

MyDataShare Developer Guide

This document describes the functionality currently available to ecosystem members.

This document is best approached with knowledge on [MyData](#) as a concept, [MyDataShare](#) as a solution and [REST](#) interfaces as well as [OpenID Connect](#) as technologies.

This document is a gentle introduction to utilizing MyDataShare functionalities.

It is complemented by deeper dives into the actual application programming interface as well as integration to the MyDataShare identity provider.

Disclaimer

MyDataShare is a work in progress.

Thus implementation details and processes are subject to change.

Do not hesitate to contact us in case you encounter bugs, inconsistencies or other hindrances.

Definitions and assumptions

This document utilizes the beta environment of MyDataShare.

This document utilizes Signicat as the identity provider.

This document utilizes curl as the command line tool to access the API. You can substitute any other tool that enables use of direct REST interactions (postman, insomnia).

The following definitions are used in the document for brevity:

- `AUTH_URL` = `https://auth.beta.mydatashare.com/`
- `ORG_URL` = `https://api.beta.mydatashare.com/organization/v2.0/`
- `O_TOKEN` = access token as received from Gluu.

Some included responses are abbreviated for brevity, as shown by ellipsis '...'.

MyDataShare API overview

The MyDataShare API is a RESTful interface.

Each entity is identified with an UUID, and this identifier is used in retrieving entities as well as describing relations between entities.

Each entity is returned as a whole in the response, and the response also contains the entities that are referred to in the returned.

This process is applied recursively, so the response may contain multiple entities.

Data model in a nutshell

The following entities are the high-level resources used in MyDataShare:

- Organization
 - company (or equivalent) entity
 - able to create data providers and data consumers.
- Data consumer
 - an organization-owned entity

- able to create processing records (= consent requests).
- Data provider
 - an organization-owned entity
 - able to serve data as a source for data consumers.
- Data subject
 - a person
 - able to receive, view and respond to processing records.
 - Note: there is no single resource for a data subject. A data subject is uniquely identified with any of the identifiers (ssn, phone number, etc) in a keychain.
- Processing record
 - a permission request created by a data consumer to a data subject on specified data provider's contents.
 - Note: a data provider can be omitted in some cases (e.g. permission to store contact details)

Preliminary steps

Obtain client credentials

The client credentials are created by a MyDataShare-employee, please provide the following information in your request:

- URLs where browser can be redirected after successful authentication
- URLs where browser can be redirected after successful logout
- Display name of the application
- Contact email addresses and phone numbers
- Organization name in Finnish and English.
- Whether both data consumer and a data provider are needed, or just one of them.
- Token endpoint authentication method (`client_secret_basic` or `client_secret_post`)

As a response you will receive multiple sets of

- Client identifier (= ORG-CLIENTID below)
- Client secret (= ORG-PASSWORD below)

Request creation of an Organization

The organizations are created by a MyDataShare-employee, please provide the following information in your request:

- Organization name in Finnish and English.
- Organization registration number (i.e. Y-Tunnus for finnish companies).
- Organization country.
- Client identifier(s) to be attached to the organization (as created in the preceding step)

As a response you will receive:

- UUID of the newly created organization (= ORG-UUID below).
- UUID of the authorization endpoint URL (= AUTH-ENDPOINT-UUID).

Use Signicat test users

As the beta environment should never contain real user identities, we utilize the handful of test users as provided by the identity brokering finnish trust network members. Please refer to [Signicat documentation](#) and select your test subjects.

Note As there is only a limited number of the test users, you are likely to see unexpected processing records in the Wallets belonging to them - these records have been created by other users of the environment.s

Familiarize yourself with MyDataShare API

Obtain an access token

Perform the following request to the MyDataShare identity provider to obtain an access token with the client credentials flow:

```
curl -X POST -d "grant_type=client_credentials&client_id=ORGCLIENTID&
client_secret=ORGPASSWORD&scope=organization" $AUTH_URL/oxauth/restv1/token
```

Extract the token from the response, and store it in O_TOKEN (organization-scoped token) variable for convenience.

