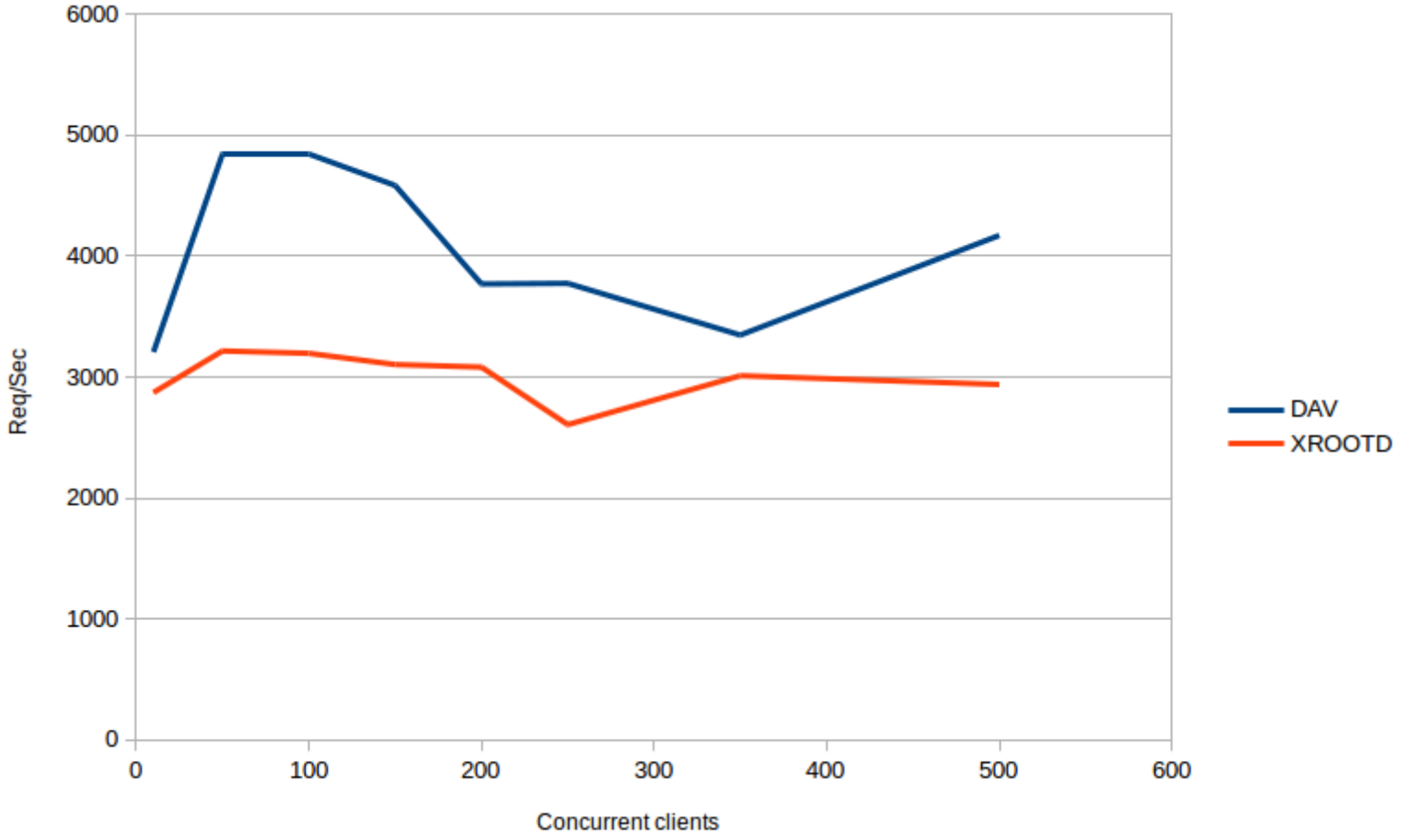
FTS / GFAL2 / Davix Update

Alejandro Alvarez Ayllon on behalf of the FTS and DMC development teams

DAVIX

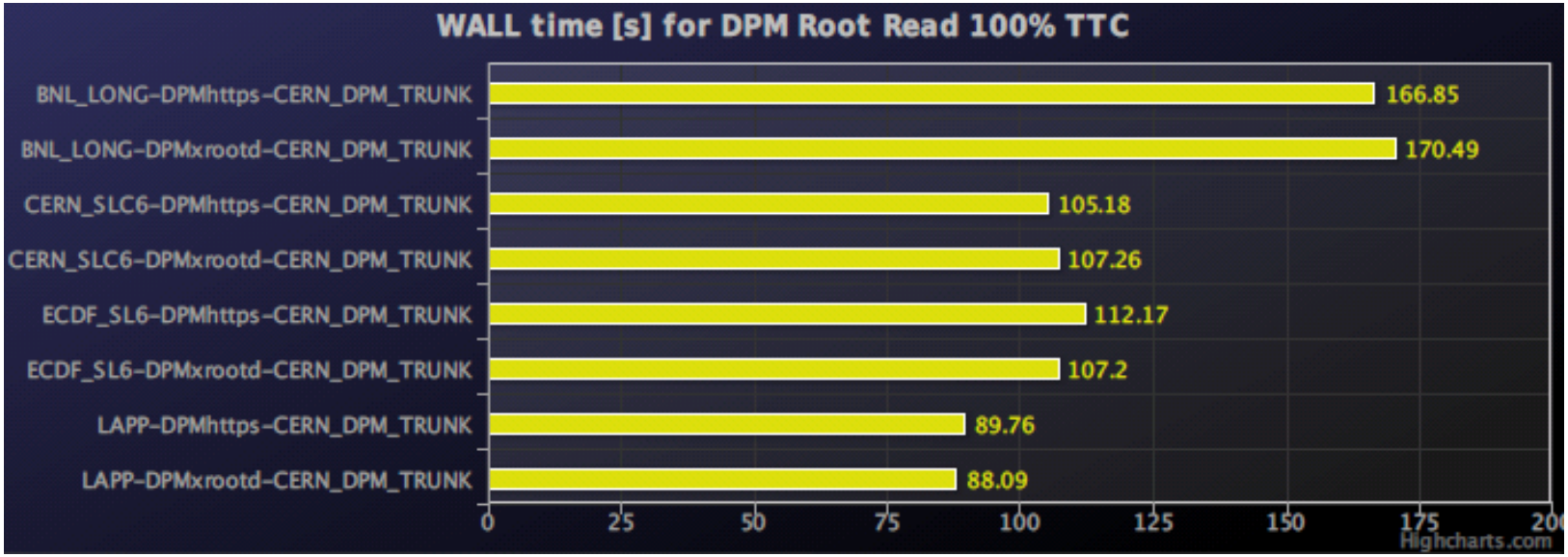
- Version **0.4.0** just released
  - Small 3rd party copy improvements
  - S3 new functionalities
  - Improved writing support

DAV vs XROOTD (Stats)



(*) DPM on LAN

DAV vs XROOTD (IO)



