Engineering the CernVM-FileSystem as a High Bandwidth Distributed Filesystem for Auxiliary Physics Data

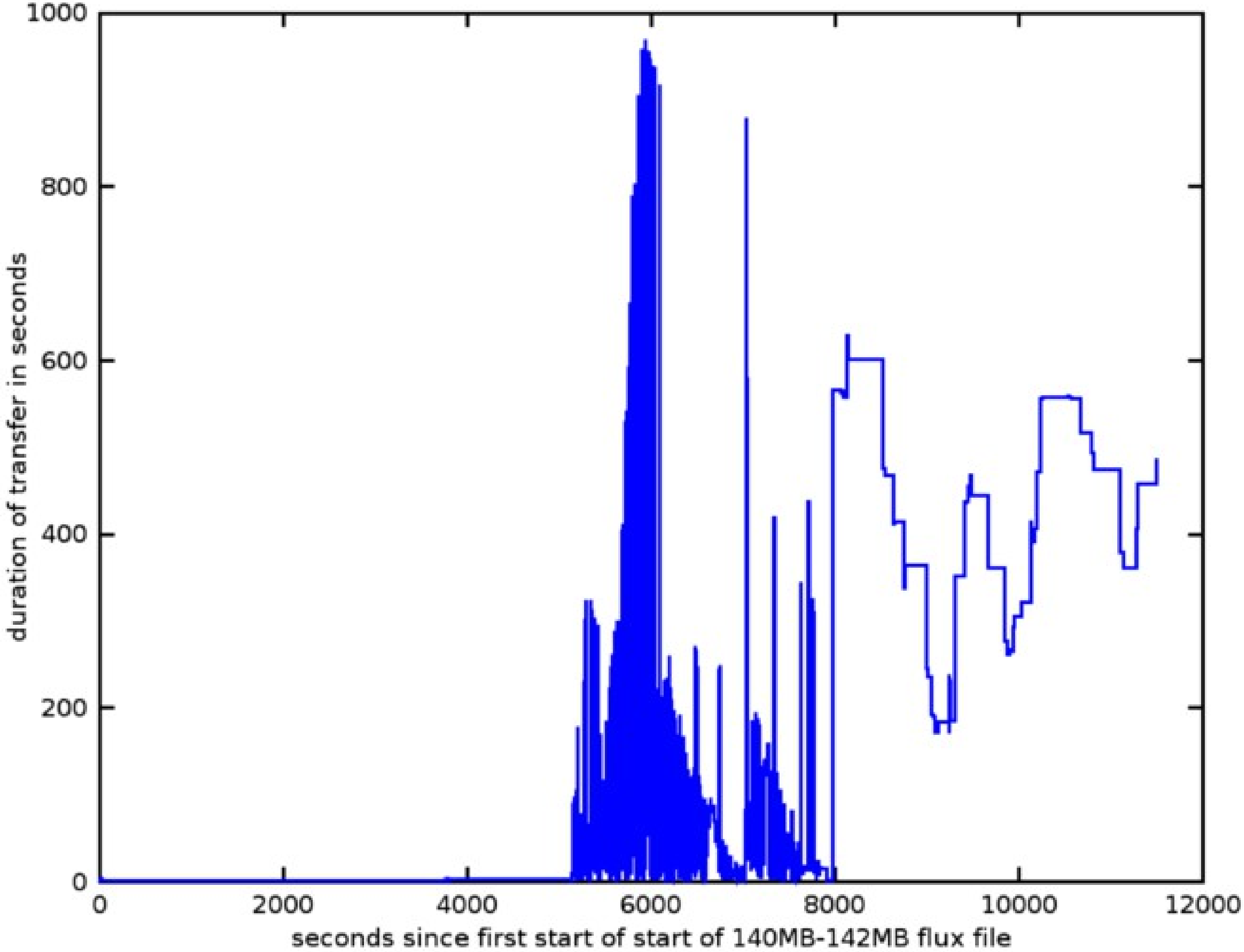
Talk written by Dave Dykstra (Fermilab)
Talk presented by Jakob Blomer (CERN)

CHEP 2015 – April 2015

What we mean by Auxiliary Data

- Some experiments need “Auxiliary Data”, distinct from
  - Event Data which is different for every job
  - Conditions Data that’s the same for every job in a batch
- Instead, there is some sharing between jobs but not complete
- Also, each job reads gigabytes similar to Event Eata rather than 100s of megabytes similar to Conditions Data
- Example: Neutrino GENIE “flux” files, with datasets ranging from 15GB to 250GB, and each job reading ~2GB subset
- Hit ratio too low for ordinary CVMFS
  - Measurements show squid performing very poorly, limited by disk speeds rather than network
  - Squid machines are not generally engineered with fast disks