Validate that you are able to access the API and your organization

Perform the following request to the MyDataShare interface to see the details of your organization:

```
curl -X GET -H "authorization: Bearer $O_TOKEN" -H "Content-Type:application/json"
$ORG_URL/organization/ORG-UUID
```

The response describes the organization itself and its identifiers as well as its client identifiers:

```

{
  "organizations": {
    "726720a1-99d7-41f0-9b5f-4c0a085f50f6": {
      "default_language": "fin",
      "url_group_id": 11,
      "country": "FIN",
      "description": "The greatest of the great old ones, Cthulhu, sleeps in the ...",
      "legal_entity_type": "oy",
      "translation_id": 25,
      "name": "Oy Cthulhu Ab",
      "suppressed_fields": [
        "data_consumers.uuid",
        "data_providers.uuid"
      ],
      "updated": "2020-06-24T10:34:09.982462+00:00",
      "organization_client_ids.uuid": [
        "31582023-5619-4dc2-a469-e1e9466fdd79"
      ],
      "uuid": "726720a1-99d7-41f0-9b5f-4c0a085f50f6",
      "deleted": false,
      "organization_ids.uuid": [
        "c5a0ae3f-25f7-4e1a-b693-34493c37beb0"
      ],
      "created": "2020-06-24T10:34:09.982457+00:00",
      "group_id": null
    }
  },
  "organization_ids": {
    "c5a0ae3f-25f7-4e1a-b693-34493c37beb0": {
      "org_id_value": "3385750-2",
      "uuid": "c5a0ae3f-25f7-4e1a-b693-34493c37beb0",
      "suppressed_fields": [
        "organizations.uuid"
      ],
      "deleted": false,
      "org_id_type": "registration_number",
      "created": "2020-06-24T10:33:05.209426+00:00",
      "updated": "2020-06-24T10:33:05.209431+00:00",
      "org_id_country": "FIN"
    }
  },
  "organization_client_ids": {
    "31582023-5619-4dc2-a469-e1e9466fdd79": {
      "name": "Client of Cthulhu",
      "client_id": "@!FF9F.35A9.9F4B.DF31!0001!6A12.9F11!0008!71EB.420C.FA9F.75A6",
      "created": "2020-06-24T10:37:15.417749+00:00",
      "updated": "2020-06-24T10:37:15.417754+00:00",
      "uuid": "31582023-5619-4dc2-a469-e1e9466fdd79",
      "deleted": false,
      "organization_uuid": "726720a1-99d7-41f0-9b5f-4c0a085f50f6"
    }
  }
}

```

The identifiers correspond to real-world identifiers (such as the finnish y-code business registration number), and the client

identifiers are used to authorize access to the interface (as described in the preceding step).

Each list of entities identifies them by their unique identifiers (UUIDs), which are used to refer to them and to link them to each other.

```
"organizations": {  
  "726720a1-99d7-41f0-9b5f-4c0a085f50f6": {  
    "default_language": "fin",  
    ...  
  }  
}
```

Create data provider

Perform the following request to the MyDataShare interface to create a data provider:

```
curl -X POST -H "authorization: Bearer $0_TOKEN" -H "Content-Type:application/json" -d '{  
  "organization_uuid":ORG-UUID, "default_language":"fin", "name":"Nimi", "description":"Kuvaus",  
  "has_live_preview":false, "static_preview":"Esikatselu" }' $ORG_URL/data_provider
```

Parameter	Mandatory	Description	Example value
organization_uuid	Yes	Identifier of the organization that owns the data provider	3fa85f64-5717-4562-b3fc-2c963f66afa6
name	Yes	Name of the data provider	Delivery address
description	Yes	Description of the data provider	The information provided includes the following: street address, city, zip code, state and country
has_live_preview	Yes	Whether data subject specific data shall be displayed in the Wallet (not supported currently)	false
static_preview	Yes	A preview that is displayed in every data subject's Wallet	Malone Avenue 32, Stockton, 85010, UT, United States
default_language	Yes	The language in which the contents are presented by default	fin

The response describes the created data provider:

```

"data_providers": {
  "a0aee0bf-27b9-42b1-a90f-bfaa2e4a27dd": {
    "url_group_id": 1474,
    "translation_id": 228,
    "suppressed_fields": [
      "identifiers.uuid",
      "processing_records.uuid"
    ],
    "default_language": "fin",
    "has_live_preview": false,
    "name": "Nimi",
    "static_preview": "Esikatselu",
    "created": "2020-06-18T09:08:50.462913+00:00",
    "updated": "2020-06-18T09:08:50.462918+00:00",
    "major_version": 1,
    "organization_uuid": "726720a1-99d7-41f0-9b5f-4c0a085f50f6",
    "minor_version": 0,
    "uuid": "a0aee0bf-27b9-42b1-a90f-bfaa2e4a27dd",
    "description": "Kuvaus",
    "deleted": false
  }
}
...

```

Take note of the identifier of the data provider (= DP-UUID). In this example: `a0aee0bf-27b9-42b1-a90f-bfaa2e4a27dd`.