700 MiB ROOT file
100% event reads
30MiB cache

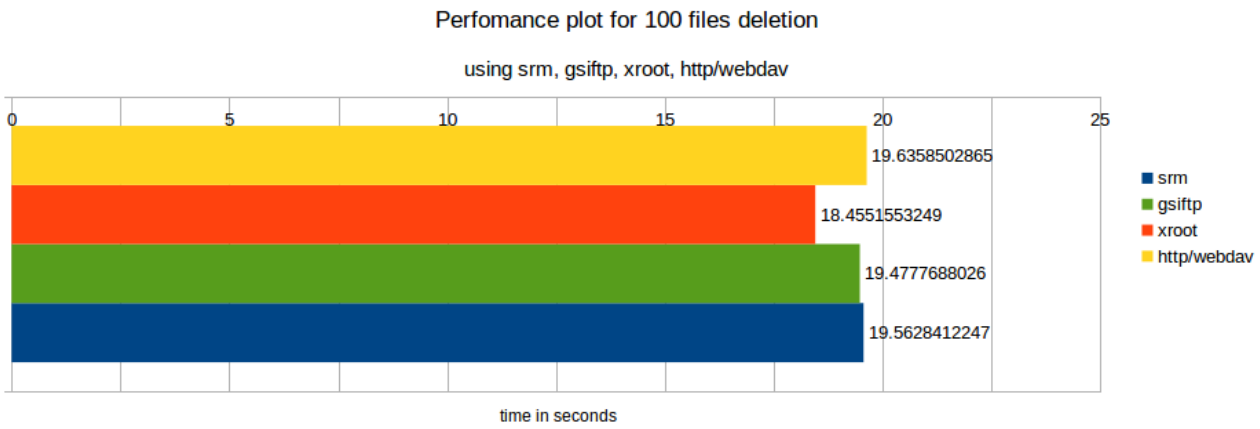
FTS3

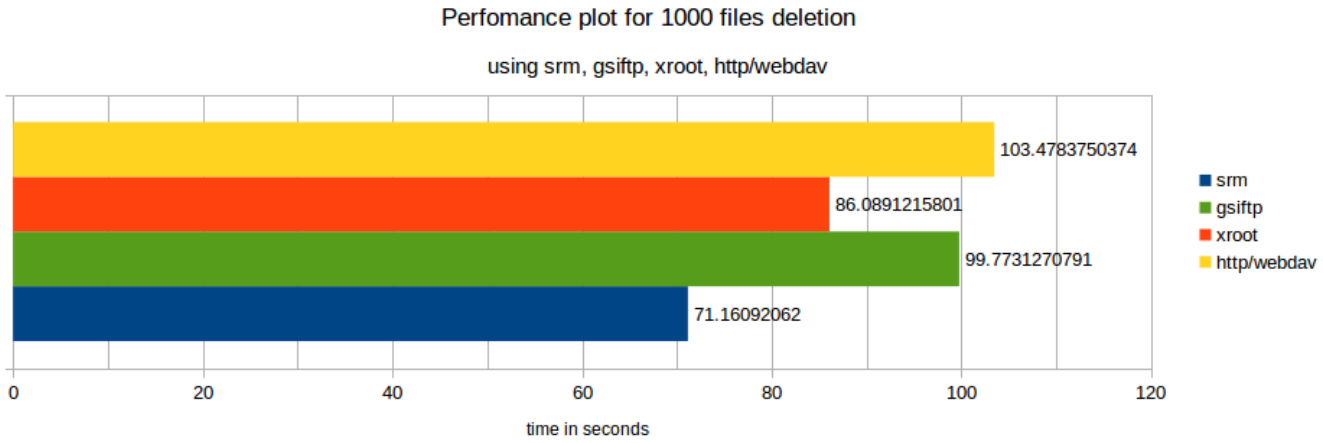
- Running smoothly, with relatively minor hiccups
  - Working to make it even smoother and easier to use
- Experimental features
  - S3 and Dropbox support
  - Deletion operations
- Future features
  - GridFTP bulk copies (pipelining)
    - Almost ready on gfal2

