Protocol benchmarking on DPM

Use cases for HTTP and Xrootd

Fabrizio Furano
Adrien Devresse
Oliver Keeble (presenter)
Valentina Mancinelli

CERN

Summary

- The protocols
- The storage system
- The infrastructure
- The tests
- The conclusions

HTTP/Dav and Xrootd benchmarks

- Protocols have similar tech features, different implementations
- Our main use cases
  - Metadata hammering
  - Full file data transfers
  - Data analysis (inc vector reads)
- Are they comparable in performance ?
- Is HTTP “good enough” for supporting scientific data processing ?

DPM multiprotocol storage

- DPM storage is used for the tests
- Different frontends (xrootd, apache)
  - HTTPS redirects with a token to HTTP for access
- Sharing the metadata backend implementation
- Running on the same hardware, on the same disks
- We have been accumulating benchmark data daily for >1 year
  - Local metatdata tests
  - WAN access tests on the production infrastructure

HTTP/WebDAV

- Descriptive and readable, extensible
- Natural to interface to other Web-ish systems, e.g. federated identities
- Ubiquitous support
- Promotes accessibility (browsers)
- Supported by many data storage solutions: dCache, DPM, EOS, StoRM, Xrootd4
- Supports vector operations and transparent redirection (used in federation)
- Support has been integrated into ROOT via the Davix plugin

