Plugins and Filesystem Integration

Alejandro Álvarez Ayllón
on behalf of the LCGM development team
Overview

▶ Existing plug-ins
  ▶ Native, MySQL, Oracle, Memcache
  ▶ HDFS, S3
  ▶ VFS
▶ Filesystem integration
Native DPM

- Interacts directly with the DPNS/DPM/LFC daemons
  - Simply redirects calls using the existing API
- Full backward compatibility
  - Both for namespace and pool/filesystem management
Plugins: MySQL

- Refactoring of the MySQL backend
  - Properly using bind variables and connection pooling
  - Huge performance improvements
- Namespace traversal comes from Built-in Catalog
Plugins: MySQL
Plugins: Oracle

- Refactoring of the Oracle backend
- What applies to the MySQL one, applies here
  - Better performance with bind variables and pooling
  - Namespace traversal comes from Built-in Catalog
Plugins: Memcache

- Memory cache for namespace requests
  - Reduced load on the database
  - Much improved response times
  - Horizontal scalability

- Can be put over any other Catalog implementation

```
UserGroupDb
Catalog
  INode
  PoolManager
  PoolDriver
  PoolHandler
  IODriver
  IOHandler
```

User domain
Namespace domain
Pool domain
I/O domain
Plugins: Memcache
Plugins: HDFS

- First new pool type
- HDFS pool can coexist with legacy pools, ...
  - In the same namespace, transparent to frontends
- All HDFS goodies for free (auto data replication, ...)
- Catalog interface coming soon
Plugins: S3

- Second new pool type
- Again, can coexist with legacy pools, HDFS, ...
  - In the same namespace, transparent to frontends
- Main goal is to provide additional, temporary storage
  - High load periods, user analysis before big conferences, ...
  - Evaluated against Amazon, now looking at Huawei and OpenStack
Plugins: VFS

- Third new pool type (currently in development)
- Exposes any mountable filesystem
  - As an additional pool in an existing namespace
  - Or directly exposing that namespace
- Think Lustre, GPFS, ...
Plugins: And more...

- **Librarian**
  - Replica failover and retrial
  - Used by the HTTP/DAV frontend for a *Global Access Service*
- **Profiler**
  - Boosted logging capabilities
  - For every single call, logs response times per plugin
- **HTTP based federations**
- **Writing plugins is very easy...**
- **ATLAS Rucio: first “external” plugin**
Filesystem integration

<table>
<thead>
<tr>
<th>Mode</th>
<th>UID</th>
<th>GID</th>
<th>Size</th>
<th>Modified</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>rwxrwxr-x</td>
<td>7</td>
<td>107</td>
<td>20</td>
<td>Wed, 14 Nov 2012 14:54:32 GMT</td>
<td>copy.txt</td>
</tr>
<tr>
<td>rwxrwxr-x</td>
<td>7</td>
<td>103</td>
<td>20</td>
<td>Tue, 16 Oct 2012 14:56:58 GMT</td>
<td>filesystem.txt</td>
</tr>
<tr>
<td>rwxrwxr-x</td>
<td>7</td>
<td>103</td>
<td>23</td>
<td>Tue, 16 Oct 2012 15:14:50 GMT</td>
<td>hadoop.txt</td>
</tr>
<tr>
<td>rwxrwxr-x</td>
<td>7</td>
<td>103</td>
<td>15</td>
<td>Tue, 16 Oct 2012 15:57:25 GMT</td>
<td>s3.txt</td>
</tr>
</tbody>
</table>

Request by /DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=aalvarez/CN=Alejandro Alvarez Ayllon (No proxy)
Powered by LCGDM-DAV 0.12.0