FTS3 - Deletions

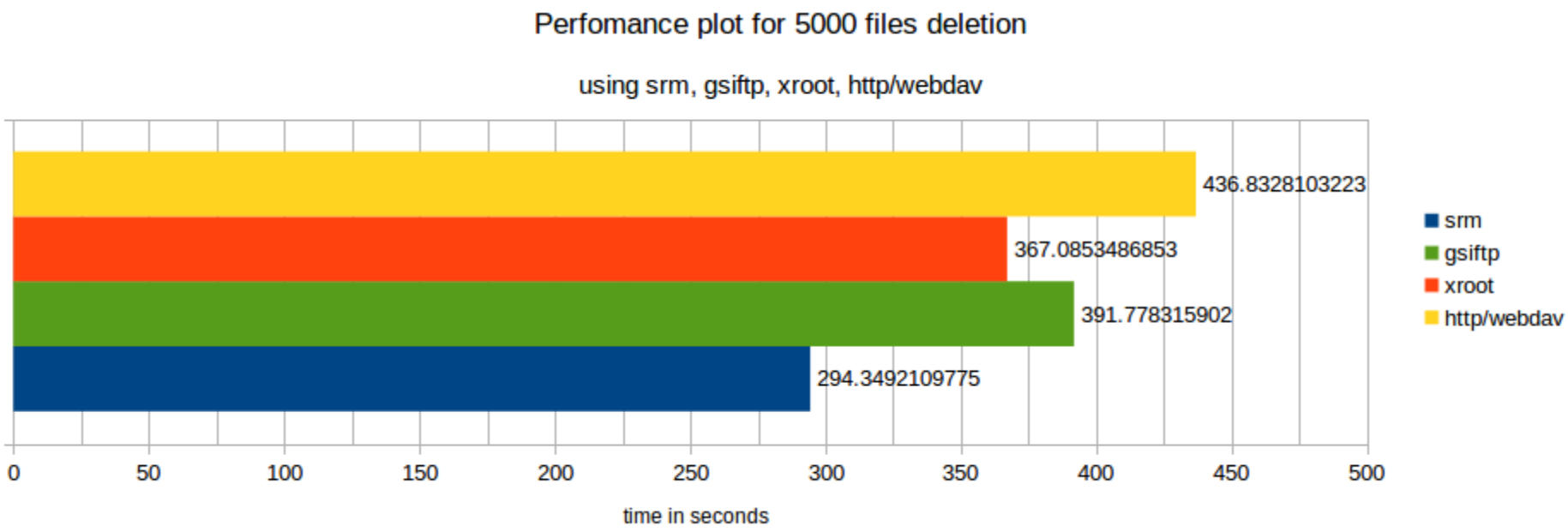
- We got some performance numbers!
  - Wall time from fts-delete until fts-transfer-status returns FINISHED
- Tested on a remote DPM node
- Credits to Anna Iutalova

FTS3 - Deletions





FTS3 - Deletions



FTS3 - Deletions

- SRM bulk deletion makes a big difference
- To be fair, HTTP is doing twice the work
  - STAT + DELETE each time, sequentially
    - To avoid unlinking a dir, or “rmdir-ing” a file
  - 2N operations!
  - Can implement a bulk operation with no stat
    - May unlink directories!!!
- HTTP DELETE could support pipelining
  - Needs quite a bit of work

FTS3 - Deletions

- For ≤ O(100) files any protocol would do
- For larger sets, SRM clearly wins
  - For the moment?
- Functionally tested every night
- Need to run the battery against
  - Different storage implementations
  - Different protocols
  - Under constant load

Protocol summary

- GFAL2 (hence, FTS3) supports
  - srm, xrootd, gsiftp, http/dav, s3, rfio, dcap, file, lfc
- I/O performance
  - http and xrootd perform similarly
- Third party copy support
  - xrootd, gsiftp, http/dav (DPM and dCache partially)
- Bulk copies
  - xrootd*, gsiftp
- Bulk deletions
  - srm, which performs best on deletions because of this
  - http seems to have room for improvement

Protocol summary

- Checksums
  - gfal2-util support
    - gfal-sum <file> <type>
    - gfal-copy with -K
  - Checksum natively supported
    - GridFTP, HTTP, XROOTD
  - On-the-fly fallback for the rest
  - Not all storages, nor all protocols, supports all checksum algorithms
    - adler32 seems to be the intersection?

Friendly reminder

- LCG-UTIL now is fully deprecated
  - Packages maintained in EL5 and 6
  - Will not be in EL7
- Please, use GFAL2!
  - Report bugs, anything you need, feel missing…
- Points of contact
  - http://dmc.web.cern.ch/
  - **dmc-support**@cern.ch

Questions?