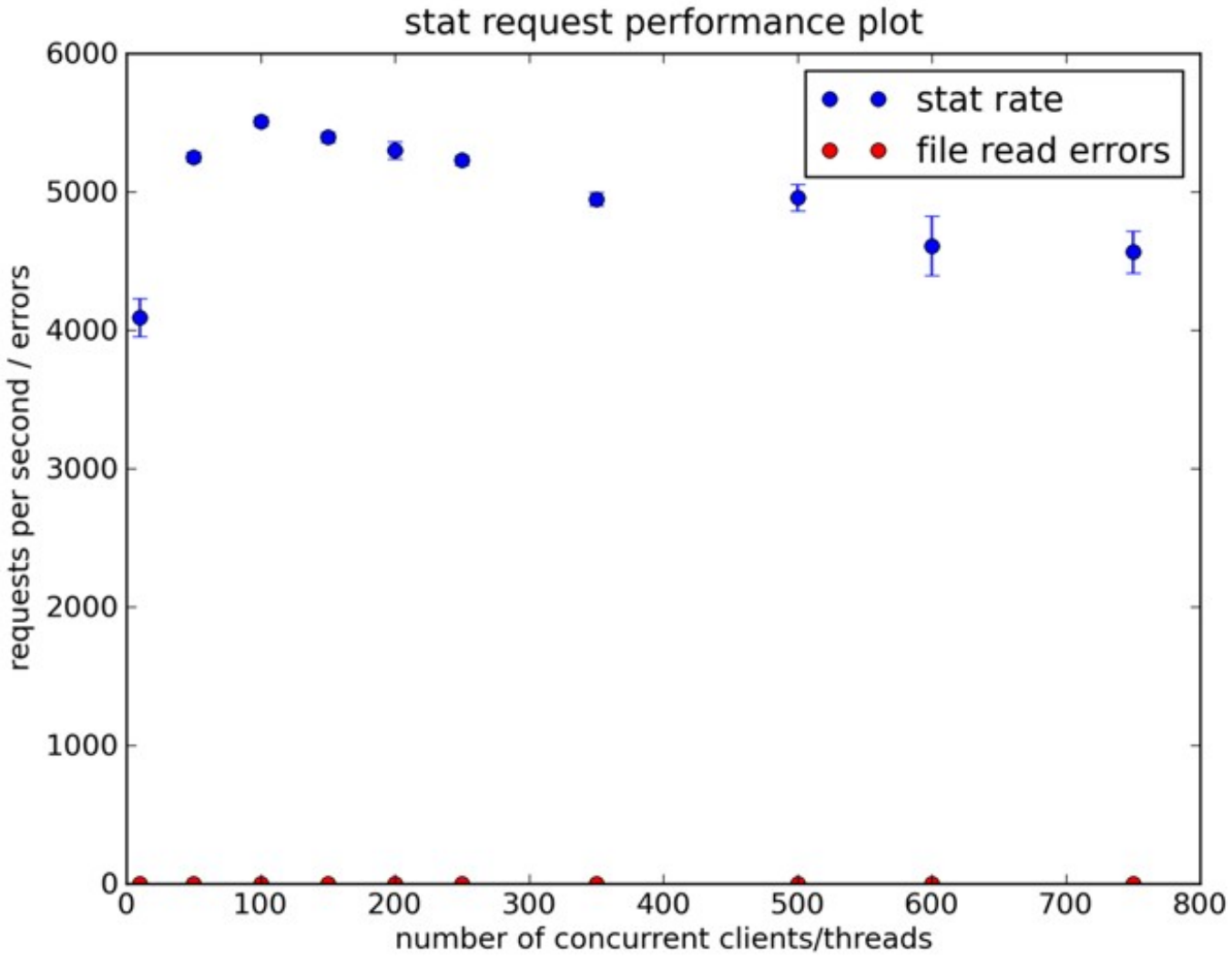
Xrootd

- Generic very scalable file server architecture, since 2003
- Efficient binary protocol
- Emphasis on low level performance
- Very well known in HEP
- Supported by many storage solutions: dCache, DPM, EOS, Xrootd4
- Supports vector operations and transparent redirection (used in federation)
- Native support in ROOT

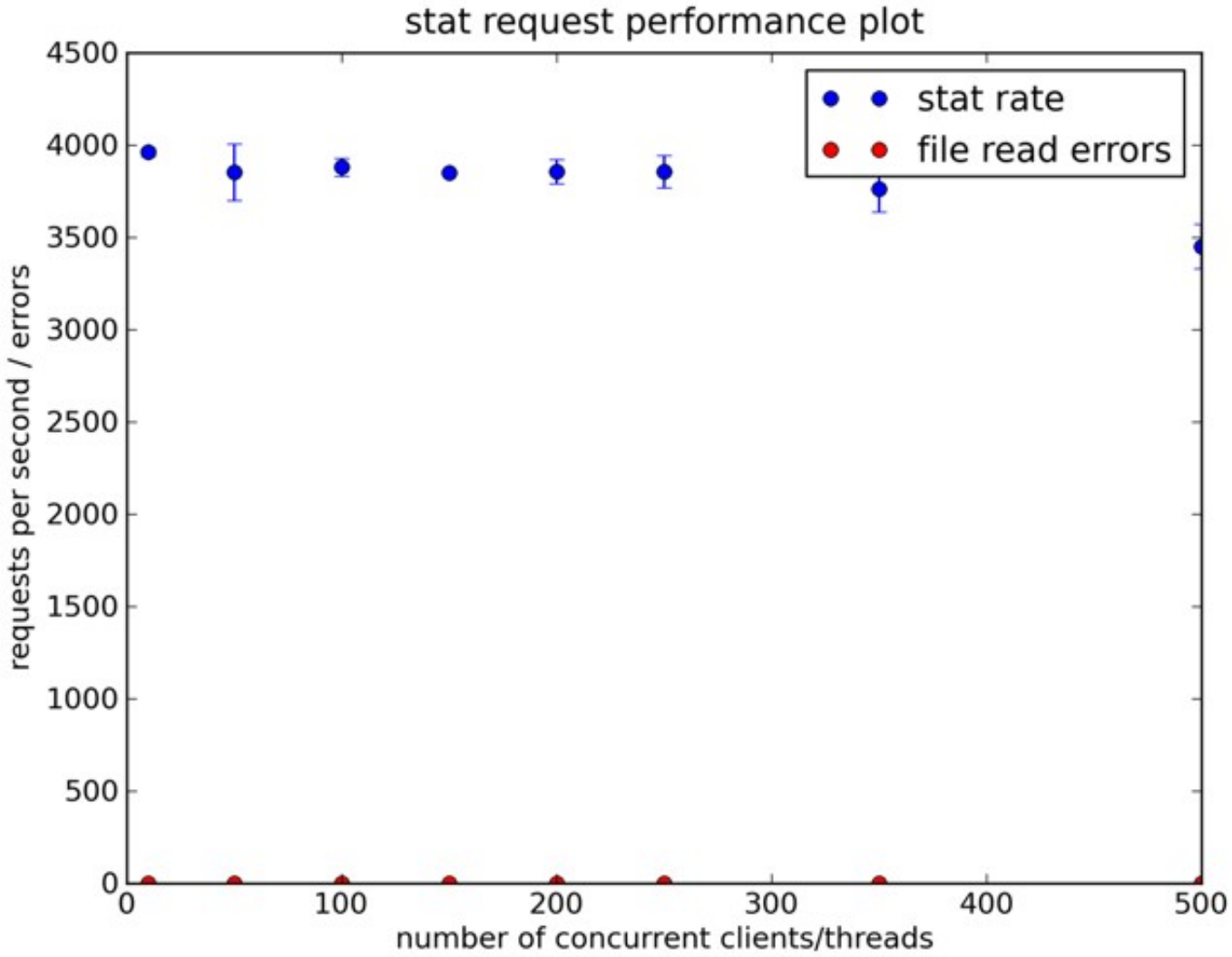
Metadata: LAN tests

- Server/client on LAN
- Stresses the peak metadata performance by hammering a fixed set of 10K files
- Important testsuite for our implemention DPM performance improvements
- Useful for developers to spot regressions
- We keep the whole history, for years

