From: **Xavier Espinal Curull** xavier.espinal@cern.ch
Subject: Some notes from today ESCAPE meeting
Date: 18 April 2019 at 11:50
To: Simone Campana Simone.Campana@cern.ch

# ESCAPE WP2 Fortnightly 18th April 2019

**Participants:** Fabio, Frederic, Guido, Manuel, Martin,Raffaele, Diego, Ron, Rosie, Simone, Stephane, Tommaso, Yan, Xavier

## Intro (Simone)

Phone meetings: Vidyo to be used from now on as a main tool for the phone conference meetings. Room booked for project length time: https://indico.cern.ch/event/814928/

WP2 workshop: from Monday 1st July afternoon to Wed 3rd of July after lunch, with one day shared with WP5 (TBC)

## Caching (Daniele)

- Besides the ongoing activities in WLCG/DOMA for data access and caching. Daniele gave an overview of caching activities in some EU projects like INDIGO and XDC and National Initiatives.
- Characteristics, Topology and Goals:
  - Cache driven by clients with no central management (cache content driven by the clients)
  - Ge-distributed caches
  - common namespace
  - The Goal to leverage national network: optimise size of stored data, add a layer of unmanaged storage and reduce redundancy and operational costs
- Proof of concept ongoing with distributed caches to asses impact of cache layer on a regional basis. Measuring: CPU efficiency, disk space and operational effort.
- CPU efficiency with remote reading: 2018 CMS analysis workflows on Italian T2s showed that
  - 15% of CPU time with remote I/O
  - 1/3 of walltime on jobs with remote reading
  - These numbers are in line with global CMS values
- Stored data: how many data is need vs. what is stored for analysis workflows (mini-AOD)
  - 20% of data is moved without reason.
- Introducing a cache layer:
  - Narrowed CPU efficiency reducing latency
  - Optimise data volume stored on disk
- Distributed cache implementation prototyping a cache model for DCC (Data and Compute Center in the Strawman model terminology)

    - Enabling storage-less T2 (CCNC - Computing Centers with No Cache)
    - Using cache for geo-distributed layer approach
    - Sites involved: CNAF, Bari and Legnaro working setup since mid-2018 and integrated with CMS workflows
        - Q(Xavi): is read-ahead enabled at the XCache nodes ?
        - A(Diego): Yes
        - C(Tommaso): Read ahead also active by default at root level with TTreeCache at job level, similar effects expected.
        - C(Xavi): TTreeCache works at WN/job level, XCache in this case seems to run on a dedicated (data can be reused)
- Deployment on cloud:
    - Opportunistic cloud/HPC resources where storage is not necessarily available
    - Caches offer: ephemeral storage and optimised WAN access
    - Recipes for XCache clusters on demand:
        - Ansible for bare metal, Marathon/Mesos and Kubernettes.
    - 2 volunteer use cases for CMS analysis (DODAS EOSC-Hub)
- Next steps:
    - Scale-up of the national testbed and synergies with other national projects.
    - Combined QoS and Caches
    - Operational intelligence initiatives to have an impact on operational costs.

- Q(Manuel) is reading restricted to xroot protocol?
    - A(Tommaso/Daniele): Based on xrootd but caches are http enabled.
    - C(Manuel): Astronomy data is in root format, much smaller volume but more sparse, users more spread. Seems a very interesting way to prototype in a different domain from HEP.
- Q(Manuel) is there any Access Control in place?
    - A(Tommaso/Daniele): Yes. CMS credentials are used. Standard VOMS.
    - C(Manuel): Access control is a key point. People is very sensitive to have their data “stolen”. VOMS is enough to setup a prototype.
- Q(Simone) Collaborations/projects that not use root format can we still make this as a valuable solution (ie. SKA)
    - A(Daniele): http reading is OK for serving data.
    - C(Tommaso): https solution was explored (in Dynafed).
    - C(all): Client access and data movement/replication can be separated in terms of authentication.

- Q(Rosie) How much will this scale up for big files ?
- Q(Rosie) Possibly need to adapt algorithms to be able to have an “ordered processing”
- A(Tommaso) XCache can bring you anywhere in the file with pointers.
- C(Simone) Posix access requirements put a lot of constraints on storage, either is local on HDD or a shared file systems across the WNs which is a challenge by itself.
- C(Fabio): LSST file sizes are not that big, rather small fro HEP standards (100MB). But depending on the phase need one file or multiple (ie. 200) files that compose the image. Also pushing to use standard protocols (http/https) but as of today we need POSIX.
- C(Tommaso): positive the current xrootd/posix to be good enough to start looking at things.
- Q(Fabio): This needs FUSE on the WN.
- Q(Tommaso): Full file or random access within the file?
- A(Fabio): Random access but at the end reading all the file.
- C(Frederic): CTA need to be processed once a year. No use case to cache raw data.
- A(Simone, all): Need to have in mind that Caching and XCache have three layers: pure-caching, latency hiding and as an Edge Service.
-