CDMI service for Cloud Frameworks

Ilja Livenson

livenson@kth.se

Motivation

- To create a stable CDMI-compliant service tuned for EGI operations model and infrastructure, incl. AAI.

- … so that EGI Fedcloud TF could create a solution based completely on open standards.

- Targeting
  - Site operators
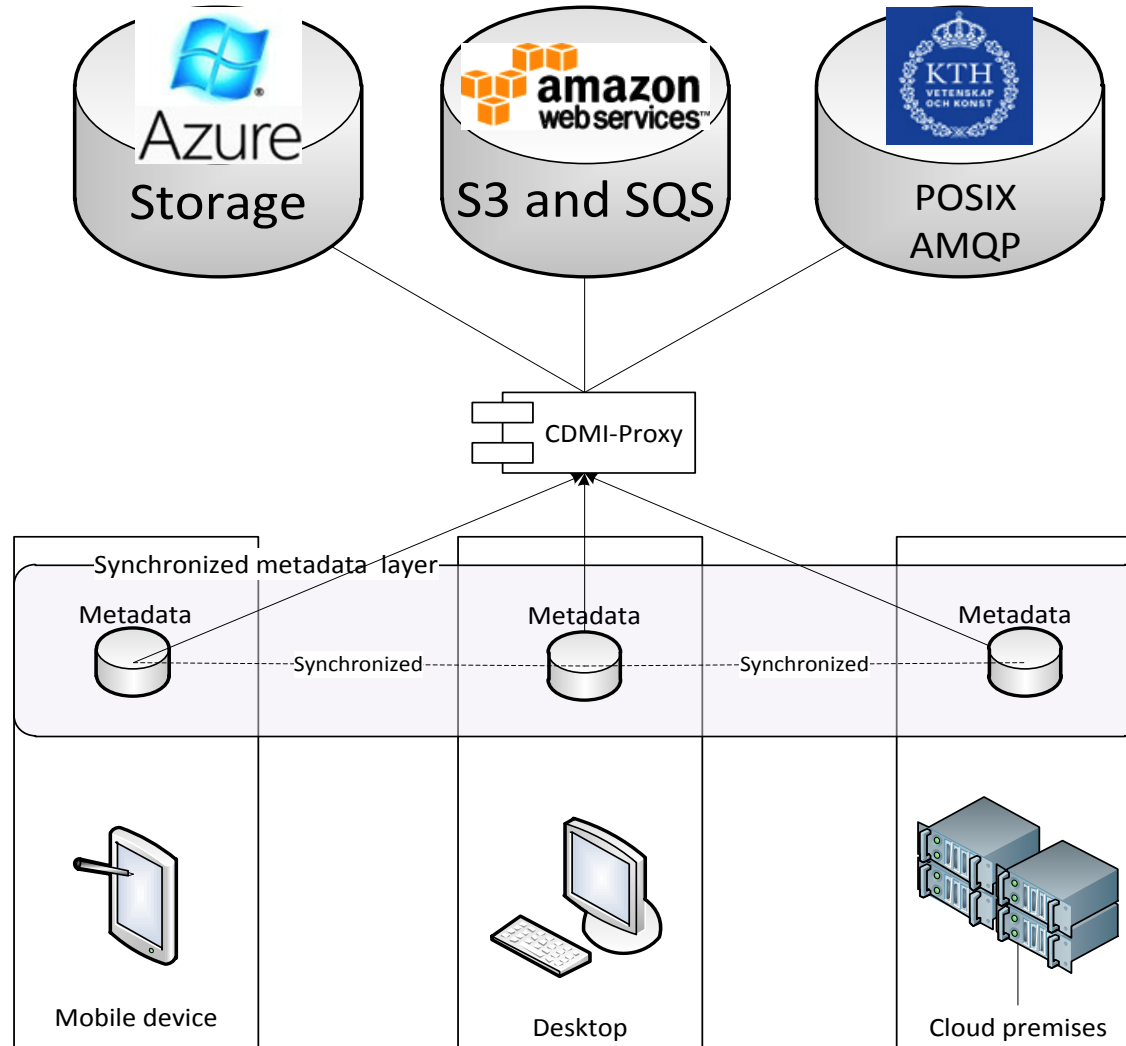 - SW developers, which could extended supported storage access methods with CDMI-compliant endpoints

Background: CDMI

- CDMI = Cloud Data Management Interface
- SNIA CDMI is an HTTP-based standard for operating on storage objects in a RESTful way
  - Similar to WebDAV, AWS S3, MS Azure Blob but backward compatible with HTTP
  - Support for rich metadata semantics
- Growing number of vendors (both open-source and commercial, e.g. NetApp)

- Latest version: 1.0.2 (still young)

Background: VENUS-C/CDMIProxy

- FP7 VENUS-C – EU project, which aimed at creating an PaaS layer on top of public and private clouds for scientific applications

- CDMIProxy prototype was developed as part of the project
  - Concentrated on object storage only
  - Only basic metadata handling
  - Adapters for public clouds

CDMI Proxy 'architecture'

Idea of the mini-project

- Base on a prototype of CDMI-compliant storage server developed within the VENUS-C project

- Refactor with accent on stability and EGI Federated TF's use cases

- Extend supported use cases to include management of block storage devices

Planned functionality

- CRUD for Objects
  - Refactoring for stability
- Block storage
  - Orchestration
- Richer metadata support
  - VM Image specific

- *User facing*
  - Lightweight CDMI-compliant javascript browser
- *Developer facing*
  - SDKs in python and java

Integration

- EGI
  - Federated cloud TF requirements
    - Integration with cloud frameworks in use
  - AAI
  - Accounting

- EUDAT
  - Export to/from Simplestore

Project outline

- Consortium
  - KTH as the main developer (EGI contribution of 60k over 12 months)
  - EGI Fedcloud as evaluator
  - (negotiation in progress about a sponsored developer from one open-source project)
- Development timeline
  - Planning started on the 1.04
  - The first stable release is planned at the end of Summer
  - Bi-weekly releases from there on
  - Last month is reserved for bug fixing only with no additional functionality being added

KPIs

- Number of downloads
- Number of running installations
- Number of support tickets

Thank you very much!

livenson@kth.se