Software Requirements Specification

for

Gaia-X Federation Service

Sovereign Data Exchange
Data Exchange Logging Service
SDE.DELS

Published by

eco – Association of the Internet Industry (eco – Verband der Internetwirtschaft e.V.)
Lichtstrasse 43h
50825 Cologne
Germany

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Document Purpose

The purpose of the document is to specify the structure, features, and requirements of the Gaia-X Sovereign Data Exchange Service subcomponent Data Exchange Logging Service as a foundation for a public tender for the implementation of this microservice. The audience of this document shall be familiar with the general concepts, vision, and ideas of Gaia-X and its Federation Services. This document does not contain explanations of Gaia-X itself or specifications of other Federation Services. The specifications of other Federation Services, which are necessary to implement the Data Exchange Logging Service, are referred to throughout this document. The main audience of this document is attendees of the public tender.

1.2 Product Scope

The Gaia-X Federation Service Data Exchange Logging Service (GX-DELS) provides evidence that data has been (a) submitted and (b) received and (c) rules and obligations (Data Usage Policies) were enforced or violated. This supports the clearing of operational issues, but also eventually the clearing of fraudulent transactions.

The Data Provider can track that, how, and what data was provided, and the consumer can be notified about this. The Data Consumer can track that this data was received or not received. Additionally, the Data Consumer can track and provide evidence on the enforcement of data usage policies or violation of data usage policies (Although the evidence is weak, as long as there are no sufficient measures for data usage policy enforcement in place). The log can be used as a basis for clearing and billing, but this is not the focus of WP3[1] for release 1. Business transactions for the GX-DELS and the GX-DCS (Gaia-X Data Contract Service) should be defined by the federator operating those services.

From a functional perspective, the GX-DELS provides an interface to track logging notifications and to read the logging messages afterward. The logging mechanism is specified in accordance with W3C linked data notifications.

The notification, therefore, includes minimal requirements, e.g., date, time, (a reference to) sender, Data Provider, Data Consumer, data exchange contract. The parties involved in the data exchange are typically sender and consumer of notifications. Some use cases may also require the consumption of the notifications by a 3rd eligible party.

The provisioning of notifications into the GS-DELS can be enforced by mechanisms implemented in the Data Consumers and Data Providers systems or by cryptographic mechanisms (forced logging).

[1] Please refer to appendix D for an overview and explanation of the Work Packages (WP).

1.3 Definitions, Acronyms, and Abbreviations

Term/Acronym	Meaning	References
GX	Gaia-X	Refer to www.gaia-x.eu
GX-CAM	Gaia-X Federation Service Compliance - Continuous Automated Monitoring	Refer to https://www.gxfs.de/federation-services/compliance/continuous-automated-monitoring/
GX-DCS	Gaia-X Federation Service Data Contract Service	Refer to https://www.gxfs.de/federation-services/sovereign-data-exchange/data-contract-service/
GX-DELS	Gaia-X Federation Service Data Exchange Logging Service	The service specified in this document.
GX-FC	Gaia-X Federation Service Federated Catalogue	Refer to https://www.gxfs.de/federation-services/federated-catalogue/core-catalogue-features/
GX-FS	Gaia-X Federation Services	Refer to www.gxfs.de
ldn	Linked Data Notification	https://www.w3.org/TR/ldn/
ldp	Linked Data Platform	https://www.w3.org/TR/ldp/
URI	Uniform Resource Identifier	RFC 4151: https://tools.ietf.org/html/rfc4151 see also ADR-XXX: Identifiers used in Self-Descriptions in appendix E
DID	Decentralized Identifier	https://www.w3.org/TR/did-core/
VC	Verifiable Credential	https://www.w3.org/TR/vc-data-model/
GX-SD	Gaia-X Self-Descriptions	http://w3id.org/gaia-x/core

