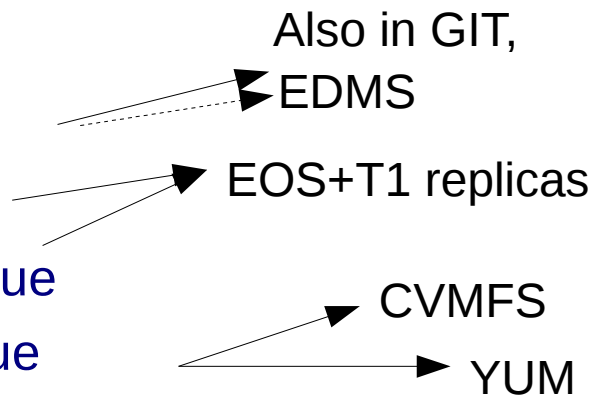
- In use since **1990**
- 35k users (5k active/day), 450TB data, **3.5B** files/dirs, **3.5B** accesses/day
  - Last year: +80TB, +500M files
- 50 (old=small) fileservers, 5 DB servers, 1.2FTE / 3 people

Split into

- Personal $HOME (2..10GB volumes)
  - Automatically created for every (UNIX) account
- Personal workspace (10..100GB)
  - Self-service
- Shared project space (1GB..10TB – vols. capped at 100GB)
  - Delegated admin powers
- group shell environments
  - “HEPIX” scripts (but apparently only remaining user..)

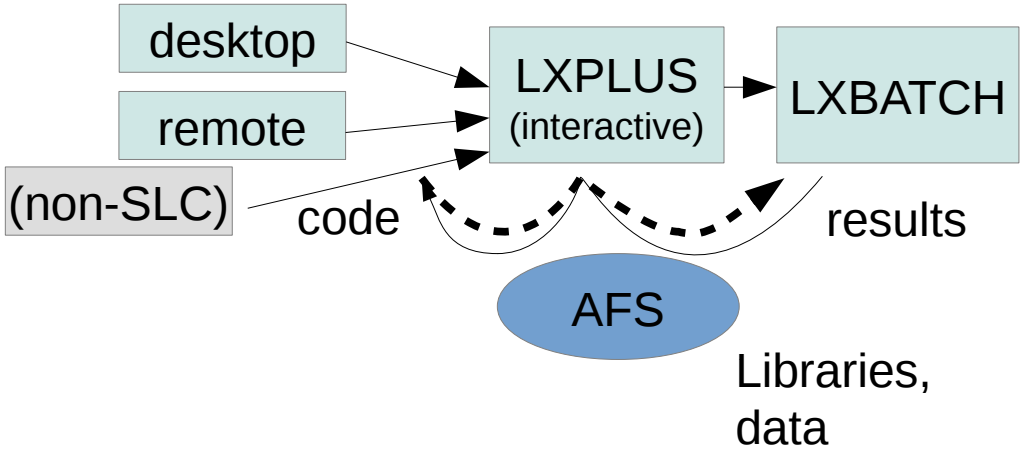
Usage 2

- By usage:
  - “Work” user data (.doc, .ppt, code, analysis executables etc)
  - Configuration: Home directories, group settings
  - Project space (=shared)
    - Experiment data
    - Software development, build & distribution (binaries, tarballs..)
    - Web (3.6k “personal”, 600 project)
    - Storage for other (distributed) services
    - Also: state machines, cheap “backup”, remote access.. and lots of abandoned data.
- By creation:
  - Human-crafted (documents, code): high-value
  - Physics data: high-value
  - Physics derived data (MC, analysis): medium-value
  - Machine-generated for re-use (binaries): low-value
  - temporary, write-once read-never (or -once): junk

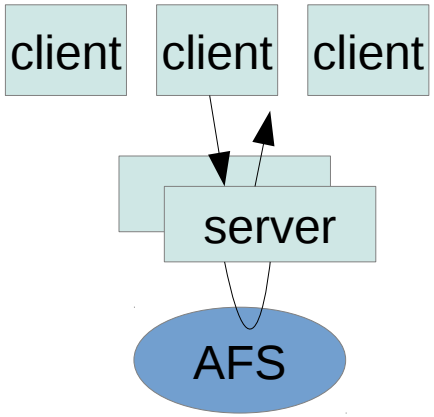


Usage 3 – Building block

- AFS is basis for local “Compute” workflow (non-grid)



- Services:



Web, Twiki,
SVN,(CVS,)
LXPLUS etc..

