Experiencing DPM distributed setup and caches in Italy

ALESSANDRA DORIA, BERNARDINO SPISSO – INFN NAPOLI

ELISABETTA VILUCCHI – LABORATORI NAZIONALI DI FRASCATI

ALESSANDRO DE SALVO – INFN ROMA1

DPM deployment in Italy

- 4 production DPM systems in ATLAS
  - 3 Tier2s at INFN Naples, National Labs of Frascati (LNF) and INFN Roma1 (DPM 1.9)
  - 1Tier3 at INFN Cosenza DPM (1.9)
- 2 Testbeds for EPEL-testing release at LNF and Roma1:
  - DPM 1.10
    - Correctness of the installation and puppet configuration validated (some bad dependencies fixed)
    - Checks on quotatoken and space reporting
    - Functionality tests
- **Testbed for advanced features like volatiles pools used as caches and distributed setup among Naples, LNF and Rome**

Motivations

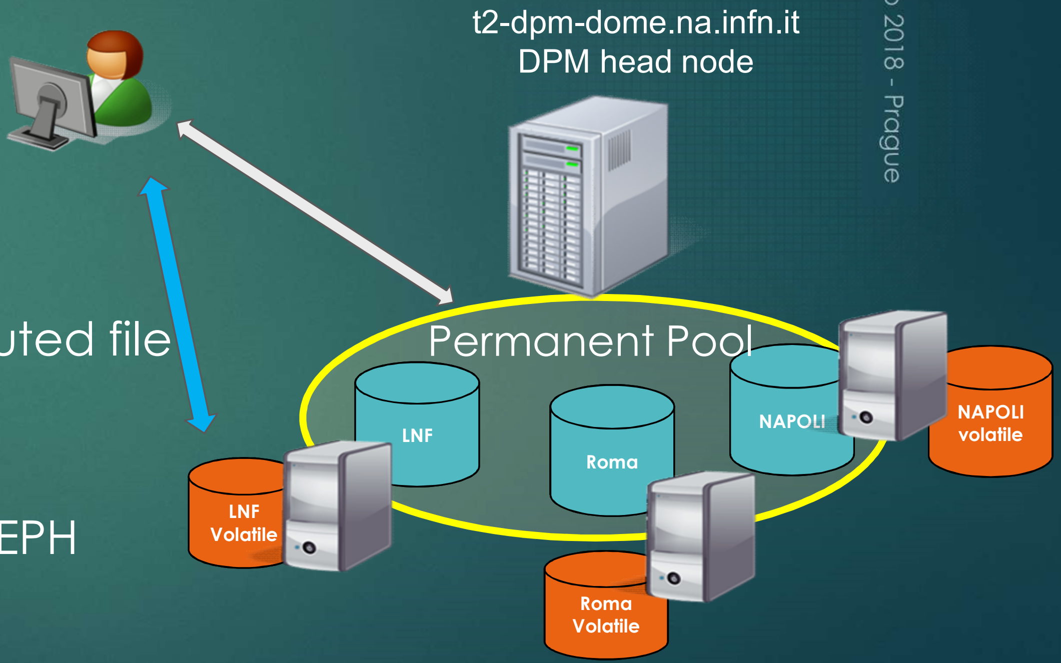
- The LHC experiments (ATLAS), WLCG and funding agencies have started a process of optimization of the storage hardware and human resources needed for storage operations .
- The keywords in Data Lakes R&D WLCG project are:
  - Common namespaces
  - Distributed storage and redundancy
  - Co-existence of different QoS (storage media)
  - Geo-awareness
  - Usage of caches
- DPM is used since 2006 in 3 out of 4 ATLAS Tier2 in Italy:
- Our interest is to keep using DPM in the future and to verify how it fits some/all of this optimization requirements

Specific use cases

- There are different use-cases for our tests:
  - Small sites can become «diskless» with a local cache
    - Simplified local storage management (no head node and DB)
    - Site users can access data in the local disk in caching mode
  - Using a distributed storage with a single end-point
    - Single common namespace
    - Simplified operations from the experiment point of view
    - Try to implement redundancy

The distributed prototype

- The testbed is installed with the latest releases available in EPEL-testing:
  - DPM release 1.10.2
  - DOME is enabled
  - Gridftp redirection is enabled
- Head Node and DB in Naples
- 3 Disk Nodes, one per site.
- One permanent pool, made of distributed file systems
- 3 volatile pools, one per site
- The file systems in Rome are built on CEPH



