



# Update on data access by jobs

- Data access efficiencies need improvements at many sites
  - Lower failure rates, better CPU-wallclock ratios
  - Depends on the supported experiments
    - May be OK for one, not so good for another
  - Depends on the site layout
    - Network infrastructure
    - Usually cannot be adapted in the short term
  - Depends on the access protocols and their usage
    - Read-ahead good for sequential access, bad for random access
    - RFIIO read ahead buffer size cannot (yet) be set by client code
      - One value per WN set by admin
    - Different experiments need different buffer sizes
      - Due to different event and processing models
    - Phase out some protocols in favor of others?
      - Probably not feasible in the short term



# Short term plans

- ATLAS are determining the best access method per site
  - Activity mostly driven by clouds that want to improve
  - Central steering through HammerCloud team
    - Global tests possibly early Oct.
  - Best method recorded per site in central configuration
  - CERN T3 testing: see next slides provided by Max Baak
- CMS intend to investigate improvements early Oct.
  - Taking note of ATLAS results
  - Each CMS site configures the protocol to be used by jobs
- LHCb, ALICE: no plans so far (?)
  - LHCb: dCache client vs. ROOT plugin vs. non-ROOT files
    - OK now

# Preliminary recommendations

- Xrootd & rfio read-ahead buffering: very inefficient
  - Lots of unnecessary data transfer (sometimes >50x data processed!)
  - 1 job completely blocks up 1Gbit ethernet card of lxbatch machines
    - Large spread in job times, ie. unreliable
- Xrootd: frustrating dependency on PoolFileCatalog.xml
- Don't use rfio protocol to loop over files on CASTOR!
  - >5x slower & takes up too much network bandwidth

# Preliminary recommendations

Different recommendations for single / multiple jobs

- Single jobs: FileStager does very well.
- Multiple, production-style jobs
  - Xrootd (no buffer) works extremely stable & fast on files in disk pool cache.
    - Factor  $\sim 2$  slow-down when read-ahead buffer turned on.
  - Two recommendations:
    - Xrootd, no buffer, for *cached* files
    - FileStager or Xrootd for *uncached* files