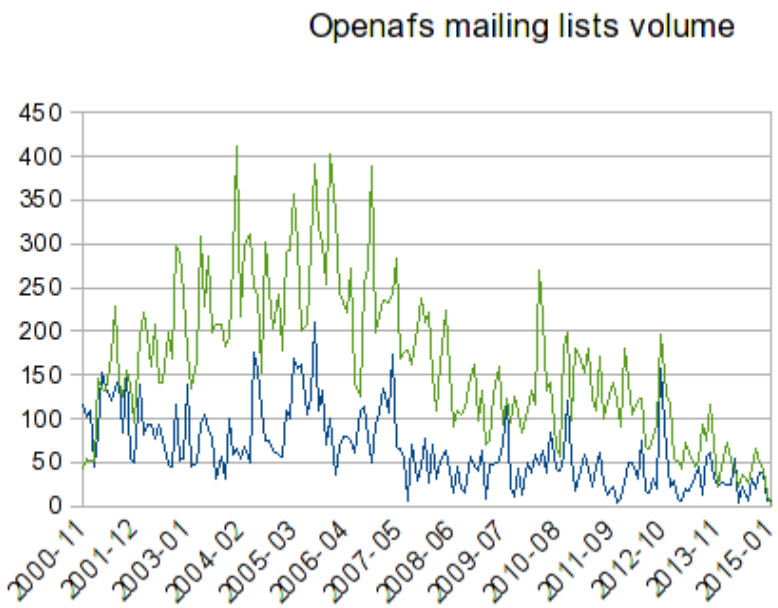
Why phase out?

?

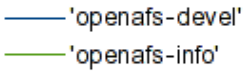
Why phase out?

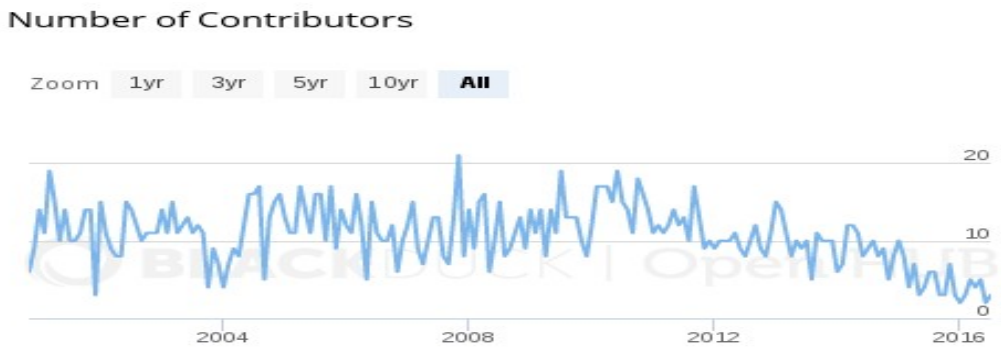
- OpenAFS project is in **(slow) decline**
  - Various “soft” indicators: releases, traffic, people, conferences,.. See also project state in S.Wiesand's talk last HEPIX
  - Pent-up changes: IPv6, DES (backward compat.. ®?)
    → Funding worries → ecosystem (2 companies, little else)
  - Ongoing client upkeep (incl signed binaries on Win+Mac)
- Technical - widening gap
  - SPOF (per-volume) architecture vs ever-bigger machines
  - RX protocol vs “long fat pipes” - volmove, replication, backup..
  - Odd limitations (32k files in directory)
- But
  - Project is **still “functional”** - new releases, slow changes

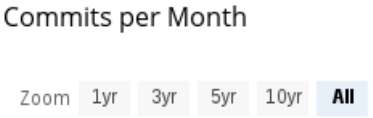
Decline?