GENIE flux file load times on ordinary CVMFS



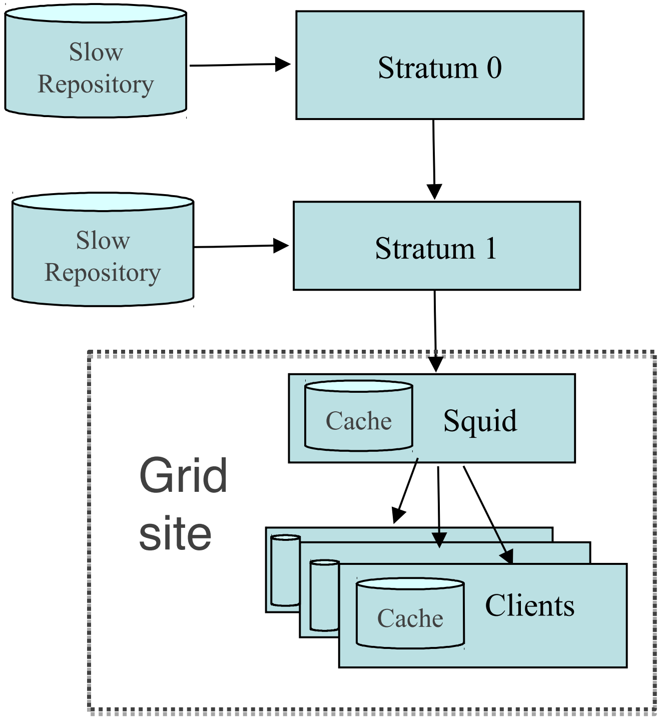
128 jobs
~140MB files
~2GB per job
15GB dataset

4 2-Gbit/s
squids
at Fermilab

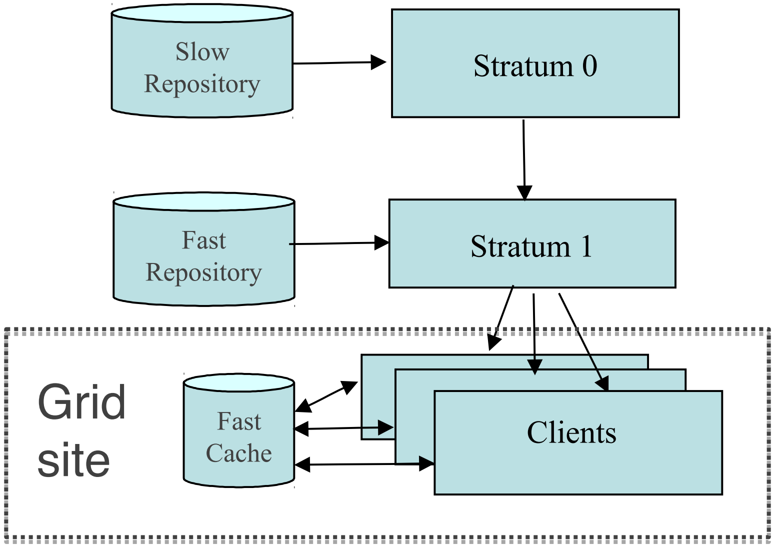
Leveraging CVMFS & high-speed Storage Elements

- Use CVMFS “alien cache” feature for Auxiliary Data
  - Use CVMFS repository domain separate from ordinary code repositories, e.g. “osgdata.org”
  - Configure all clients in a cluster to share a POSIX-accessed alien cache on site’s high-speed storage element for that domain, instead of using local disk cache and site squid
    - Already engineered for much higher disk bandwidth than squid
    - Repeats from same node make use of kernel filesystem buffers on that node
  - Tested Auxiliary Data using alien cache on Lustre, Hadoop-Fuse, and NFSv4.1-dCache with very good results
- To accommodate large data sets, use dCache http server for Stratum 1

