



DSS

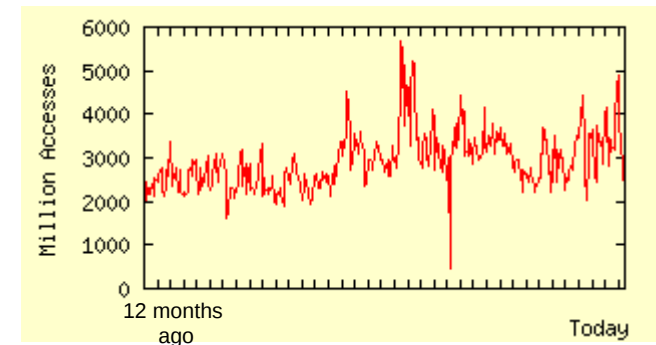
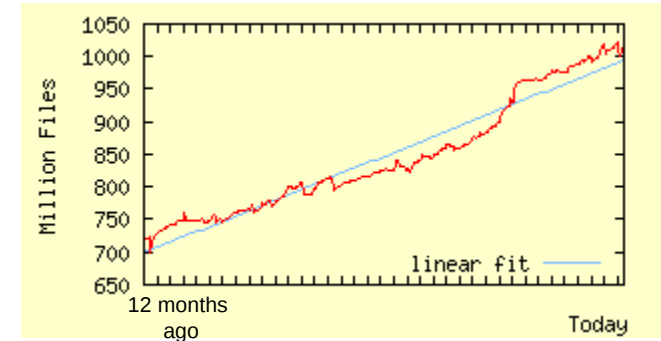
Evolving the AFS Service at CERN

Jakub T. Moscicki, Arne Wiebalck

HEPiX Spring 2012 Workshop
Prague, April 2012

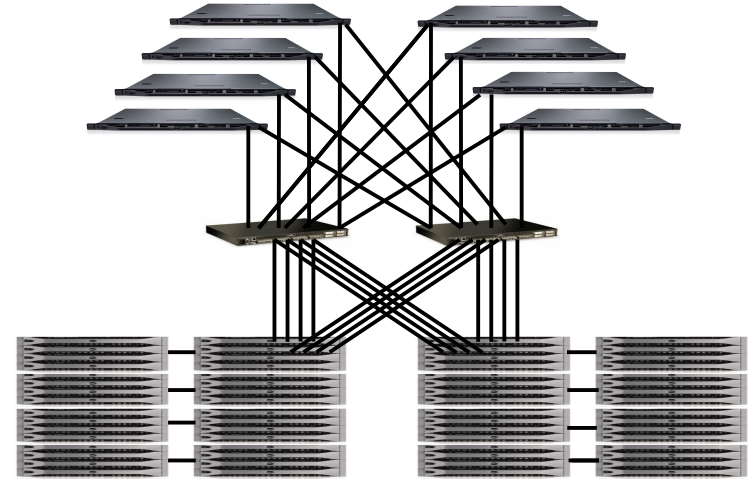


- Service provides networked storage to CERN users
 - small data / low latency
 - 32'000+ home directories
 - 300+ project spaces
- Service key data
 - ~55 file servers, ~900 disks
 - ~64'000 volumes
 - ~72TB of data
 - >1 billion files (+300 M/yr)
- Service activity
 - 10'000 CERN clients
 - 5'000 off-site clients
 - ~30'000 accesses/sec
 - ~300 million reads and writes/day



➤ Fibre channel fabrics

- some 40 servers
- ~80 FC disk trays
- 4 fabrics
- 300GB SAS disks
- home dirs, projects



➤ Standard disk servers

- some 10 servers
- directly attached SATA disks
- s/w RAID over iSCSI
- scratch data

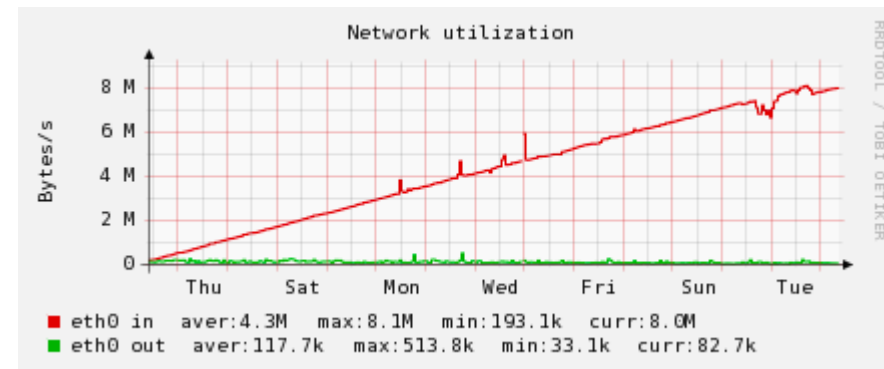


- Current production version is openafs-1.4.14 ...
 - both client and servers
 - since February 2011
 - no major issues

- ... plus a set of CERN-only patches
 - on-demand attach
 - handling of clients behind NATs
 - request throttling
 - remote network statistics
 - forced offline
 - client shutdown
 - ...