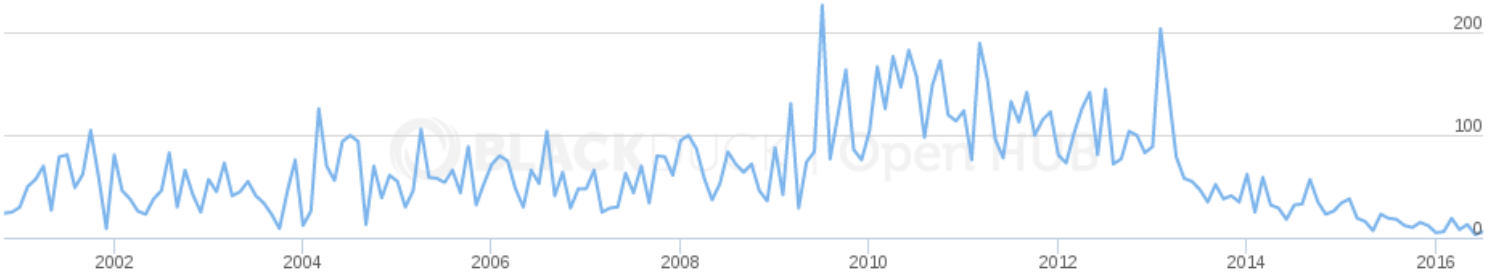


Some “soft” indicators









- Nevertheless:
  - The AFS architecture is still impressive
  - [and we are grateful to the OpenAFS project and community]
- “FUD” ?
  - Less “**f**ear” and “**d**oubt”, but huge “**U**ncertainty”
  - Combined with many dependencies: (potential) high **I**mpact
  - Effect on user + experiments: timelines / intervention slots

- Phaseout = **controlled** service evolution, no panic
- Early(?) start = effort can be spread, tied to other upgrades

- YMMV

Alternative to phaseout?

- Keep OpenAFS “forever” .. ?
  - Self-support
    - Possible, need increasing manpower.
    - Probably no more architectural changes.
    - Unlikely to turn around the whole project
  - Commercial support for OpenAFS
    - Possible now, but unclear business perspectives
  - Commercial alternative (AuriStor)
    - unclear business perspective, single-source
    - Compatible with AFS, but non-wide-HEP solution
  - Segregate + decay (firewall)
    - Contingency. Will eventually die out due to lack of clients

delay ..

Migrate when?

Where to go?

- AFS is **very good:**
  - Many small files – decentralized=scalable namespace
  - Rapid create/delete on single client = writeback cache
  - POSIXy enough for many applications (locks etc)
  - Cache and Readonly replicas can cope with (moderately) high loads
  - Secure (enough) for access from untrusted clients and remote
  - Multiplatform and free.
- Did not find a single ready-made drop-in replacement

= Need to go over use cases one-by-one

Where to go?

- CERN Migration targets
  - **CERNBOX** – human-generated content
  - **EOS-FUSE** – filesystem access
  - **EOS** – live data
  - **CVMFS** – (massive) software distribution
  - **CASTOR** – archive + dead data
  - Delete (?) – machine-generated junk & obsolete
  - Special cases: cluster-level filesystems (NFS, CEPHFS, HDFS)
- Review: Some use cases {c|sh}ould **change:** (after 26 years...)
  - Interactive analysis: SWAN
  - Temp files : use local disk or memory
  - Browsers, Mail: stay local
  - “defined” OS+compiler: VM / containers

Why EOS?