Ordinary CVMFS vs Auxiliary Data CVMFS

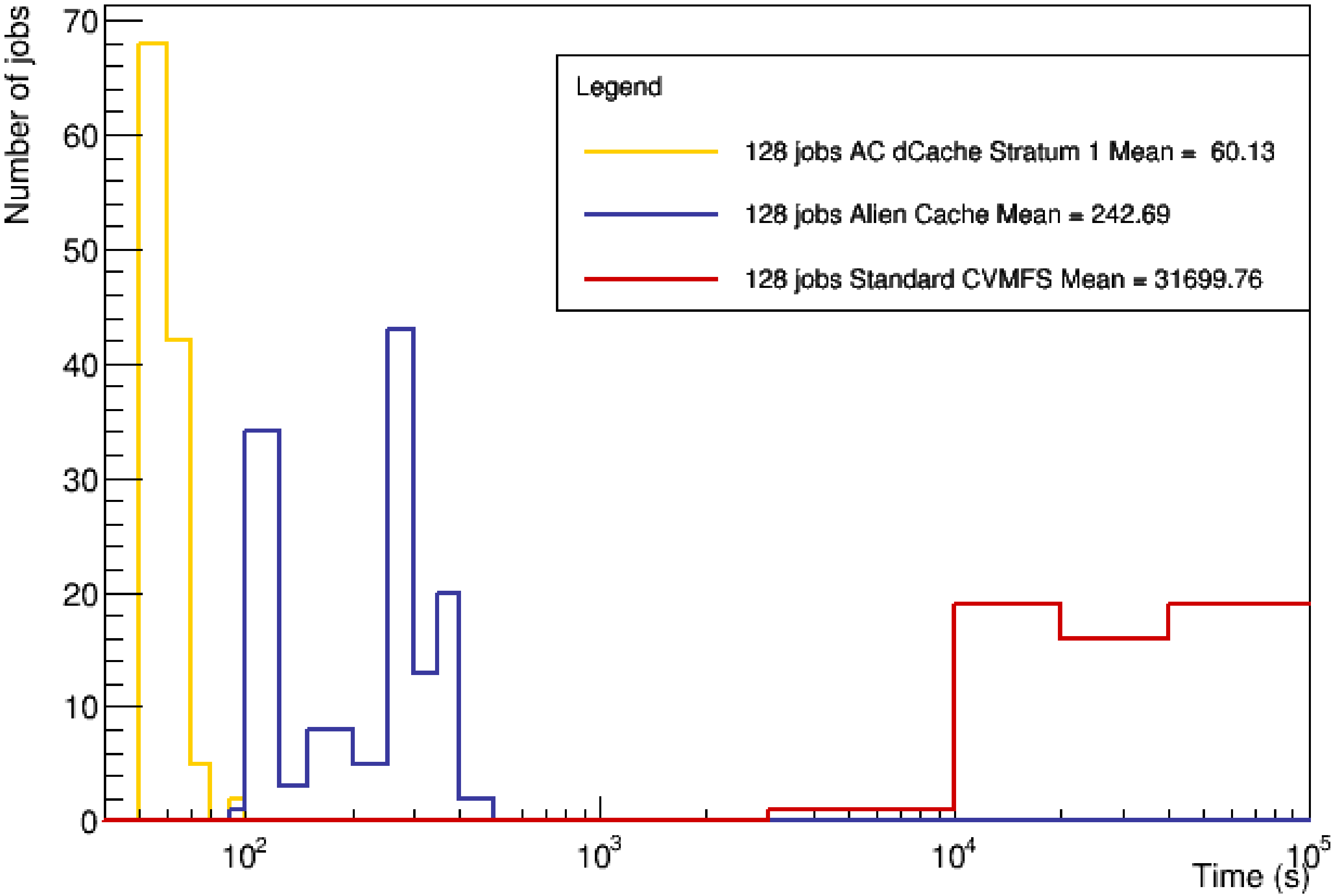


Ordinary CVMFS



Auxiliary Data CVMFS

GENIE Auxiliary Data load times on Nebraska Lustre



Details

- Each storage type required patches to cvmfs client
  - Patches are in cvmfs-2.1.19-1.8.osg and in cvmfs-2.1.20
- When multiple clients read same file from Stratum 1 to cache, only first copy is saved
  - Files downloaded to temporary names, renamed, first wins
- Configured repository with larger (~64MB) chunk sizes than ordinary CVMFS (~8MB)
- Stratum 1 uses ordinary cvmfs-server software except apache configured to forward data requests to dCache WEBDAV door
  - redirects requests to one of multiple high speed data servers

Client configuration

- Client domain.d config:

```
CVMFS_SERVER_URL="http://cvmfss1data.fnal.gov:8000/cvmfs/@fqrn@”
CVMFS_ALIEN_CACHE=/path/to/cache
CVMFS_HTTP_PROXY=DIRECT
CVMFS_QUOTA_LIMIT=-1
CVMFS_SHARED_CACHE=no
CVMFS_FOLLOW_REDIRECTS=yes
```

Future work

- So far have run ~400 parallel jobs on dCache, working on scaling up further
- Test on EOS storage
- Cache cleanup is native in dCache; cleanup for other storage types designed but not yet implemented
- Add HTTP-based access to alien cache (e.g. WEBDAV or S3)
- Create official data domain & set up second Stratum 1 at another site

Conclusion

- End result: high-speed distributed centrally-written (Write Once read Many, or WORM) POSIX filesystem using existing software & hardware
  - Convenient to use and configure

- Related work:
  - Distributed xrootd cache: https://twiki.opensciencegrid.org/bin/view/SoftwareTeam/SW023_XrootdAcrossOsg
  - XrootdFS: http://wt2.slac.stanford.edu/xrootdfs/xrootdfs.html

- Contributors:
  - Dave Dykstra, Brian Bockelman, Jakob Blomer, Ken Herner, Tanya Levshina, Marko Slyz