Leveraging Cloud Storage Through Mobile Applications Using Mezeo Cloud Storage Platform REST API

John Eastman
Mezeo
Cloud Storage

- On-demand, API-based access to storage
- Storage accessed through REST Web Services APIs via a network
- Persistent versus dedicated to a server

Reservationless:
- Pay for use model (Business Decision)
- Able to massively scale in terms of size and access in near real time
Mobile Access to Storage

- Growing desire to interact with data while away from traditional access points
- Most users now carrying a computer (AKA a phone)
- Cloud storage mobile development trends
  - Integration with existing applications
  - As a means for additional access to user content
  - New applications – solving problems that were previously unsolvable (data and compute platform were unavailable)
Cloud Storage APIs

- Amazon S3, Eucalyptus, Google
- CDMI
- OpenStack Object Storage
- Mezeo Cloud Storage API
Mezeo

- Software company that provides service providers and enterprises with a platform for Cloud Storage services
  - Mezeo is **not** a service provider

- Mezeo platform
  - Designed to enable service providers to enable branded, differentiated Cloud Storage services
  - Enables enterprises to stand up a secure private storage cloud
  - Exposes multiple Cloud Storage APIs for developers
    - All APIs access the same storage

- Growing developer community
  - Partner ecosystem – *Mezeo Ready*
  - Larger community of independent developers
Why Support Multiple APIs?

- A lot of applications already exist
  - Little to no application changes to use a different service provider or move application in-house
  - Nice to have an application that works with all providers

- The same reason there is not only one programming language out there
  - Each has its own set of strengths and weaknesses

- Pick the right tool for the job
The Mezeo Cloud Storage API

- Focus on providing storage access with application-level services
- Content accessible via unique identifier (URL) and namespace (path)
- Leverages SSL and HTTP Authentication
  - Wide support
  - No sessions – each request is authenticated
- REST API closely tied to HTTP
  - Minimal overhead
  - Easy to use on any device / programming language
The Mezeo Ready API (cont.)

- Implemented based on REST architectural style over HTTP
- Everything is a resource
  - Resources support HTTP methods (GET, POST, PUT...)
  - X-Method-Override header support for libraries that don’t support all methods
- API publishes **one** well-known URI (**https://<api server>/v2**)
- Other URIs are discovered from content returned in representations
  - URIs (nested in representations) for related resources
- Resource representations available as JSON and XML
Web Example

- Using only Javascript, we can create a complete storage access application on top of Mezeo API
- Directly uses JSON resource representations
SDKs

- API documents HTTP RESTful interface for maximum interoperability
- Low-level network/HTTP programming, especially on mobile devices, can be complex and tedious
- Optional SDK client libraries can accelerate development
  - Reduce client complexity
  - Abstract most details of HTTP protocol
SDKs (cont.)

- Support the same RESTful architectural style as API
- Everything is a resource
  - Service
    - Exposes methods (GET, PUT, DELETE, etc.)
  - Resources
    - Containers
    - Files
    - Tags
    - ...

- Cloud Resource represents well-known URI
- Python SDK available from Python Package Index (http://pypi.python.org/pypi/mrcsp)

Example (retrieve list of all Tags):

```python
service = Service("api.example.com", "user@example.com", "my_password", APIKEY, "MyPythonExample/1.0")

cloud = Cloud(service.get_serveruri())
try:
    cloud = service.get_resource(cloud)
    tags = service.get_resource(cloud.get_tags())
except RestException, ex:
    print("Failed to retrieve resource: %s" % repr(ex))
```
CloudShell Example

- ‘FTP-like’ shell access to storage account
- Implements additional operations like file tagging
- Complete code available (online) as an example

Example session:

$> python cloudshell –a user@example.com -s api.example.com

CloudShell - Cloud Storage Platform Shell

Type 'help' for list of commands

$(cloud /)> tag sdc.png demo
$(cloud /)> tags
List of all tags:
demo
$(cloud /)>
Objective-C SDK

- Provides abstraction of each resource
- Support for both synchronous and asynchronous requests

Example code from view controller:

```objective-c
AuthCredentials* credential = [[[BasicAuthCredentials alloc] initWithCredentials: @"user@example.com" password: @"mypassword"] autorelease];
Service* service = [[Service alloc] initWithCredentials: credential]
TagList* tagList = (TagList*)[service createNewResourceOfType: TAG_LIST_RES];
tagList.uri = [Cloud sharedCloudInstance].tagsLink;
[service get_resourceAsynchronously: tagList delegate: self startImmediately: YES];
```
An iPhone Application

- Simple tag browsing on the phone
- Code walk-through available online
Summary

- Mobile is a new growth area for storage applications
- Mezeo API provides another tool for storage application development
- Mezeo SDKs
- Additional information is available at http://developer.mezeo.com
Mezeo Ready Developer Site: [http://developer.mezeo.com](http://developer.mezeo.com)

- **SDKs:**
  - Python - [http://pypi.python.org/pypi/mrcsp](http://pypi.python.org/pypi/mrcsp)

  - Python CloudShell
  - iPhone Tags client