- Strategic:
  - EOS already holds most physics data at CERN
  - Building block for several new services
    - CERNBOX – very popular
    - SWAN – huge interest
    - disk subsystem of future tape archive (CTA)
  - EOS-FUSE (single-user) is widely used in experiments
    - Despite not really being encouraged..
- Full control over implementation
  - Flexibility
  - Non-standard – can extend at will

Is EOS a drop-in? [not yet]

- EOS-FUSE has made huge progress since start of the year, but ..
  - Still (much) slower than AFS for file creation/delete
  - Still odd behaviour for some applications

    Ongoing: rewrite/cleanup
- EOS namespace: split & centralized
  - All CERN EOS instances together: 820 M files (ALICE alone 370M @ 311GB RAM)
  - Boot time: 15min..1h

    Alternative implementation in the works
- Problem: “$HOME” mixed bag of use cases

See A.Manzi's talk

$HOME, sweet $HOME

- Different things in there:
  - Configuration (shell + per-application settings)
  - Code, binaries (not so much looking at .doc)
  - Result (ROOT files, job logs)
  - Temp data (caches, .o files)
- Short-lived low-value files are bad for EOS-FUSE
  - “Firefox Cache” bogeyman
- General idea – split Desktop and Compute $HOME
  - Desktop – stays mostly local (no backup!)
  - Compute – EOS(-FUSE)
  - Overlap – kept in Sync via CERNBOX
    - Symlink hell for .bashrc et al

DRAFT/ongoing

Impact on other IT services

- Multi-role LXPLUS:
  - External SSH access gateway
  - LSF submission machine
  - “default” SLC6/CC7 validated environment
    - Analysis compile, debug, run
  - 'acrontab' recipient, mail reading, browsing..

    → disentangle from “AFS”

- BATCH: LSF → CONDOR migration
  - Opportunity for better efficiency
  - (CONDOR will have AFS access)

Future Computing @CERN

- Account: split “UNIX” account from “AFS” account
  - Home directory is optional.
- WEBAFS → WEBEOS: same setup, different FS. Works.
- AFS-the-free-backup: make explicit. We have tapes.

Phaseout Timeline

1998: “look at alternatives..”

2014: “rather bad state”

✓ 2015: NOISE & Discovery

- initial communications
- establish experiment contacts
- Use case discovery

➛ 2016: EASY

- Web → EOSFUSE (✓)
- Projects → EOSFUSE (✓)
- Software → CVMFS (✓)
- + Obvious cleanups
- + Test & Improve alternatives

2017: HARDER

- Dead experiments – DPHEP?
- More elaborate use cases
- Home directories(?)

2018+19: HARD

- As in 'die-hard'..
- (No more “LHC-stop-threat” in 2019)

☠ (End of) LongShutdown2

Phaseout – How?

Coordinators → **Common use cases**
- Default actions

→ **Track migration**

Users & Experiments
- Communication:
  - ITUM
  - Direct meetings
  - SNOW disclaimers

Tree:

/afs/cern.ch
|-- afs
|-- aleph
|-- alice
|-- ams
|-- asis
...

AFS

Projects:
- Contact admins
- SNOW: block new requests

Clients:
- Scan + graph to track usage

Summary

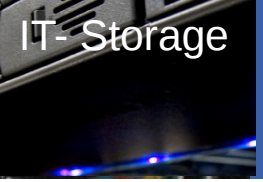
- AFS Phaseout has to happen.
  - But not in “panic” mode
    - Hopefully started early enough
    - OpenAFS will be OK on SLC6/CC7 ..
  - LS2 provides contingency for mop-up
- Impact all over CERN

Externals: RU affected?

→ **get in touch..**

- Attractive new services & tools – use them
  - Rethink use cases, not transplant 1-to-1

https://twiki.cern.ch/twiki/bin/viewauth/IT/AfsPhaseout



Questions?

Frequently asked questions

- Will this work? Are you serious?
- Why don't you “rescue” the OpenAFS project (invest resources / manpower, drive changes)?
- EOS is a CERN-only solution, how is this better than a community-supported project?
- What about external users / sites relying on CERN AFS?