T4.2 report

Data Management for extreme scale computing

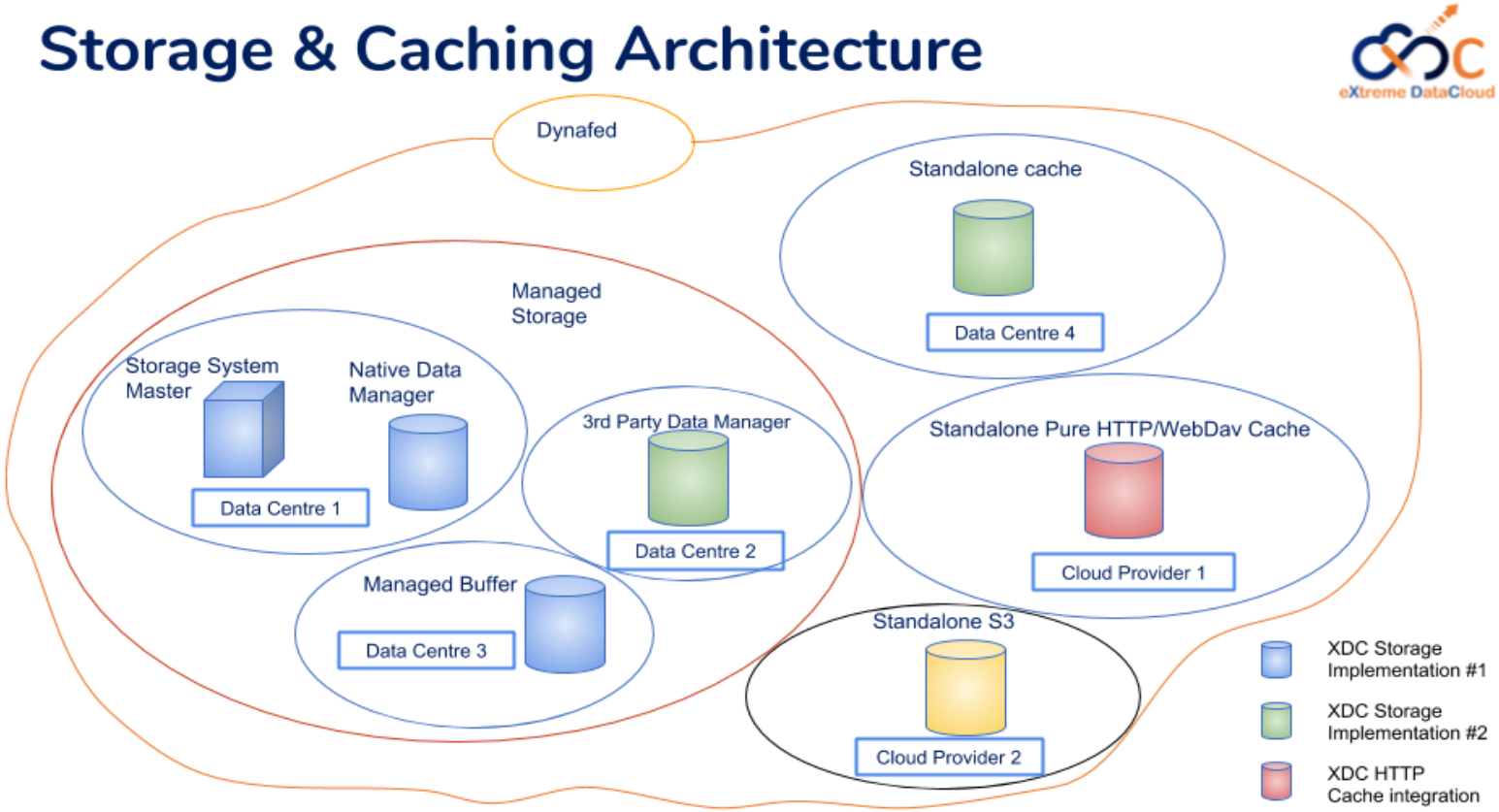
Oliver Keeble

on behalf of T4.2 participants

eXtreme DataCloud is co-funded by the Horizon2020 Framework Program – Grant Agreement 777367

