

Plans for LFC

Simone Campana
CERN IT-ES

- LFC in DDM is used to map a GUID to a SURL
 - i.e. map the logical file to a physical one
- DDM design: one catalog per endpoint
 - LFC/SE consistency up to the site
 - LFC could have even been merged with SE
- In reality
 - Maintaining an LFC was load for small sites
 - LFC/SE consistency non trivial (and requires dedication)
- The current deployment model: one LFC per T1 serving the all cloud
 - There are exceptions (US T2s)

- Each LFC is a single point of failure for the cloud
 - Replication scenarios are non trivial
- Lots of duplicated information between catalogs
 - LFC vs LFC
 - LFC vs DDM
 - kept lazily consistent at the client level
- LFC/SE consistency non trivial
 - Many SEs for one LFC

- Merge all LFCs into one
 - Its natural location would be CERN
 - In the same rack as DDM Central Catalogs
- Replicate the master LFC to one (more) T1s
 - Provides a live backup in case of disaster
 - Slave can become master at any point
 - Can be used for other purposes
 - Generate catalog dumps for example
- In a longer term, LFC and DDM Central Catalogs can be merged
 - Many commonalities for file tables

- Performance: can one LFC sustain all the load?
 - New developments in LFC (SSL sessions) will help
 - Still...
- Migration should be gradual
 - One cloud at the time (start with 5% T1s)
 - At each step we should carefully evaluate the load on LFC and the error rate from SS and Panda
- Asynchronous registration would improve the workflow
 - Can Panda server register the outputs?
 - Can we use ActiveMQ?

- Scalability: would a central LFC scale with the number of entries
- Partitioning at the logical file level (GUID)
 - Per time interval
 - 90% of ATLAS GUIDs embed creation time information
- Partitioning at the replica (SURL) level
 - Per site
 - Will allow to generate fast dumps

- Need to agree about DB service in CERN IT
 - Resources, manpower impact ..
- Need to write tools for migration
 - Import and merging
 - Both from Oracle and MySQL backends
 - In a “rolling” fashion to minimize downtime impact
 - LFC experts can help on this
- Need some support from T1s at the time of the migration
 - Provide dumps, access, ...
- With a well thought plan, estimate less than a couple of days of downtime per T1 during migration

- Consolidating LFCs would
 - Simplify the scenario for disaster recovery and prolonged downtimes
 - Simplify the DDM architecture
 - Simplify operations
- We have to be careful to
 - Performance impact
 - Scalability impact
 - Migration operational impact
- This is why the proposal is a staged rollout and constant feedback loops
- In order to get support from service providers and developers we need a concrete and well defined plan
- Can we agree?