



SDC Projects

Markus Schulz



Areas

- Data access and storage
 - HTTP Ecosystem
 - Caching
- Data transfers and data management
- Namespace and Metadata

Why HTTP?

- HTTP as an additional protocol opens access for other communities
- Many building blocks available and emerging
 - supported by a very large user community
- “Future Safe”
- Support for data federation
 - easy integration with cloud storage (S3)

Ecosystem components

- HTTP/DAV support for most storage systems
 - dCache. DPM (LFC), StoRM
 - EOS
 - “will adapt xrd-http later” –for now (NGINX proxy)
 - Xrootd
 - xrd-http currently being integrated into xrootd 4

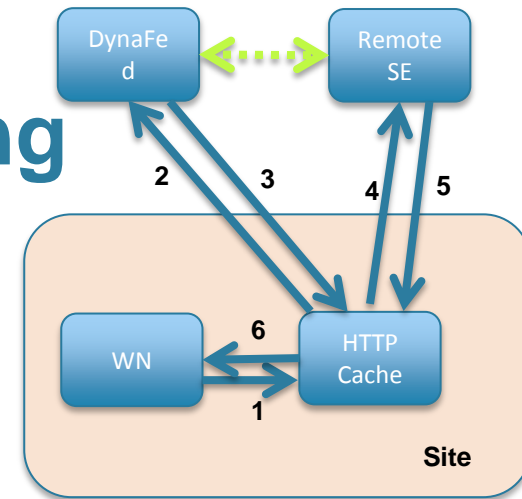
Ecosystem components

- 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 soon)
- Federation

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)

Integration with Caching



- Vision:
 - Stateless Sites (cache only sites)
 - **low maintenance**
 - (high/good) performance, efficient use of disk space
- On-going developments
 - SDC investigates Squid and Varnish
 - early stage
- Questions to be answered:
 - 10% extra WAN translates to XX% local access
 - for which use cases does it work?
 - Full file / block caches ..
 - Object stores and caching?

Data Transfer

- FTS-3
 - WebFTS
 - web icing on the existing FTS-3
 - almost done, usable prototype exists
 - Shrink Wrapped Endpoint
 - export and import to and from the Grid
 - Laptop, non grid cluster, non grid components
 - HTTP and cloud storage support
 - S3 like and personal storage (owncloud, dropbox)
 - Interesting for non HEP communities

Data Handling

- FTS 3
 - file management operations
 - example: “mega” rm
 - whatever you do on a command line
 - but: scheduled and on scale
 - removing 100k files etc.

Scalable MetaData Rich Namespace

- Scalability of name spaces has been addressed by CMS and ATLAS
 - trivial catalogue (algorithmic)
- Alice and non-HEP users have a need for managing their namespaces and meta data
- Work didn't start, some ideas have been discussed