***Table 1**: Definitions, Acronyms, and Abbreviations*

1.4 References

Abbreviation, Title	Description	Link
[GX-TAD], Gaia-X Technical Architecture Document	Gaia-X Technical Architecture Document outlining the general vision and concepts of Gaia-X.	Refer to annex "Gaia-X_Architecture_Document_2103"
[PRD], Gaia-X Policy Rules	Gaia-X Policy Rules intend is to identify clear controls to demonstrate European values of Gaia-X, such values including Openness, Transparency, Data Protection, Security and Portability.	Refer to annex "Gaia-X_Policy Rules_Document_2104"
[LDN], Linked Data Notification, W3C	Linked Data Notifications, W3C Recommendation 2 May 2017	https://www.w3.org/TR/ldn/
[EUCS] EUCS – Cloud Services Scheme	EUCS candidate scheme (European Cybersecurity Certification Scheme for Cloud Services)	https://www.enisa.europa.eu/publications/eucs-cloud-service-scheme
[TR02102] BSI TR-02102 Cryptographic Mechanisms	Within this Technical Guideline, the BSI presents an assessment of the security of selected cryptographic mechanisms, thereby giving some longer-term guidance in the selection of suitable cryptographic schemes.	https://www.bsi.bund.de/EN/Service-Navi/Publications/TechnicalGuidelines/tr02102/tr02102_node.html
[SOG-IS] SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms	Specify the requirements of the SOG-IS Crypto Evaluation Scheme related to the selection of cryptographic mechanisms. This document is primarily intended for developers and evaluators.	https://www.sogis.eu/documents/cc/crypto/SOGIS-Agreed-Cryptographic-Mechanisms-1.2.pdf
[GX-SD], Gaia-X Self Descriptions	A Self-Description expresses characteristics of an Asset, Resource, Service Offering or Participant and describes properties and Claims while being tied to the Identifier.	Refer to annex "Gaia-X_Architecture_Document_2103"
[IDM.AO]	GXFS IDM&Trust Architecture Overview	Refer to annex "GX_IDM_AO"

[SDE.DCS]	Specifications for Gaia-X Federation Service Sovereign Data Exchange – Data Contract Service	Refer to annex "SRS_GXFS_SDE_DCS"
[NF.SPBD]	Specification of non-functional Requirements for Gaia-X Federation Services - Security and Privacy by Design	Refer to annex "GXFS_Nonfunctional_Requirements_SPBD"
[TDR]	Gaia-X Federation Services Technical Development Requirements	Refer to annex "GXFS_Technical_Development_Requirements"

***Table 2**: References*

1.5 Document Overview

The first section of this document provides the general background for the Gaia-X Federation Service Data Exchange Logging Service (GX-DELS). The document is structured as follows. The second chapter will provide a general overview of the context of the GX-DELS (Product Perspective) and its interactions with other services and roles (user classes), including a general outline of the functionality and the internal mechanisms to achieve the functionality (Product Functions). The third chapter covers the functional and non-functional requirements to the GX-DELS, as well as the interfaces provided. In the fourth chapter the main feature and its relation, the requirements are described. Chapter five and six conclude with additional requirements and the validation aspects of the GX-DELS.
This document relies on the Gaia-X Self Descriptions Ontology (https://w3id.org/gaia-x/core), which is not part of this document. The GX-DELS interacts with and relies on the Gaia-X Federation Service Data Contract Service (refer to https://www.gxfs.de/federation-services/sovereign-data-exchange/data-contract-service/) and the Gaia-X Federation Service Identity & Trust (refer to [IDM.AO]). Compliance aspects in this specification rely on the Gaia-X Federation Service Compliance (refer to https://www.gxfs.de/federation-services/compliance/), which is not part of this document. The functionality of those services is not part of this document.
The content of this document relies on the Gaia-X Technical Architecture (refer to [GX-TAD]) and the Gaia-X Policy Rules (refer to [PRD]) , which are not part of this document but should be considered as important.

2. Product Overview

The promise of Gaia-X is twofold: First, often depicted as the lower half of the "X", Gaia-X will be a federated meta-cloud of European platform and infrastructure service providers. This is a mouthful, but it means just that a mesh of existing and vastly diverse PaaS and IaaS solutions are being built, joined by the overarching Gaia-X principles, and connected by the Gaia-X Federation Services. Second, often depicted as the upper half of the "X", Gaia-X promises a fully-fledged data ecosystem to enable flexible, secure, and sovereign data exchange. The Gaia-X Data Exchange Logging Service (GX-DELS), which is specified in this document, will improve data traceability and contract negotiation by creating a logging service that

can be used by the GX-DCS to guarantee lossless and auditable logging. The functionality of this version is limited but serves as a baseline for future data sovereignty aspects.