Stat() through WebDAV



Stat() through Xrootd



Metadata

- Bottom line is that they are comparable
  - Differences of a few percent are not significant in this use case
- Differences may be due to
  - Relative complexity of implementation of the DPM plugins for the specific frontend
  - Intrinsic differences
    - Security models and protocol features
      - SSL session caching in davix
      - Connection multiplexing in xrootd

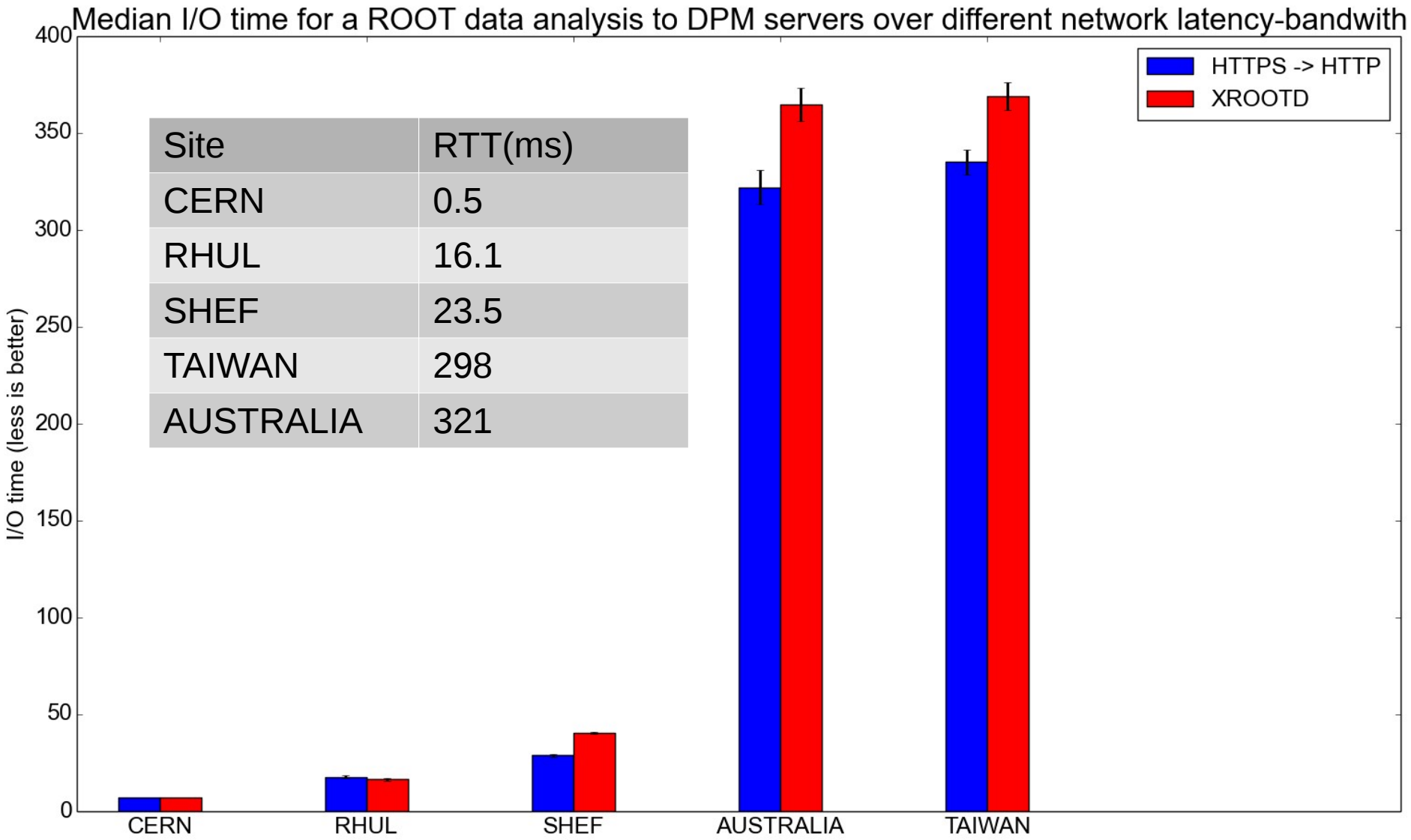
Access: WAN tests

- We run a ROOT analysis on an ATLAS data file, whose pattern we know
- ROOT TTreeCache is used for boosting performance

