Using the Dynafed dynamic data federator as a Rucio storage element

Frank Berghaus
University of Victoria

much help from
UVic, TRIUMF, CERN-IT, ATLAS, Belle-II
Introduction & Motivation

- Distributed cloud system
  - cloudscheduler
  - In production for >8 years
  - User: DIRAC (Belle-II) or PanDA (ATLAS)
- Cloud Scheduler at UVic and CERN
- Cloud Resources:
  - In Canada, US, UK, Germany, Austria and at CERN
  - $O(10^3)$ cores – easy to add more
- CE: HTCondor & Cloudscheduler
- SE: dCache (UVic), EOS (CERN)
- Limited by remote access to storage
Dynafed: Redirect To Nearby Storage

- Dynafed redirects to close storage
- Operating three configurations:
  - Belle-II at UVic:
    - R/O access (production)
  - ATLAS at CERN:
    - R/W to cloud storage (dev)
    - R/W to grid storage (dev)
- Instances operated by others:
  - data-bridge at CERN for *@home
  - Belle-II Dynafed at INFN
  - RAL ECHO
- Part of a WLCG Demonstrator
• With gfal2 support Belle-II will be able to use Dynafed as SE
• Workaround for Belle-II DIRAC:
  – gfalFS provides fuse mount within Linux directory tree:
    gfalFS -s ${HOME}/b2data/belle davs://dynafed02.heprc.uvic.ca:8443/belle
  – Jobs access Belle-II data from “local” directory ~/b2data/belle
• In production for the last two MC campaigns

• Easy addition of new endpoints
  – Added traditional Belle-II SEs while transferring new input data sets to own Endpoints:
    • Instant access to new files without configuration change on jobs/workers
• gfalFS and Dynafed work well for reading input data
  – Output is still written to UVic dCache using SRM
  – Waiting on gfal2 to be added to Belle-II offline computing

• Load is balanced across co-located storage endpoints
  – MC campaign: longer running jobs request at least one file
  – User analysis: short jobs request one file
  – Skimming & merging: shorter josb request multiple files
  – ~3000 job slots → 35TB per day
• Easy and effective network usage
  – Same configuration for all workers (6 separate clouds are used for Belle-II)
  – With same files used by many jobs network transfers stay local
Dynafed as ATLAS Storage Element

- **Grid Rucio Storage Element:**
  
  `dynafed-atlas.cern.ch/data/grid`
  
  - CERN (EOS), LRZ (dCache), ECDF (DPM)
  - CERN-EXTENSION/GridDisk

- **Cloud Rucio Storage Element:**
  
  `dynafed-atlas.cern.ch/data/cloud`
  
  - CERN (CephS3)
  - CERN-EXTENSION/CloudDisk

- **Authenticate with X.509+VOMS:**
  
  ```
  glb.allowgroups[]: "/atlas/*" /data rw1
  glb.allowgroups[]: "/atlas/Role=production/*" /data rlwd
  ```

- **Allow ATLAS Users to browse Dynafed by harvesting DNs from VOMS:**
  
  ```
  glb.allowusers[]: "/DC=ch/DC=cern/OU=Organic..." /data r1
  ```

- **Rucio supports and SEs expose HTTP+WebDAV**
Experience With ATLAS and Dynafed

- **Workload management:**
  - Functional tests run against Dynafed

- **Data management:**
  - Works:
    - Reading, writing, deleting
    - Redirection
  - Work in progress
    - Checksums
    - Third party copy
    - Space reporting/accounting

![Transfers to and from Dynafed](image.png)
Rucio, Dynafed, and Checksums

- **Mechanism:**
  - Grid: User is responsible, Want-Digest [RFC3230]
  - Cloud: Provider is responsible, Content-MD5 [RFC1544]

- **Algorithm**
  - Grid: ADLER32 [RFC1950], MD5 [RFC1321] (Rucio uses both/either)
  - Cloud: MD5 [RFC1321] only

- Rucio expects the grid *mechanism*
  - *Workaround:* Flag for Rucio not to request checksum from Dynafed

- Dynafed ongoing development:
  - On Want-Digest: call out to get checksum (if not in cache)
  - Cache checksum
  - *note*: hide implementation details

- In the pipeline – sometime this year
Third Party Copy – Cloud Storage

- Functionality released in December 2018
- On a copy COPY:
  - Redirect copy request, if supported
  - Else local call:
    - Default: gfal-copy
    - Note: if non-dynafed endpoint supports TPC it will push/pull
- Evaluating with DOMA-TPC

See Alastair’s presentation on RAL ECHO
See Robert’s presentation on DvNE
Dynafed Storage Plugin

- Issue with writing to Dynafed
  - Free space on endpoints unknown
- Query usage and quota from endpoints using script
  - Add results to cache
  - Generate JSON to inform Rucio
- Use:
  - WebDAV [RFC4331]
  - CephS3 r/o admin interface when possible
- Commercial providers don't provide quota
  - Query usage form billing
  - Manually set quota
- Work in progress: file list
Summary

- HPC workloads on distributed clouds works
- Dynafed shown to provide data access for $O(10^3)$ workers
- Dynafed as a Storage Element is work in progress
  - Not be the design purpose of Dynafed
  - Work done will be interesting for others, hopefully :-)
- The code-camp and the contribution work flow are great!
Thank You!