Every data transaction within Gaia-X consists of the following parts:

1. A **Data Asset**: Without Data Assets no data ecosystem. It is assumed that not only existing datasets, databases, and sensor streams will be part of the Gaia-X data ecosystem, but also that new business models and future start-ups will pop up everywhere as soon as the opportunities and benefits become apparent. Data Assets can be any static or dynamic data item that is a potential "thing" to be bought, e.g., a database of high-quality photos for ANN training, sensor streams from environmental measuring stations, models for 3-D printers, camera footage from a certain location and time interval, etc.
2. The **Self-Description** of the Data Asset: The Self-Description (GX-SD) is at the core of Gaia-X – every service and participant has one, and so do Data Assets. The SD of Data Assets has some rather tight restrictions and requirements because the SD contains not only the usual metadata information like data type, size, content description, etc. but also legal statements that transform the SD from a set of key-value pairs into the template of a legally binding contract. Data Contract negotiation leads to the transmission of the Data Asset either manually or automatically triggered, but as soon as the Data Asset is in the hands of the Data Consumer, enforcement of the Data Asset's usage policies is impossible. (This can be changed in the future by adding software or hardware connectors to the Gaia-X data ecosystem, but this is out of scope at the moment.) Therefore, legal policy enforcement becomes the natural fallback option. In order to get a trustworthy legal basis, a real contract must be forged between Data Provider and Data Consumer. The Self-Description is a *Ricardian contract*: A contract at law that is both human-readable and machine-readable, cryptographically rendered tamper-proof, verifiable in a decentralized fashion, and electronically linked to the subject of the contract, i.e., the Data Asset.
3. **Publication** of the Data Asset Self-Description: While the Data Asset resides with the Data Provider, the Data Asset Self-Description (Data Asset SD) has to be published in the Gaia-X Federated Catalogue (GX-FC) by the Data Provider.
4. **Search and Choice** of a Data Asset by the Data Consumer: Data Consumers can either access GX-FC's API directly to search, filter, and otherwise navigate available Data Asset SDs, or they can make use of the Gaia-X Portal, which accesses GX-FC behind the scenes and provides a neat GUI for browsing Data Asset SDs. Naturally, GX-FC must be able to search for and filter by all relevant Data Asset SD parameters, beginning with the data type, transmission type, keywords, pricing model, negotiation, and transmission details. As the last step, the Data Consumer electronically signals their interest and is then forwarded to GX-DCS, where contract negotiation takes place.
5. Contract **Negotiation and Signing**: In the case of solid, low-price Data Assets like picture or sound databases, contract negotiation consists of a simple signature of the data contract, which is then forwarded to the Data Provider who then initiates the data transmission. In many cases, however, Data Provider may wish to choose their customers or, at any rate, determine the price and terms of usage depending on the specific Data Consumer in question. A contract can contain logging

requirements for future billing, dispute resolution, and general auditing needs. If the contract is signed by both sides, data transmission begins.

6. Data **Transmission and Logging**: After a contract has been agreed upon and has been signed by both parties, data transmission from the Data Provider to the Data Consumer can commence. The contract negotiation can lead to both sides agreeing on a logging service (this document) which is then used by both sides to log data transactions. The GX-DELS provides features to enable integrity and access protected logging in a decentralized manner for both parties. Incidentally, GX-DCS issues and renews *log authorization tokens* needed for logging (see also *Renew Log Token* in GX-DCS specification in [SDE.DCS]).
7. **Billing**: Although it's clear that a fully functioning data ecosystem entails a marketplace where things can be bought and sold at a price, Gaia-X won't contain "Billing-as-a-Service" in its first incarnation. Nevertheless, the Data Asset SD surely contains pricing details; only the payment process itself must be realized by the Data Providers themselves. In future versions, a Gaia-X billing service will look up the data transmission logs in the appropriate GX-DELS instance and initiate the money transfer.

2.1 Product Perspective

GX-DELS is a stateless microservice. Storage of notifications is required and can be extended with performance-enhancing caching. GX-DELS cannot exist on its own: As depicted in the diagrams below, many Gaia-X Federation Services (GXFS) are needed to a data ecosystem and valid data exchange contracts are required and supported by the Data Contract Service (GX-DCS):