- Participating sites supply storage and compute
- Remote jobs are scheduled using the HammerCloud framework
- We share the reporting infrastructure with FAX

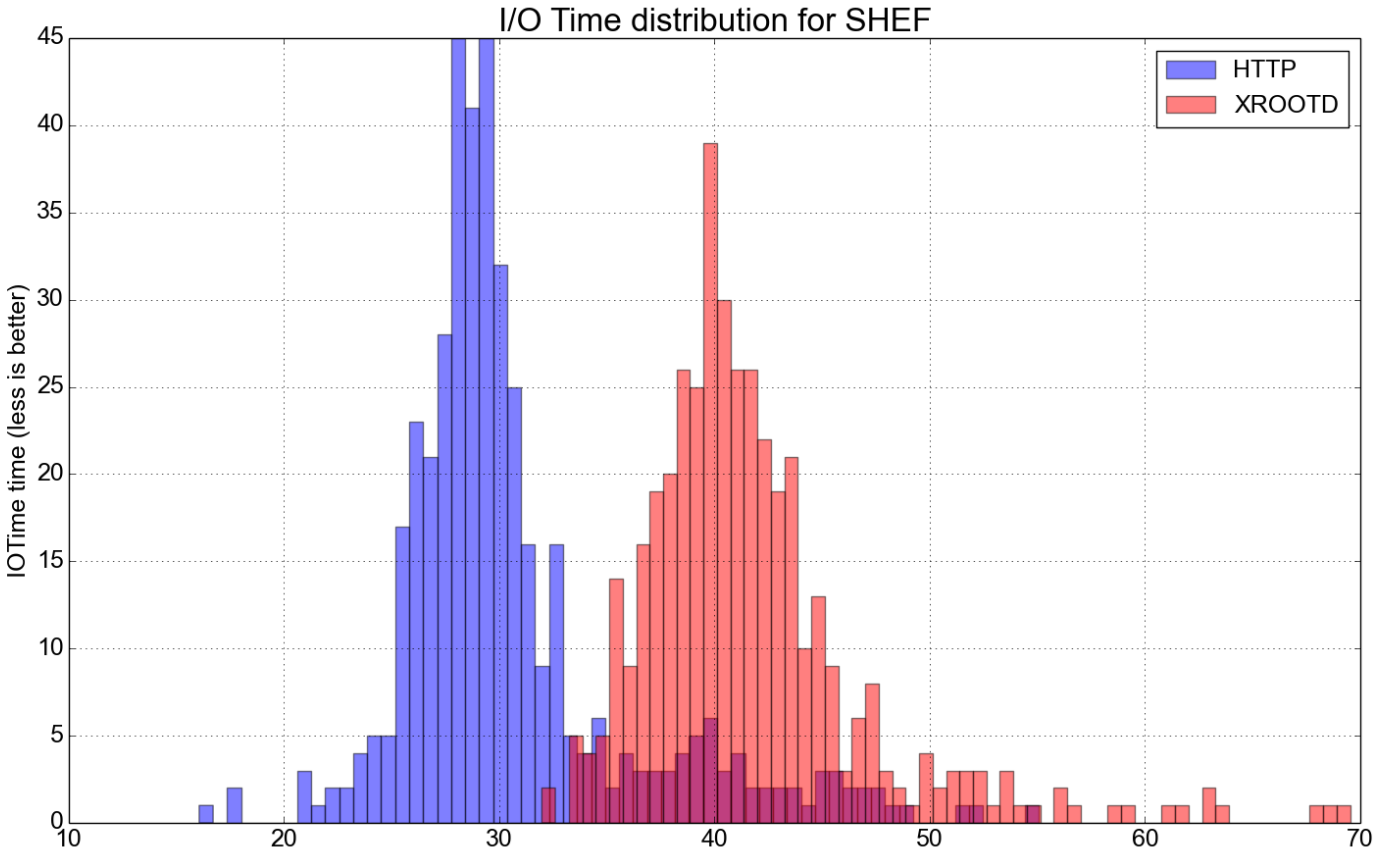
- Graphs show the **pure IO time** of these jobs
- Data averaged over many months
- Error bars represent the 95% confidence interval

