SNIA CDMI

CDMI - The Cloud Data Management Interface

A protocol for self-provisioning, administering and accessing cloud storage:

- RESTful HTTP
- Concept of containers and objects
- Users and groups management
- Metadata and arbitrary queries
- Retention and holds for compliance purposes
- Logging, auditing, billing
- Moving data between cloud systems
- Exporting via iSCSI or NFS
REST / RESTful

- **Representation State Transfer**
  - Started with Dissertation by Roy Fielding outlining the principles
  - A form of web services (but not based on WS-*)
- **Addressability**
  - Every object (resource) is addressable through a unique identifier
- **Uniform, Constrained Interface**
  - Use only HTTP verbs and model other semantics in the data model
  - Allows for Familiarity (low learning curve), Interoperability and Scalability
- **Representation Oriented**
  - Complexity is in the representations
- **Communicate Statelessly**
  - No persistent client-server connections

Why not just adopt one of the existing API?

- Despite the “open” licensing of several existing cloud storage interfaces, they all remain under the change control of a single vendor
- No cloud vendor wants to have a competitor have change control over their interface
  - Thus they release their own interface which they do have change control over
- This leads to the propagation of multiple interfaces, each essentially locking developers/customers into that service
- CDMI is under change control of a standards body, accommodates requirements from multiple vendors and can be extended for proprietary functions
Model for the interface

The resources which are accessed through the RESTful interface:

- **Root**: https://offering/>
  - **Capabilities - cdmi-capability**: https://offering/cdmi_capabilities
  - Key Value
  - Key Value

- **Container A - cdmi-container**: https://offering/containerA
  - Key Value
  - Key Value

- **Container B - cdmi-container**: https://offering/containerB
  - Key Value
  - Key Value

- **DomainA - cdmi-domain**: https://offering/cdmi_domains
  - Key Value
  - Key Value

- **DataObject1 - cdmi-object**: https://offering/containerA/dataobject1
  - Key Value
  - Key Value

- **DataObject2 - cdmi-object**: https://offering/containerA/dataobject2
  - Key Value
  - Key Value

- **Queue - cdmi-queue**: https://offering/containerA/queue1
  - Key Value
  - Key Value
### CDMI is maturing as a standard

<table>
<thead>
<tr>
<th>Maturity Level</th>
<th>Description</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. No Standards</td>
<td>Standardization needed</td>
<td>Recomend standards development</td>
</tr>
<tr>
<td>2. Under Development</td>
<td>Discussions within standards groups; Open source project launched</td>
<td>Monitor and provide feedback to standards development</td>
</tr>
<tr>
<td>3. Specification Document Published</td>
<td>Initial specification posted for public review</td>
<td>Review specification and plan testing</td>
</tr>
<tr>
<td>4. Initial Reference Implementation</td>
<td>Reference implementation available</td>
<td>Evaluate reference implementation</td>
</tr>
<tr>
<td>5. Early Third Party Testing</td>
<td>Evaluation in test environments</td>
<td>Pilot Projects should consider use</td>
</tr>
<tr>
<td>6. Initial Production Implementations</td>
<td>Successful use in production</td>
<td>Mainstream projects should consider use</td>
</tr>
<tr>
<td>7. Many Deployments</td>
<td>Widespread use by many groups</td>
<td>Projects should use the standard as a default</td>
</tr>
<tr>
<td>8. Accepted Standard</td>
<td>De facto or de jure acceptance as a standard</td>
<td>Projects should use unless special circumstances require exemption</td>
</tr>
<tr>
<td>9. Aging Standards</td>
<td>Newer standards are under development</td>
<td>Projects should explore alternatives</td>
</tr>
</tbody>
</table>

*Source: Draft NIST Cloud Standards Roadmap*

### Learn more at

- http://groups.google.com/group/snia-cloud
- http://twitter.com/SNIACloud (@SNIACloud)
- http://www.google.com/profiles/SNIACloud

Learn more at http://snia.org/cloud
Panel discussion
http://www.scality.com

Thank You
http://www.scality.com
A multi-tier architecture with Replication and Erasure code, and a NoSQL database to deliver POSIX compliance.