- The Gaia-X **Notarization Service** is needed to provide an anchor of trust; from the onboarding of new Gaia-X Participants up to and beyond the digitization of to-be-invented data quality labels, the Notarization Service provides an (not: the) interface between the Gaia-X and the rest of the world and anchors all other chains of trust within Gaia-X.
- The Gaia-X **Federated Catalogue**: This is the place where Data Providers publish Data Asset SDs. The fine details don't belong here, but in general, Gaia-X Federated Catalogue (GX-FC) expects Data Providers to operate an endpoint where GX-FC can subscribe to Data Asset SDs. This puts some initial setup work in the hands of the Data Providers, but it also solves the problem of keeping the catalog up to date in dealing with updates, certificate revocations, and the occasional Data Asset recall in case of legal problems or quality issues. GX-FC is always as up to date as possible.
- The Gaia-X **Portal**: While the bigger Participants will possibly wire up their own systems with the various GXFS backends, a neat and tidy frontend for browsing and searching, and comparing Gaia-X Assets of all kinds is an important service in itself. Naturally, behind the scenes, the Portal is provided all its information by GX-FC – in fact, the Portal is largely a GUI of GX-FC.
- The Gaia-X **Trust Service**: The Trust Service is paramount for validating signatures. To this end, it provides functionality to resolve DIDs and retrieve public keys of any Participant. The Trust Service is only part of Gaia-X's Identity and Access Management framework, but for GX-DCS it's the only part needed.
- The Gaia-X **Data Contract Service**: GX-DCS complements GX-DELS. Specific transactions on Gaia-X have to be agreed on by Data Provider and Data Consumer including a contract as a foundation

for the data exchange. The GX-DCS supports the signing of a contract and provides log tokens that are a prerequisite for the data transaction including the logging or notification on data exchange events.

GX-DELS is directly coupled to other GXFS services. This coupling is realized through defined interfaces; the inside of GX-DELS is a black box to the rest of Gaia-X.

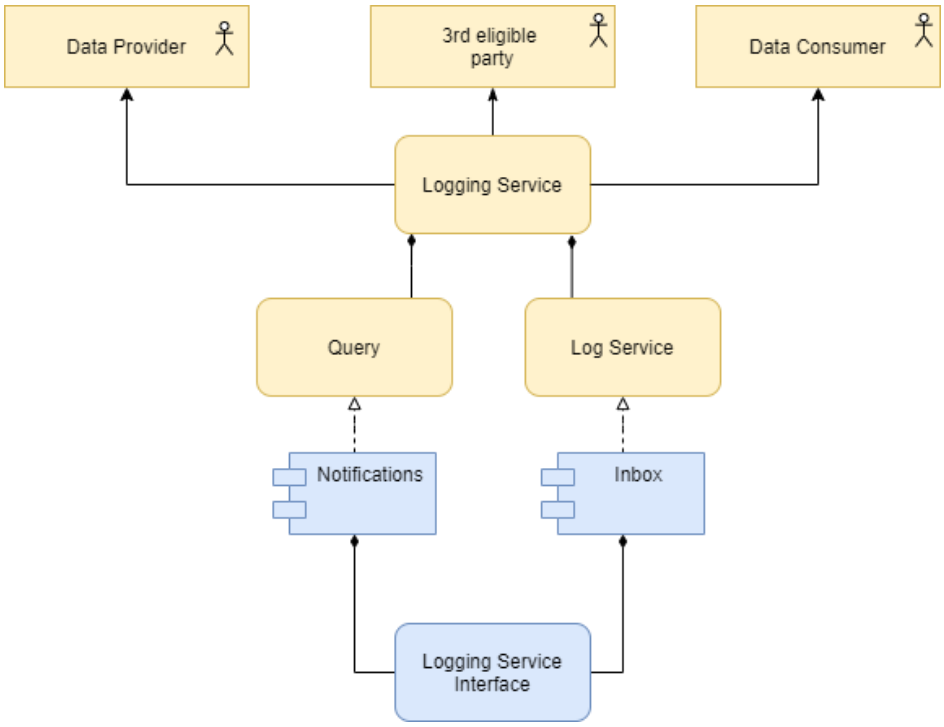


***Figure 1**: Overview of the Logging Service and its dependencies*

2.2 Product Functions

The two main functions of the Gaia-X Service Instance (GX-SI) GX-DELS are, as explained above in section 2:

- **Log Notifications into the GX-DELS Inbox**: Data Provider and Data Consumer can send notifications on events to the GX-DELS Inbox including a logging token to verify the existence of a contract.
- **Query Log Notifications from GX-DELS Inbox:** Data Provider and Data Consumer can query Log Notifications from the GX-DELS. 3rd eligible parties can be enabled to query information from the GX-DELS.