Client at CERN, various servers

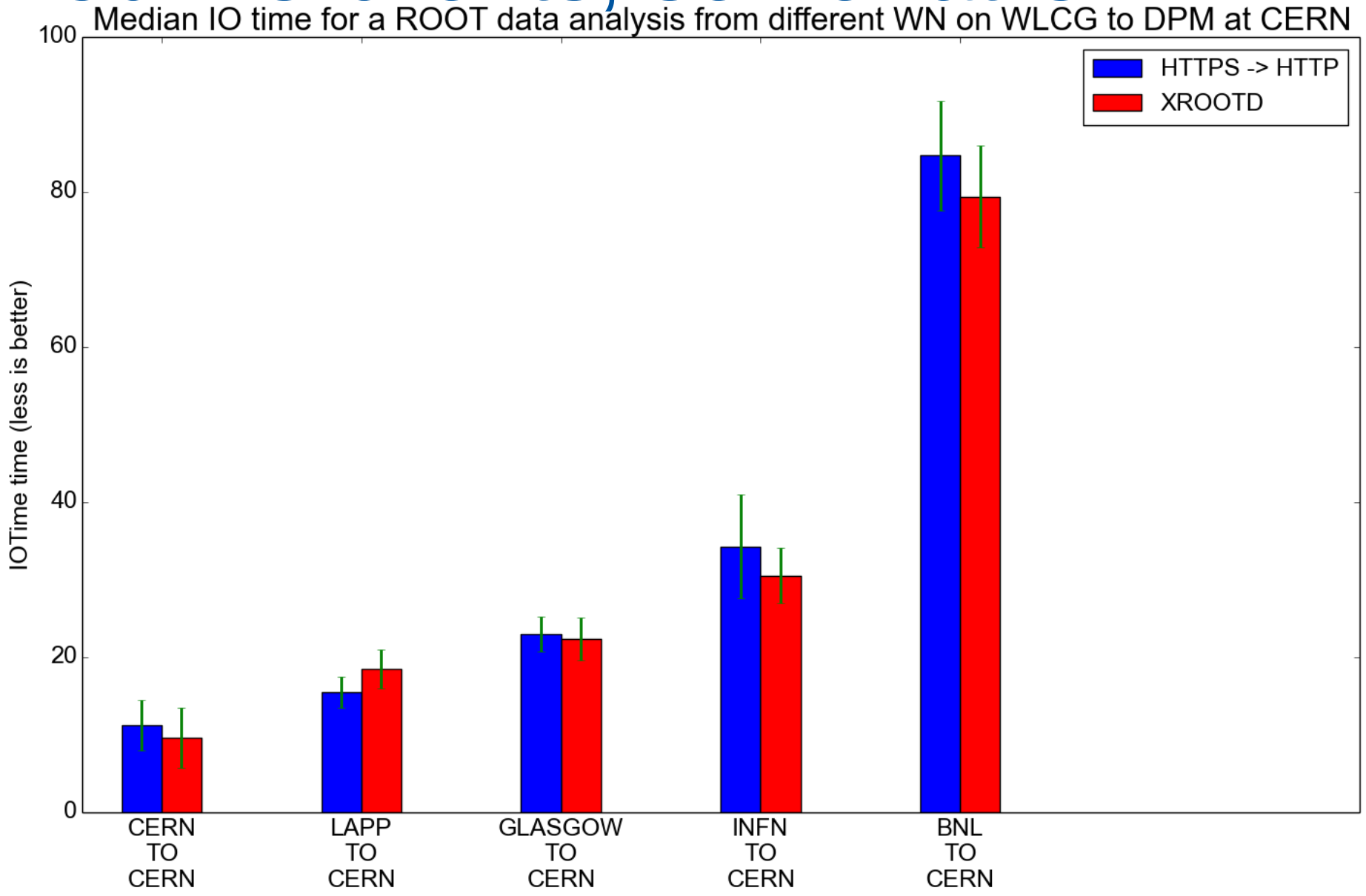


Site	RTT(ms)
CERN	0.5
RHUL	16.1
SHEF	23.5
TAIWAN	298
AUSTRALIA	321

IOTime distribution



50 HC clients, server at CERN



Access tests

- Bottom line: comparable again
- The difference in the IO time is a few percent
- The differences are not systematic
- The IO time itself is a small part of the total for smaller latencies (<50ms)
- Storage systems and TCP stacks are configurable
  - Contributes to the variation between sites

Conclusion

- We have compared the two protocols for metadata and for access, on LAN and WAN
- According to our test cases, HTTP and xroot access to DPM have equivalent performance
- While systematic differences may exist for a particular site, they do not exist for the infrastructure
- Our 50 parallel WAN jobs are likely not enough to differentiate concurrency support
  - Needs tougher tests able to stress more the servers