Create data consumer

Perform the following request to the MyDataShare interface to create a data consumer:

```

curl -X POST -H "authorization: Bearer $O_TOKEN" -H "Content-Type:application/json" -d '{
  "organization_uuid":ORG-UUID, "default_language":"fin", "name":"Nimi", "purpose":"Tarkoitus",
  "description":"Kuvaus", "legal":"Laki.", "pre_cancellation":"Pre", "post_cancellation":"Post" }'
$ORG_URL/data_consumer

```

Parameter	Mandatory	Description	Example value
organization_uuid	Yes	Identifier of the organization that owns the data consumer	3fa85f64-5717-4562-b3fc-2c963f66afa6
name	Yes	Name of the data consumer	Delivery details
description	Yes	Description of the data consumer	In order to be able to ensure a smooth delivery, the recipient is requested to provide all pertinent details. These details include ...
purpose	Yes	Purpose of the data consumer	Recipient's delivery address
legal	Yes	Legal flavoured purpose of the data consumer	The delivery details provided will be forwarded to all agents involved in the delivery process. The list of commonly utilized agents is available at ...
pre_cancellation	Yes (can be empty)	Text displayed to data consumer at permission declination/revocation	If you revoke access to your delivery details, the delivery of the goods may be significantly delayed
post_cancellation	Yes (can be empty)	Text displayed to data consumer after permission declination/revocation	You have now revoked access to your delivery details. Any delays experienced during the delivery process have been explicitly disclaimed by us
default_language	Yes	The language in which the contents are presented by default	fin

The response describes the created data consumer:

```

{
  "data_consumers": {
    "95322354-b116-47e3-b210-d3d672f8139f": {
      "url_group_id": 1472,
      "translation_id": 226,
      "legal": "Laki.",
      "post_cancellation": "Post",
      "default_language": "fin",
      "name": "Nimi",
      "suppressed_fields": [
        "processing_records.uuid"
      ],
      "created": "2020-06-18T09:03:56.760426+00:00",
      "updated": "2020-06-18T09:03:56.760431+00:00",
      "major_version": 1,
      "purpose": "Tarkoitus",
      "organization_uuid": "726720a1-99d7-41f0-9b5f-4c0a085f50f6",
      "minor_version": 0,
      "uuid": "95322354-b116-47e3-b210-d3d672f8139f",
      "pre_cancellation": "Pre",
      "description": "Kuvaus",
      "deleted": false
    }
  }
  ...
}

```

Take note of the identifier of the data provider (= DC-UUID).

Create processing record

Perform the following request to the MyDataShare interface to create a processing record that defines a consent request for a data subject to access content at a data provider:

```

curl -X POST -H "authorization: Bearer $0_TOKEN" -H "Content-Type:application/json" -d '{
  "data_consumer_uuid":"DC-UUID", "data_provider_uuid":"DP-UUID", "identifier": { "country":"FIN",
  "type":"ssn", "id":"TESTUSER-SSN" }, "record_type":"consent" }' $ORG_URL/processing_record

```

Parameter	Mandatory	Description	Example value
data_provider_uuid	Yes	Identifier of the data consumer	3fa85f64-5717-4562-b3fc-2c963f66afa6
data_consumer_uuid	Yes	Identifier of the data provider	3fa85f64-5717-4562-b3fc-2c963f66afa6
record_type	Yes	Legal basis for permission request	consent
identifier.type	Yes	Type of the data subject's identifier	ssn
identifier.country	Yes	Country of the data subject's identifier	fin
identifier.id	Yes	Value of the data subject's identifier	220188-145U

The response describes the created processing record (and all related entities):

```

"processing_records": {
  "7b5b7c9d-a9fb-41c2-9fd7-673f3a23b61b": {
    "data_provider_uuid": "a0aee0bf-27b9-42b1-a90f-bfaa2e4a27dd",
    "record_type": "consent",
    "legal": null,
    "status": "pending",
    "name": null,
    "supersedes_uuid": null,
    "expires": null,
    "accepted_language": null,
    "identifier_uuid": "7bf3dc27-d69b-4225-a100-0da9710cbc48",
    "not_valid_before": null,
    "url_group_id": 1475,
    "description": null,
    "created": "2020-06-18T09:10:20.268744+00:00",
    "updated": "2020-06-18T09:10:20.268749+00:00",
    "purpose": null,
    "data_consumer_uuid": "95322354-b116-47e3-b210-d3d672f8139f",
    "uuid": "7b5b7c9d-a9fb-41c2-9fd7-673f3a23b61b",
    "group_id": null,
    "deleted": false
  }
},

```

Take note of the identifier of the processing record (= PR-UUID).

