Smart caching, dCache and Data Lakes

eXtreme DataCloud

Data Management for extreme scale computing

Paul Millar

paul.millar@desy.de

DOMA/ACCESS Meeting
2019-03-12

European Commission

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

https://indico.cern.ch/event/769502/

Geographically distributed dCache

- Over a decade of experience in distributed storage:
  - NDGF is a distributed dCache, spread over five countries.
  - AGLT2 is a distributed dCache, spread over two campuses.
- dCache also supports (automatic) **caching** of data:
  - In AGLT2, reading data only present at the other campus results in the data being cached locally.
  - Campus-local cache is limited, with an LRU-like algorithm used to control which data is kept.
- dCache can already provide **protocol-based cache control**
  - For example, caching data if accessed with NFS, but don't cache if data is requested with HTTP or FTP.

A demonstrator for DOMA

- The dCache team, in collaboration with dCache admins at DESY Hamburg and Zeuthen, and the Kurchatov Institute, are building a **dCache DOMA demonstrator**.
- Storage is located in Hamburg, Zeuthen and Moscow.
  - Sites have roughly the same available capacity: ~100 TiB.
  - Initially deployed as a hub-and-spoke topology, with Hamburg as a central storage and Zeuthen and Moscow caching data.
- Two “satellites” allows for comparison with different delays:

  → **Zeuthen** (RTT: ~5 ms), → **Moscow** (RTT: ~70 ms)

Demonstrating what works, right now

- Initial deployment is with regular, **released version of dCache**: 5.0.x
- Support **protocol-based** caching decision
- Measure the effect of caching on various work-flows.
  - What are the overheads in a cache-miss?
  - What is the effective working set?
  - Can this be used to gauge whether cache is “large enough”.
- Configure how **writes** are handled: write-back or direct-write.

A spring-board for development

- Use the **observations** to drive new developments.
- What would improve Data Lake deployments of dCache?
  - Providing “off-line” access to cached data.
  - Optimise cache misses / manage data movement.
  - Enhanced decisions on whether to cache data (e.g., ARC/CLOCK/CAR).
  - Automatically populate caches, based on popularity or datasets.
  - Allow clients to warm-up caches.
  - Expose file’s cache locality to enhance scheduling decisions.
  - Explore a “full mesh” where caches take data from other caches.
- Provide an environment where these new solutions may be **tested**.

eXtreme DataCloud

dCache

Thanks for listening.

Any questions?