dCache, Bridging Between Sync-n-share and Compute Cloud

Paul Millar

LSDMA „Community Forum“ Spring 2015
Need a sync-n-share service at DESY

- **Requirements:**
  - Easy to use,
  - Store everything at DESY,
  - Integrate with existing infrastructure.

- **Anticipated future usage:**
  - change data between syncing and non-syncing storage,
  - like Amazon, provide different QoS with different costs,
  - share data without syncing,
  - 3rd party transfers between sites,
  - direct access to sync space from compute facilities.
How we solved it at DESY

• Looked around, chose two open-source projects:
  • **dCache**: powerful managed storage system
    Integration with scientific data life-cycle;
    “Hot” data can be stored on SSDs, “cold” on cheaper HDDs, “archive” tape;
    … but no sync and share facilities.
  • **ownCloud**: popular front-end
    Our collaborators adopting ownCloud makes it more attractive;
    … but assumes storage is managed.

• Combining these two gives DESY the best of both worlds:
  dCache is mounted on servers with **NFS v4.1/pNFS**, running community edition ownCloud.
  Integrated with DESY Kerberos, LDAP and “Registry”.
What is dCache?

LHC data stored on each storage system

- dCache (96 PB)
- DPM (34 PB)
- EOS (0 PB)
- StoRM (20 PB)
- CASTOR (14 PB)
- BeStMan (7.6 PB)
- Globus FTP (6.1 PB)
- ARC (0.01 PB)
- xrootd (22 PB)

Source: BDII (2014-11-14)

Collaborations

- Student mentor programme
  - Hochschule für Technik und Wirtschaft Berlin
  - 3 students

Core team

- DESY: 5 FTEs
- Fermilab: 2 FTEs
- neic: 1.5 FTEs
The scientific cloud vision

**HPC & Grid Clusters**
- Low latency access

**Cloud storage**
- Standard back-end for clusters and portals

**Fast data ingest**
- Standard devices at high data rates

**DropBox-like storage**
- Devices synchronise with storage

**Bulk WAN transfer**
- Moving huge datasets

**Remote access**
- Rich access via web-browser
The DESY Cloud service

• Status: production(-ish), but only for a few groups: 219 users, $2 \times 10^6$ files, 2.4 TiB

• Required minor patches to ownCloud & dCache:
  Changes always pushed into regular dCache releases; ownCloud 8 has all changes.

• Have a blueprint for any site to reproduce.
Demo:

http://desycloud.desy.de/...

WebDAV

NFS 4.1

WebDAV

~/ownCloud/

Worker Node

~/ownCloud/
Demo: sync-n-share
Demo: processed image, from WN
Integration within DESY infrastructure
Development and future work

- Allow **direct access** to ownCloud files:
  - Supporting direct access from worker-nodes, 3\textsuperscript{rd} party transfers, …
  - Files in dCache need to be owned by the **user** (i.e., not user owncloud)
  - Couldn’t fix ownCloud: work-around within dCache

- **Consistency** between ACLs and shares:
  - dCache ACLs to honour ownCloud shares and vice versa

- **Integrity**: e.g., propagate and handling checksums,

- **Notification**: avoid client polling,

- **Redirection** support for sync-client:
  - ownCloud server proxying data is bottleneck; want syncing to be more efficient by taking data from where its stored.
NFS v4.1/PNFS vs ownCloud (currently)
ownCloud: currently vs with redirect

Diagram showing the flow of data and metadata between ownCloud and dCache, with and without redirect.
Thanks for listening … any questions?
Backup slides
Not just ownCloud ...

- dCache team hosted a **two-day workshop** with project- and technical-lead of DCORE
  - Provides cloud storage with features beyond ownCloud
  - Some “big name” customers
- Initial “lite integration” by **December 2014**
  (includes redirection support)
- Then providing “tight integration” with shared namespace
Experience: problems with ownCloud

- If underlying FS disappears, all sync-clients delete all data.
- If underlying FS returns EIO on read, sync-client creates 0-length file: impossible to recover.
- Bulk delete through web interface is unreliable (under investigation).
- Rename directory causes client to delete all files and upload them again.
- Admin interface awkward with O(5k) users.
Thinking about sync-and-share

• Like other systems, small fraction of data is “hot”
  SSDs provide better performance, but can't afford only SSDs; nice to have system that places hot data on SSDs, cold data on HDD.

• Amazon had a smart idea: allow people to choose how much to pay
  Let users choose between Normal and Glacial QoS; e.g., disable sync for Glacial-like storage but allow access via web interface
WLCG dCache instances (only WLCG sites shown)
Over 10 years “Big Data” experience

<table>
<thead>
<tr>
<th>Era</th>
<th>Disk cache</th>
<th>Grid Storage</th>
<th>Generic Storage</th>
<th>Cloud Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. Trusted host</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>X.509, Kerberos</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Username+PW</td>
<td>SAML, OpenID,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OAuth, Token,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. Industry</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. Cloud Storage</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Additional Communities

- hermes
- ZEUS
- CDF
- CMS
- ALICE
- ATLAS
- IceCube
- Fermilab
- XFEL
- Belle II
- LOFAR
- SNIC
- CDF
- PETRA III

Additional Authentication

- Trusted host
- X.509
- Kerberos
- Username+PW
- SAML, OpenID, OAuth, Token, ...
The plan for dCache

• We will be adding support for CDMI.
  • Initial support is for the filesystem
    (it's a bit like WebDAV)
  • Then adding object-store and metadata support.
• Plan on providing CDMI this summer.