- dmlite-shell > qryconf

```
POOL staticPool DEFSIZE 2147483648 GC_START_THRESH -1 GC_STOP_THRESH -1 DEF_LIFETIME -1 DEFPINTIME -1
MAX_LIFETIME -1 MAXPINTIME -1 GROUPS  FSS_POLICY  GC_POLICY  MIG_POLICY  RS_POLICY  RET_POLICY  S_TYPE -
            CAPACITY 3.84TB FREE 874.94GB ( 22.3%)
      atlas-dpm-pool-02.roma1.infn.it /data1 CAPACITY 2.00TB FREE 580.87GB ( 28.4%)  ONLINE
      t2-disk01.na.infn.it /data/t2-disk01-static CAPACITY 1.82TB FREE 294.07GB ( 15.8%)  ONLINE
      atlaswn024.lnf.infn.it /data/data01 CAPACITY 19.99GB FREE 20.00kB ( 0.0%)  ONLINE
POOL lnf-volatile DEFSIZE 2147483648 GC_START_THRESH -1 GC_STOP_THRESH -1 DEF_LIFETIME -1 DEFPINTIME -1
MAX_LIFETIME -1 MAXPINTIME -1 GROUPS  FSS_POLICY  GC_POLICY  MIG_POLICY  RS_POLICY  RET_POLICY  S_TYPE V
            CAPACITY 19.99GB FREE 6.57GB ( 32.9%)
      atlaswn024.lnf.infn.it /data/data02 CAPACITY 19.99GB FREE 6.57GB ( 32.9%)  ONLINE
POOL na-volatile DEFSIZE 2147483648 GC_START_THRESH -1 GC_STOP_THRESH -1 DEF_LIFETIME -1 DEFPINTIME -1
MAX_LIFETIME -1 MAXPINTIME -1 GROUPS  FSS_POLICY  GC_POLICY  MIG_POLICY  RS_POLICY  RET_POLICY  S_TYPE V
            CAPACITY 1.82TB FREE 1.77TB ( 97.5%)
      t2-disk01.na.infn.it /data/t2-disk01-volatile CAPACITY 1.82TB FREE 1.77TB ( 97.5%)  ONLINE
POOL roma1-volatile DEFSIZE 2147483648 GC_START_THRESH -1 GC_STOP_THRESH -1 DEF_LIFETIME -1 DEFPINTIME -1
MAX_LIFETIME -1 MAXPINTIME -1 GROUPS  FSS_POLICY  GC_POLICY  MIG_POLICY  RS_POLICY  RET_POLICY  S_TYPE V
            CAPACITY 2.00TB FREE 1.96TB ( 97.8%)
      atlas-dpm-pool-02.roma1.infn.it /data2 CAPACITY 2.00TB FREE 1.96TB ( 97.8%)  ONLINE
```

Using DPM volatile pool as local caches

- DPM offers the possibility to fill volatile pools with a custom mechanism.
- As a first implementation we decided to use the permanent pool as data source for the local caches.
- The pull script make a davix-get of the file from the permanent pool
  - When a file is required to the volatile pool, for the first time it's retrieved from the permanent storage, any other access finds the file locally in the cache
- More complex mechanisms can be implemented, we plan to interact with rucio to get any ATLAS file in the cache

Paths and Quotatokens

- A quotatoken defined for each cache pool, associated to a different path.

We decided to try to use a different domains in the path, to do this we had to:

  - comment dpm_defaultprefix in dmlite manifest xrootd.pp
  - create the paths manually

- Users can address different paths to local cache or permanent storage:

/dpm/fed-t2.infn.it/home/atlas/ATLASDATADISK/file1
and
/dpm/lnf.infn.it/home/atlas/ATLASDATADISK/file1

Will be the same file, read from the permanent pool or from LNF cache.

PERMANENT POOL

Token Name: fed-t2-static
Token Path: **/dpm/fed-t2.infn.it/home/atlas**
Token Pool: staticPool

CACHE POOL at LNF

Token Name: lnf-volatile-quota
Token Path: **/dpm/lnf.infn.it/home/atlas**
Token Pool: lnf-volatile

CACHE POOL at ROMA

Token Name: roma1-volatile-quota
Token Path: **/dpm/roma1.infn.it/home/atlas**
Token Pool: roma1-volatile

CACHE POOL at NAPOLI

Token Name: na-volatile-quota
Token Path: **/dpm/na.infn.it/home/atlas**
Token Pool: na-volatile

Cache flush

- dpm-qryconf shows a large number of parameters associated with the pool

```
POOL Inf-volatile DEFSIZE 2.00G GC_START_THRESH 10 GC_STOP_THRESH 20
DEF_LIFETIME 1.0h DEFPINTIME 1.0h MAX_LIFETIME 10.0h MAXPINTIME 10.0h
FSS_POLICY maxfreespace GC_POLICY lru RS_POLICY fifo GIDS 0 S_TYPE V
MIG_POLICY none RET_POLICY R
```

There are several paramenters that seem to be related to volatile pools flush: threshold, lifetime…

- Actually we found only one condition that causes the removal of the oldest files in a volatile pool:
  - Used Space > Total Space – DEFSIZE

- Is there a way to make file pinning?
- Can lifetime be defined?

Open issues

- Several performance test are needed:
  - What are the minimal netwok requirements for the distributed setup to work without problems?
  - Under which conditions the usage of a local cache is advantageus compared to remote file access? This is difficoult, it depends on how many time the same data files are re-used in a site
- Test difference in performance using local file systems and CEPH FS
- We need deeper testing for multiple domainpath

Conclusions

- Very simple to add new storage from distributed sites to a single head node.
  - DPM installation and configration can be done centrally with the same puppet/Foreman master
  - Small sites that have some disk resources can make them available to the community with a minimal effort
- Would it be possible to merge existing storages in a single namespace with a reasonable effort?
- Volatiles pools as caches can be used easily and effectively, but we have to understand the real user requirements.
- Further results will be presented in a poster to CHEP18.