HTTP Ecosystem & Federation Brainstorming

Oliver Keeble

Alexandre Beche, Fabrizio Furano, Stefan Roiser

22/11/2013

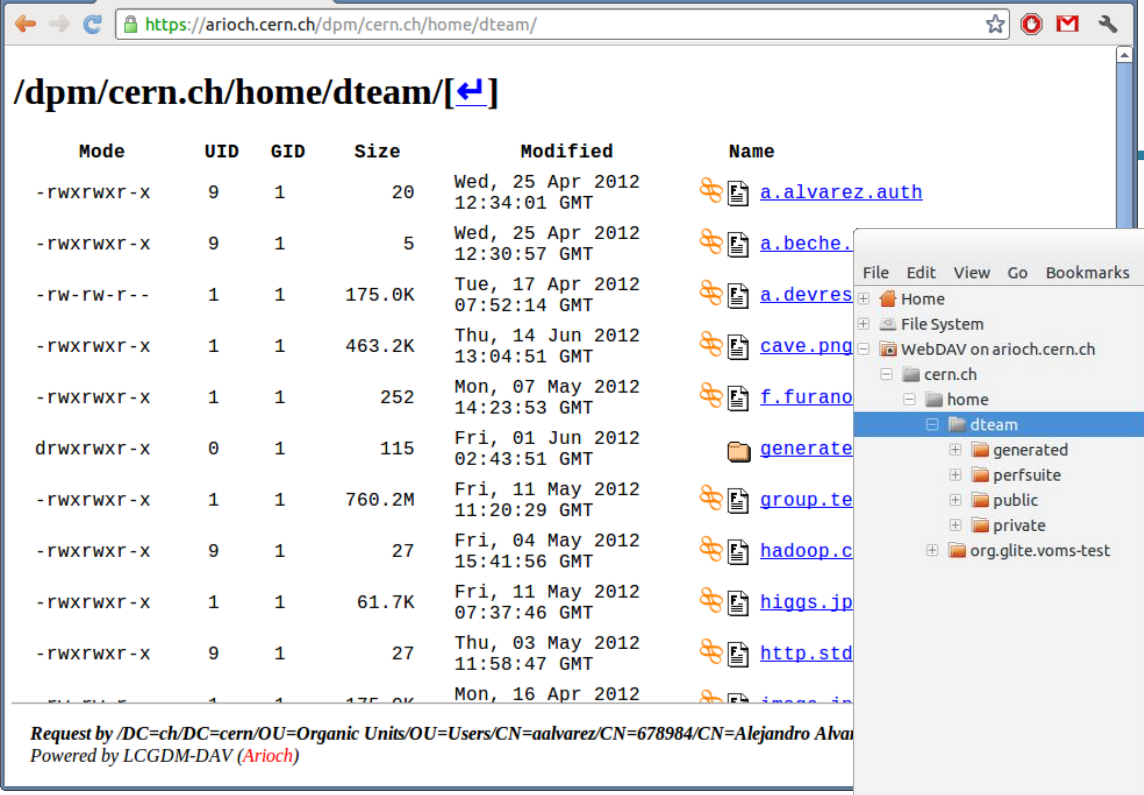
Ecosystem components

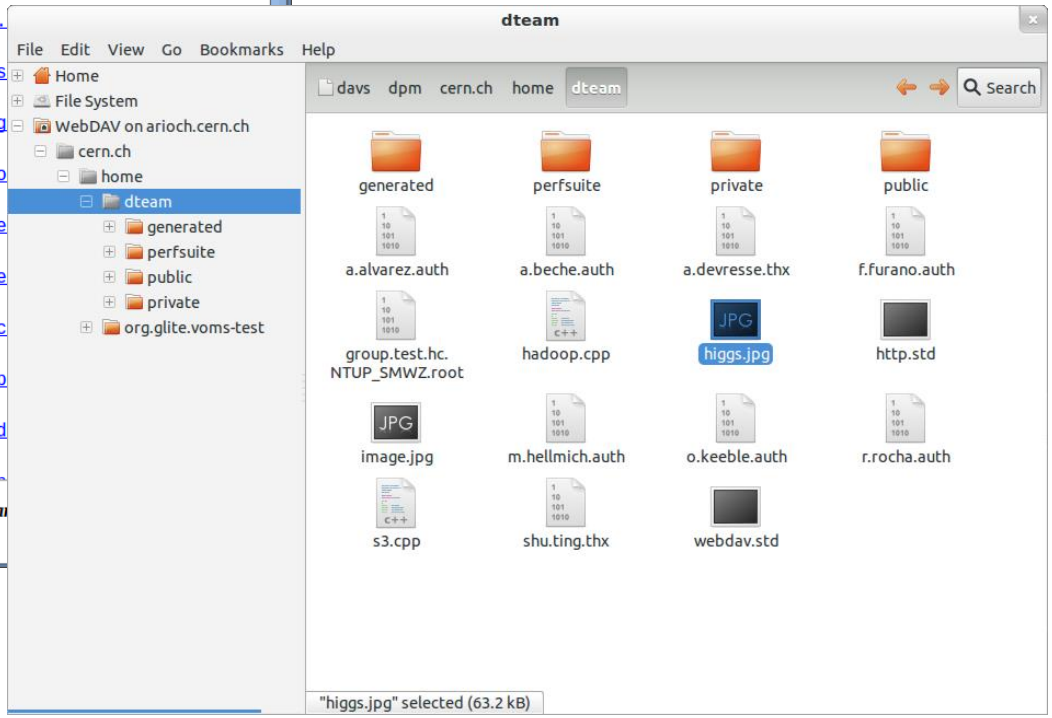
- HTTP/DAV support in storage
  - dCache
  - DPM
    - LFC too
  - EOS
    - “will adapt xrd-http later” – different solution for now (NGINX proxy)
  - StoRM
    - http://italiangrid.github.io/storm/documentation/webdav-guide/
  - Xrootd
    - xrd-http currently being integrated into xrootd 4 (to be released)

Ecosystem components

- Access library
  - Gfal2 support
  - Davix – access library addressing shortcomings of existing clients
    - NB – existing clients will still work within their scope!
    - TDavixFile for ROOT
      - Performance numbers are promising…
- Transfer
  - FTS3 support
  - 3rd party copy implemented by DPM (dCache forthcoming)
- Federation

TTP / DAV





- A browser click will download your file. Also…
- `> aria2c https://fed.cern.ch/mydata?metalink`
  - Parallel download from multiple replicas

DynaFed

- Dynamically federates HTTP endpoints
  - Including other catalogues
- Fast in-memory namespace cache
- Transparent redirection for clients
  - Closest replica chosen (geoip)
  - Other info sources could be integrated (eg perfsonar)
- Officially released and under evaluation
  - EUDAT, Victoria (CA)