- openafs-1.6.x ?

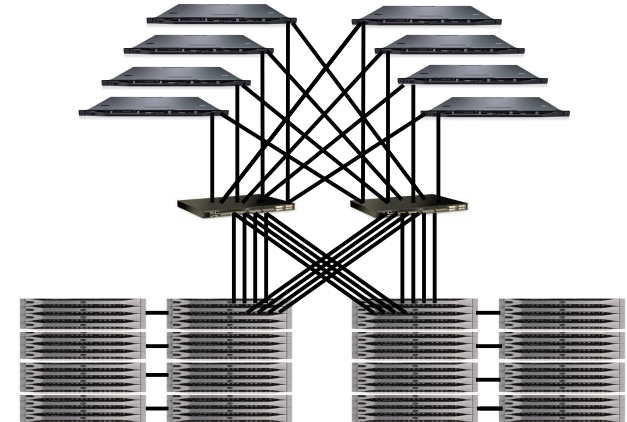
- New release 1.6.0 deployed on remote sites
 - Client keep-alive packets (for NAT port mapping)
 - Bug: no proper connection cleanup (?)
 - linearly increasing packet rate
 - a remote 1,500 machine cluster generating 1MHz packet rate after few days
 - impacted CERN's firewall and AFS fileservers
 - 1.6.1-pre* patch improved the situation but not solved it
 - After ~2 weeks → 300-400 pings/s per client
 - 1.6.1 seems OK again
- Please beware
 - **share your upgrade plans with us** if your site uses heavily the CERN AFS cell
 - we will protect stability of our cell if endangered by remote sites





- **More space**
 - 1GB home directories
 - some GB of work/scratch space
- **More IOPs**
 - access is always 'too slow'
- **Maintain service quality**
 - availability (no service interruption)
 - features (e.g. daily backup)
- **Service simplification**
 - uniform backup policy, requests handling (→ self-service), space types (scratch vs 1st class space), ...
- **Service clarification**
 - quotas, retention policies, incident handling, ...

- + **Homogenous**
 - simplifies operations
- + **Reliable**
 - hardware, dualpath
 - hw interventions do work (hot-swappable hw parts)
- + **Sufficient performance**
 - high-end, small SAS disks
- **Non-standard hardware at CERN**
 - procurement & metadata, testing & burn-in, tools (console)
- **Complicated**
 - initial setup
 - stress situations
- **Price/GB**
 - up to an order of magnitude more expensive
 - prevents easy scaling



➤ Hardware Setup

- 2 servers and 2 trays form a “unit”
- all disks visible on both servers
- 16x 2TB NLSAS, 4x 256 GB SSDs

➤ Reliability

- JBODs (no h/w raid controllers)
- s/w RAID across arrays
- “volume take-over”

➤ Performance

- make use of SSDs to compensate larger disks:
FACEBOOK's flashcache



Storage Unit (SU)

➤ General purpose block cache
for the Linux kernel

- developed by facebook for MySQL
- integrated into device mapper
- supports writeback/writethrough

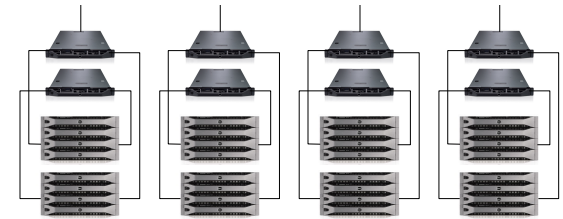
The Facebook logo is displayed in white lowercase letters on a dark blue rectangular background.

```
# dmsetup status
...
vicepcd: 0 585937224 flashcache-wt stats:
      reads(396941574), writes(153485540)
      cache hits(293703891), cache hit percent (73)
      ...
      disk reads(103238560), ...
      ...
```

- In production since early 2011
- Can give significant speed-up

<https://github.com/facebook/flashcache>

- **Lower Price/GB**
 - even with 5% SSDs we gain a factor 4
- **Data availability**
 - no SPOF (same as before)
- **Simpler setup**
 - harder to incidentally destroy a “good” data disk as opposed to FC setup
 - ease things in stress situations
- **Performance**
 - SSD/flashcache shall compensate for larger disks
- **Streamlined hardware**
 - profit from well-established procedures (procurement, testing, burn-in, support)





