Federated sharing with the Open Cloud Mesh API

Dimitri van Hees (@dvh)
Joost Farla (@joostfarla)

Workshop on Cloud Services for Synchronisation and Sharing - February 1st 2017 - Amsterdam
Company

- Founded in 2015 by Dimitri van Hees & Joost Farla
- API Strategy & Design
- API Management
- API Development & Ops
Assignment

Design a future-proof and vendor neutral API specification to facilitate federated sharing of files and directories across different sharing platforms.
Approach

• *Design first* principle
• Agile
• Open
• Community driven
**API Design**

- Open API Specification (fka *Swagger*)
- Vendor neutral
- Open source
- Adopted by The Linux Foundation
- Machine readable (YAML/JSON)
- Large community
- Backed by i.a. Google, Microsoft, IBM, Adobe and Atlassian
swagger: "2.0"
info:
  title: Open Cloud Mesh API
  description: Open Cloud Mesh Open API Specification.
  version: 0.0.3
schemes:
  - https
consumes:
  - application/json
produces:
  - application/hal+json
parameters:
  id:
    name: id
    in: path
    description: Unique ID to identify the share at the consumer side.
    required: true
    type: string
paths:
  /shares/{id}:
    get:
      summary: Retrieve a single share
      description: With this call the provider can retrieve a single share.
      parameters:
        - $ref: "#/parameters/id"
      responses:
        200:
          description: Representation of the single share.
          schema:
            $ref: "#/definitions/Share"
Open Cloud Mesh API (v0.0.3)

Download OpenAPI (fka Swagger) specification: [Download]

Open Cloud Mesh Open API Specification.

Retrive a single share

With this call the provider can retrieve a single share it created at the consumer side. This is a convenience call which could be used to check whether everything is still in sync etc.

PARAMETERS

Path Parameters

- id string Required
  Unique ID to identify the share at the consumer side.

Responses

~ 200 Representation of the single share.

RESPONSE SCHEMA

- shareWith string Required
  Consumer specific identifier of the user or group the provider wants to share the resource with. This is known in advance. Please note that the consumer service endpoint is known in advance as well, so this is not part of the request body.

- name string Required
  Name of the resource (file or folder).

- description string
  Optional description of the resource (file or folder).

- providerId string Required
  Identifier to identify the resource at the provider side. This is unique per provider.

- owner string Required
  Provider specific identifier of the user that wants to share the resource. Please note that the requesting provider is being identified on a higher level, so the former remote property is no part of the request body.

- protocol protocol Required
  The protocol which is used to establish synchronisation. At the moment only webdav is supported, but other (custom) protocols might be added in the future.

- id string Required
  Unique ID to identify the share at the consumer side.
Scope and assumptions

- Authentication between services is already established
- Provider knows the consumer endpoint and user in advance
- Consumer doesn't have to accept a share
- Dealing with incoming/outgoing shares is vendor specific
- Actual file sync/traffic is not in scope of this specification
Endpoints

- **POST /shares** Share a resource
- **GET /shares** Retrieve all shares
- **GET /shares/{id}** Retrieve a single share
- **POST /notifications** Send a notification
- **GET /notifications** Retrieve all notifications
- **GET /notifications/{id}** Retrieve a single notification
Share a resource

POST /shares

201: Created share
Send a notification

POST /notifications

201: Created notification
Request message format

- **JSON** *(application/json)*
- Machine readable
- Commodity for most modern languages
- Validation with JSON Schema
- Community standard
Response message format

- HAL+JSON (application/hal+json)
- Same benefits as plain JSON
- Added *hypermedia controls*
- Possibility to maintain vendor specific URLs
{ "_embedded": { "notifications": [{ "type": "SHARE_REMOVED", "message": "I don't want to use this share anymore.", "id": 9303, "createdAt": "2016-12-05T15:06:58Z", "_links": { "self": { "href": "/notifications/9303" } } } ], "_links": { "self": { "href": "/notifications" }, "next": { "href": "/notifications?page=2" } } } }

nextPage = "/notifications?page=" + page
{  
  "_embedded": {  
    "notifications": [{  
      "type": "SHARE_REMOVED",  
      "message": "I don't want to use this share anymore.",  
      "id": 9303,  
      "createdAt": "2016-12-05T15:06:58Z",  
      "_links": {  
        "_self": {  
          "href": "/notifications/9303"  
        }  
      }  
    }]  
  },  
  "_links": {  
    "self": {  
      "href": "/notifications"  
    },  
    "next": {  
      "href": "/notifications?page=2"  
    }  
  }  
}  

nextPage = "/notifications?page=" + page  
nextPage = response._links.next.href
**Sync protocol independent**

- Protocol specific properties wrapped in share object
- Probably only webdav supported at the moment
- Possibility to extend the spec (future-proof)
- Possibility to implement own protocol (flexible)
{  
"shareWith": "peter.szegedi@geant.org",
"name": "spec.yaml",
"description": "This is the Open API Specification file.",
"providerId": "7c08226-d9a1-11e6-bf26-cec0c932ce01",
"owner": "dimitri@apiwise.nl",
"protocol": {
  "name": "webdav",
  "options": {
    "username": "dimitri",
    "permissions": 31
  }
}
}
Thank you!

- https://github.com/GEANT/OCM-API
- https://rawgit.com/GEANT/OCM-API/v1/docs.html
- https://github.com/OAI/OpenAPI-Specification
- https://github.com/mikekelly/hal_specification
- https://www.apiwise.nl
- https://twitter.com/dvh
- https://twitter.com/joostfarla