The most interesting returned entity, in addition to the processing record itself, is the identifier and its relatives that describes the data subject, the person to whom this processing record is addressed to:

```

"identifiers": {
  "7bf3dc27-d69b-4225-a100-0da9710cbc48": {
    "url_group_id": 311,
    "verified": null,
    "suppressed_fields": [
      "data_providers.uuid",
      "processing_records.uuid"
    ],
    "id_type_uuid": "49679532-9c4c-4a46-9e5b-a914472f9612",
    "created": "2020-06-12T10:39:11.567107+00:00",
    "updated": "2020-06-12T10:39:11.567108+00:00",
    "id": "010200A9618",
    "id_provider_infos.uuid": [
      "9cf5cfe3-9412-4a31-b4f8-dabb191d260e",
      "d199be2f-d8f4-4a9f-a778-c7f69fe75d87"
    ],
    "uuid": "7bf3dc27-d69b-4225-a100-0da9710cbc48",
    "group_id": null,
    "deleted": false
  }
},
...
"id_providers": {
  "4e161a9e-f1f3-4e89-b30a-6854730bcda4": {
    "url_group_id": 1463,
    "translation_id": 222,

```

```

    "type": "gluu",
    "suppressed_fields": [
      "identifiers.uuid"
    ],
    "name": "Gluu",
    "created": "2020-06-12T11:07:07.850535+00:00",
    "updated": "2020-06-12T11:07:07.850540+00:00",
    "id": "https://gluu.beta.mydatashare.com",
    "uuid": "4e161a9e-f1f3-4e89-b30a-6854730bcda4",
    "description": "Gluu (beta)",
    "deleted": false
  },
  ...
  "id_provider_infos": {
    "9cf5cfe3-9412-4a31-b4f8-dabb191d260e": {
      "first_name": "Onni Juhani",
      "language": null,
      "last_name": "Korhonen",
      "created": "2020-06-18T08:56:21.056922+00:00",
      "updated": "2020-06-18T08:56:21.056929+00:00",
      "last_login": "2020-06-18T08:56:20.959805+00:00",
      "attributes": null,
      "source": "signicat",
      "uuid": "9cf5cfe3-9412-4a31-b4f8-dabb191d260e",
      "identifier_uuid": "7bf3dc27-d69b-4225-a100-0da9710cbc48",
      "id_provider_uuid": "4e161a9e-f1f3-4e89-b30a-6854730bcda4",
      "deleted": false
    },
    ...
  },
  "id_types": {
    "49679532-9c4c-4a46-9e5b-a914472f9612": {
      "url_group_id": 1,
      "translation_id": 1,
      "type": "ssn",
      "suppressed_fields": [
        "identifiers.uuid"
      ],
      "name": "Finnish personal identity number",
      "created": "2020-06-12T10:37:07.518073+00:00",
      "updated": "2020-06-12T10:37:07.518073+00:00",
      "country": "FIN",
      "uuid": "49679532-9c4c-4a46-9e5b-a914472f9612",
      "verify_interval": -1,
      "description": "Identifier for tracking individuals",
      "deleted": false
    },
    ...
  }
}

```

View the processing record in the MyDataShare Wallet

Log in to the MyDataShare Wallet at: <https://wallet.beta.mydatashare.com/> (use the same test user to whom you targeted with the processing record in the preceding step).

You can interact with the processing record in the Wallet (grant the request, decline the request, re-grant the request).

Obtain status of the processing record

Perform the following request to the MyDataShare interface to retrieve the processing record:

```
curl -X GET -H "authorization: Bearer $TOKEN" -H "Content-Type:application/json"  
$ORG_URL/processing_record/PR-UUID
```

The `status`-field describes the current status of the processing record.

```
...  
"status":"active"  
...
```

Create an application that uses MyDataShare

The interfaces as described in the preceding chapter are usable programmatically as well. The requests and responses are plain REST. Generating and interpreting them is significantly easier using a convenience library (obviously dependent on the programming language used).

If you implement your solution as a web application, it is strongly suggested to split the application into two parts, and confine the use of any client secret in the backend component only.

However, client applications (native mobile applications or Javascript based single-page applications) that cannot keep the client secret secure can use the Authorization Code Grant Flow with [PKCE extension](#).

Please refer to the accompanying MyDataShare OpenID Connect Integration Specification for details about the authentication flows of the application and integration in general.

Allow the user to interact with the MyDataShare Wallet

You can forward the user's browser to a single processing record (and return back to your application) using the following URL:

```
http://wallet.beta.mydatashare.com/wallet/requests?processing_record=PR-UUID&auth_endpoint_uuid=AUTH-  
ENDPOINT-UUID#pr
```

Where the AUTH-ENDPOINT-UUID is provided by MyDataShare upon registration.

API reference

The [MyDataShare API reference](#) provides full details on the programming interface.