- **Hardware stability**
 - Mixed initial experience with new hardware
 - SSD firmware problem made disks disappear after 5 reboots
 - sent back to vendor at initial burn-in tests
 - SSDs drop out of the system mysteriously
 - needed to patch the flashcache to handle it correctly
- **Manual handling of sw RAID via mdadm**
 - Hot spares added manually
 - Need to keep track of replaced disks manually
 - Still need to get experience and possibly automate it
- **Backup**
 - some 10 TB/server
 - depends on data change rate
 - backup setup under review
- **User behaviour**
 - more space, new use cases
 - separation of home dirs and work spaces

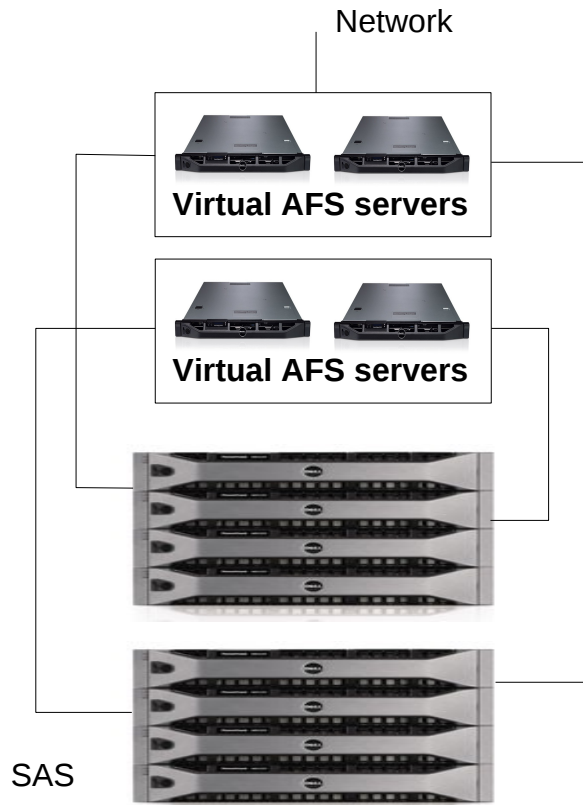


- **10GB home directories (10x more)**
 - we went to 2GB already
- **100GB work spaces**
 - 20GB already available for all CERN users
 - ~1000 workspaces created in first 4 weeks (good uptake)
 - new hardware provides 96TB of usable space
- **SSD read caching**
 - chose safe write-through option
 - currently observed cache hit rate 97-99% on workspaces
- **Daily backups**
 - retention of 6 months
- **Streamlined interfaces & procedures**
 - - web-based self-service



- Fixed number of threads per FS to serve reqs
 - limit often hit, immediately visible to users
 - **isolation**: move overload off
 - **protection**: shield precious data
- Throttling
 - done: max 80% of threads per volume, reschedule
 - protects only from overload of one volume
 - trade-off between lower threshold and QoS
 - too high? needs a revision...
- Increase thread limit within FS process
 - tried: did not work out-of-the-box, dump analysis not conclusive, openAFS 1.6 (?)
 - also: thread management may become issue (other limits?)
- Use more, smaller (“virtualized”) servers
 - virtual machines, linux containers
 - looks promising, needs further investigation

- Increase thread limit within physical server
 - virtual machines or linux containers



- Get 256 threads per virtual server
- Dual active mode possible (quick failover)
- Usable for isolation / protection?
- I/O penalty by virtualisation?
- Software stack getting larger

Storage Unit (SU)

- AFS cornerstone of data services at CERN
- User demands drive a change → the service needs to adapt
 - from FC fabrics to external SAS-based storage units
 - from small expensive to large mainstream disks
 - compensate performance penalty with SSDs
- Aggressive service growth currently under way
- Thread shortage main challenge in 2012



Option 1: Both servers of the SU are active	Option 2: Only one server in an SU is active
+ both servers contribute to file serving memory, network, CPU	– one server “wasted” mitigated by 2 different servers?
+ impact by file server problem is less severe	– large servers backup, flexibility, cross-talk
– “joining” the data on one server is difficult: no sysid copy possible syncserv slow (1/sec, no bulk!) syncserv not (easily) possible when the other server is completely gone	+ “volume take-over” fast via sysid incidents/maintenance
– split after join?	+ switch back and forth possible
	+ understand limits add capacity if needed