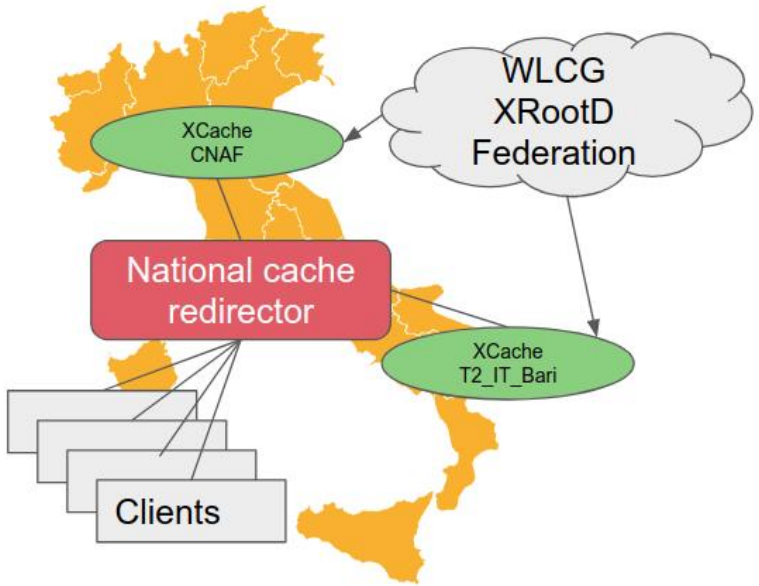
T4.2 : “Smart Caching”

- **Goal: maximise the accessibility of data to clients while minimising global infrastructure costs.**



INFN : XCache

- National Cache Federation
  - Distributed cache federation
    - Currently 2 sites
  - Used by sites without local data
  - Integrated into CMS
    - metrics being collected
  - Next steps
    - increase scale
    - employ a predictive model



- Automated Standalone Deployment
  - DODAS automated deployment
  - Kubernetes and/or Mesos/Marathon orchestration
  - Demonstrated with 2k CMS jobs on OpenTelekomCloud

INFN : HTTP(S) Cache

- HTTP(S) caching solution based on nginx
  - VOMS auth provided by a plugin
  - Upstream endpoint selection based on VO information
- Client contacts nginx which either retrieves (miss) or serves (hit) the data
- Cache operates with a service credential
  - No delegation
- Smarter caching algorithms may be contemplated
  - nginx plugin mechanism

dCache

- Storage Events (notifications)
  - SSE driven notifications have been added to dCache
  - Adding storage-event support in storage systems can bring numerous benefits
    - Smart Caching
      - remote parties can learn about changes within a storage system
      - A cache could take pre-emptive action
        - deleting cached data if the parent items are deleted
        - fetching "interesting" data before the client makes a request
    - Storage adoption
      - Running a remote storage system as part of a larger custodial system
      - This would allow the remote system to be accessed independently of the custodial system
    - Orchestrator integration, Federator integration …
- Locality information [forthcoming]
  - Do “data lakes” need a data locality interface?
  - The use-case : a dataset is stored in "the data lake". The client wants to know to which compute centre(s) it should submit its work to reduce the access latency.
- Storage adoption [forthcoming]

CERN : EOS XCache

- Integration of XCache as a caching solution for EOS
  - Reference scenario : XCache deployed at a remote centre to accelerate its local CPU. Front end is HTTP/xroot, backend communication is xroot.
  - As EOS is based on XRootD, this offers a number of opportunities for closer integration of the two systems
  - Identity forwarding plugin created whereby the cache can identify on whose behalf it is acting
    - Upstream storage can respond appropriately
  - Full system deployed on XDC testbed and integrated with VOMS
- Evaluation of further developments
  - write support
    - would be write-through
  - ACL synchronisation

CERN : EOS Storage Adoption

- EOS can now adopt
  - External storage systems
    - Through an S3 or a WebDAV interface
    - Demonstrated with dCache/WebDAV
  - External data
    - Data already present on a system described above can be incorporated into EOS
    - It can then be replicated, moved, managed in the usual way
      - Can even be removed from the original storage while preserving access
- At present, EOS takes over management of the storage system
  - Independent access is… possible but highly discouraged.
  - This can be addressed through notifications