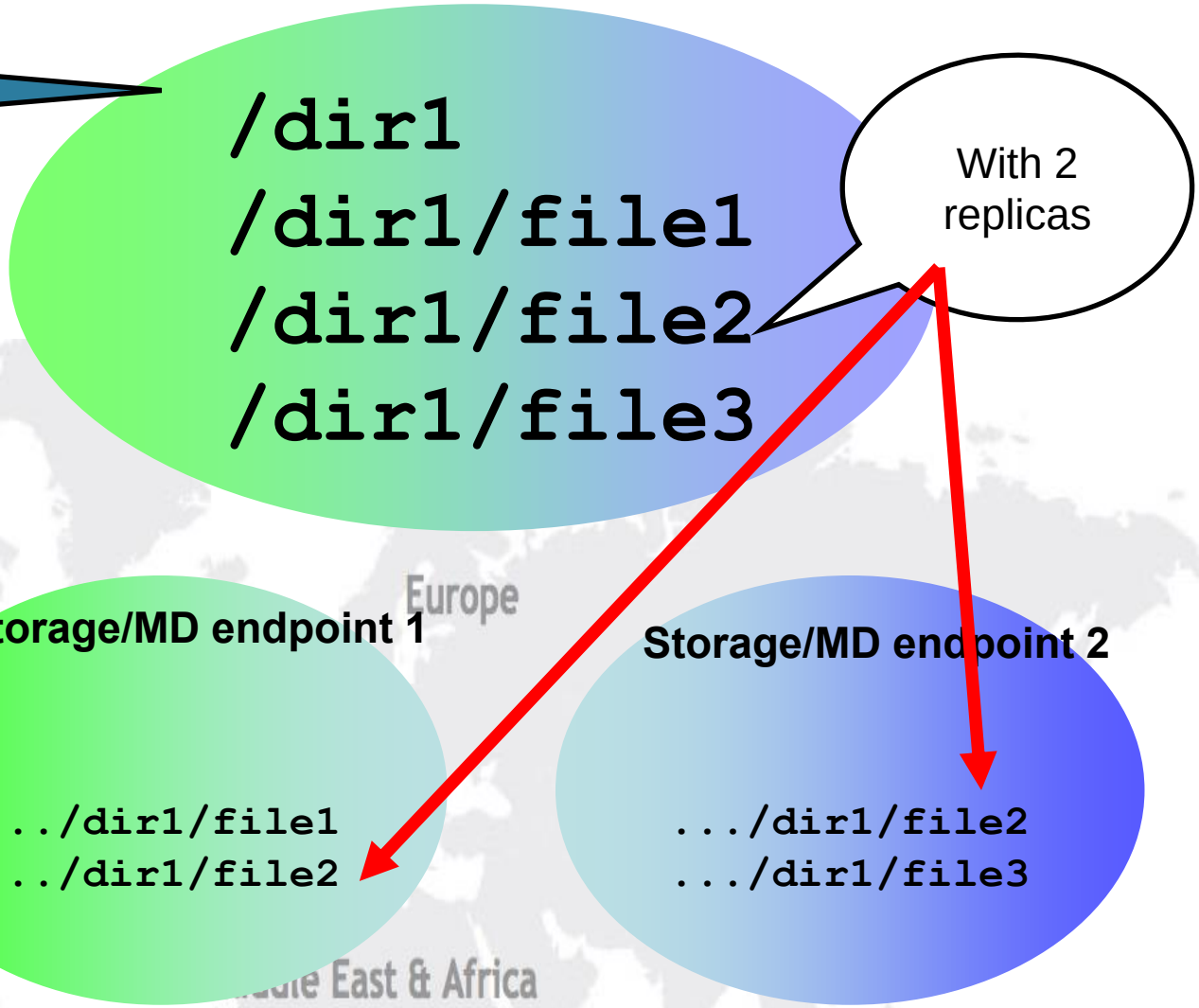
Aggregation

This is what we want to see as users

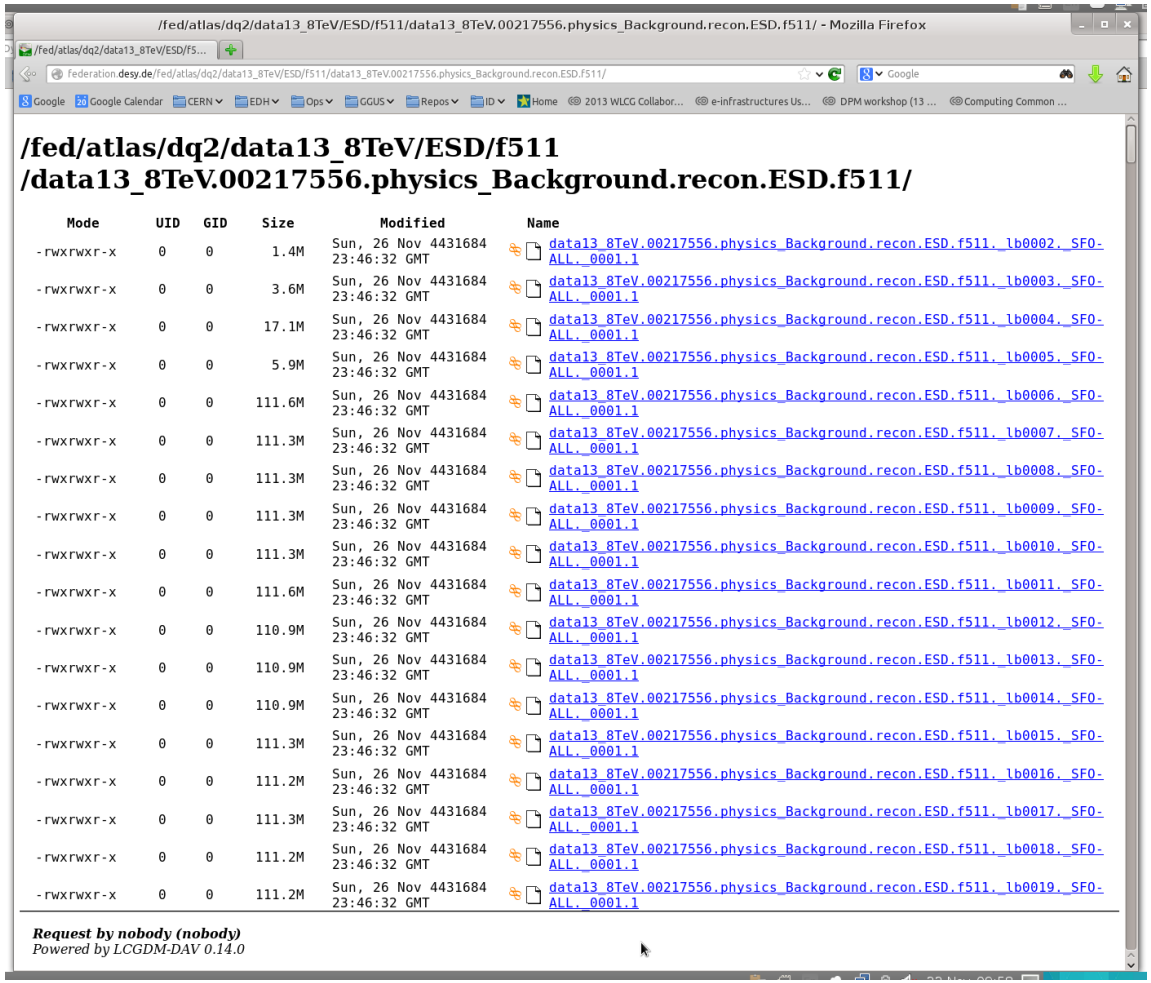
Sites remain independent and participate to a global view

All the metadata interactions are hidden and done on the fly

NO metadata persistency needed here, just efficiency and parallelism



http://federation.desy.de/fed

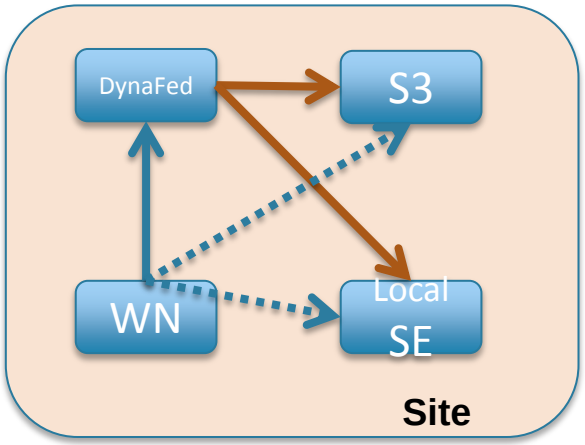


Use Case #1: Storage sharing

- Two well connected sites can share a dataset
  - Esp effective if each site concentrates on a subset but needs access to the whole
- Advantages
  - Transparent access via clients
  - Closest replica chosen in case of duplicates
  - Efficient use of storage
- Can be scaled up to national level or beyond
  - Additional monitoring work would be required (both for endpoints and federator).

Use Case #2: Integrating Cloud Resources

- A site wishes to elastically expand storage to support a short campaign
  - S3 storage is provisioned (privately or publically).
  - Data is imported
- Federation provides an integrated view of existing storage and elastic storage through namespace integration and translation



Use Case #3: Integration with caching

- eg Pure cache site
- Reuse existing cache tech
- Much easier if data is “group readable”
- Integration with CDN