This functionality is extended by the support of forced logging. Here an authorization token contains the information on whether forced confirmation logging is enabled:

- This enables the possibility for the Data Provider to send the data encrypted and provide the decryption key only after reception of the logging confirmation

- Note: A solution where GX-DELS takes care of transmitting the decryption key from a trusted third party is not possible due to patent restrictions

The following figure provides an overview of the main functionality of GX-DELS:

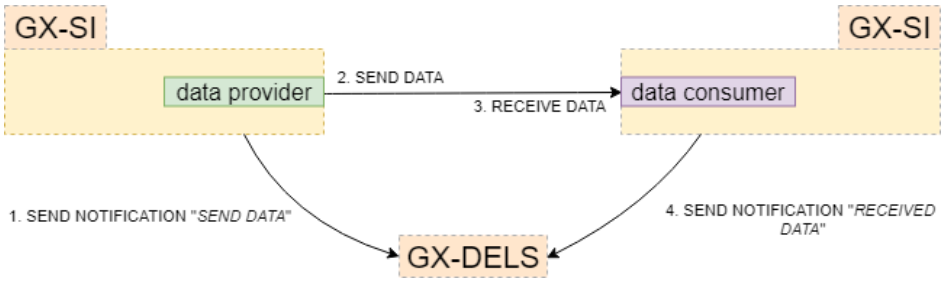


***Figure 2**: Overview of Gaia-X Data Exchange Logging Service (GX-DELS)*

2.2.1 Data Provider SEND NOTIFICATION "*SEND DATA*"

Data Provider as given GX-SI is prepared to send data to Data Consumer and sends first notification to GX-DELS. The payload is equipped with a unique identifier (generated in providers domain) and also with a given identifier of receiving Data Consumer (among other attributes). This leads to a Log Entry (as an inbox-notification) in GX-DELS. GX-DELS responds with a corresponding *notification identifier* [n-id-P].

2.2.2 Data Provider SEND DATA

Data Provider sends data, the main information as expected by given Data Consumer. This payload is wrapped with some meta-data (like a data contract), but also with given notification identifier [n-id-P] exposed by GX-DELS' response to 'SEND NOTIFICATION "*SEND DATA*"'.

2.2.3 Data Consumer RECEIVED DATA

Data Consumer receives data sent by Data Provider.

2.2.4 Data Consumer SEND NOTIFICATION "*RECEIVED DATA*"

Data Consumer sends a notification to GX-DELS, containing given notification identifier [n-id-P], provided by Data Provider.

This leads to a Log Entry (inbox-notification) in GX-DELS.

- GX-DELS will check Data Consumer for correctness (here: by its own identifier, for example) related to given [n-id-P]
- GX-DELS will finish and responses with
    - information provided by [n-id-P]
    - identifier [n-id-C] of new notification

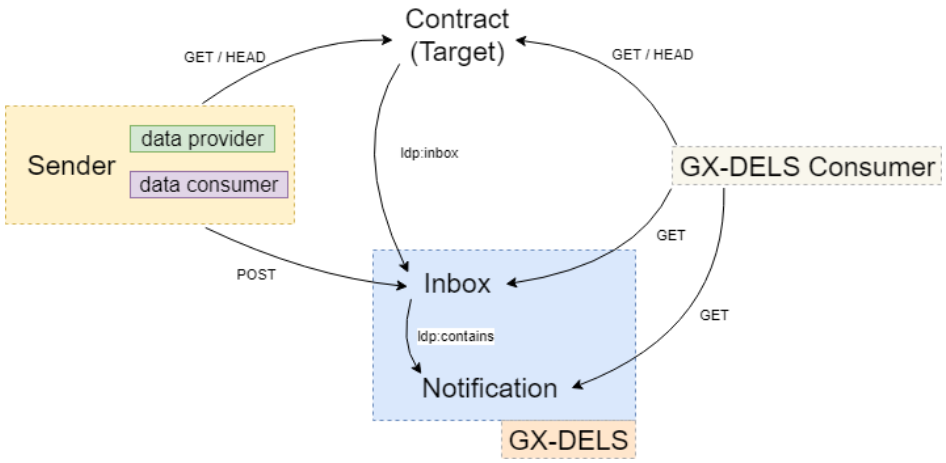


***Figure 3**: Inbox Communication of Gaia-X Logging Service (GX-DELS)*

2.3 Product Constraints

SDE.LS.01 Technical Architecture Document
The document [GX-TAD] provides the common basis for all specifications and implementation. The specifications and requirements from the Architecture Document [GX-TAD] MUST be taken into account during implementation.

2.4 User Classes and Characteristics

The Gaia-X Technical Architecture Document [GX-TAD] specifies different classes of users and roles in Gaia-X. For a general overview refer to this document. The Specification for Identity Management in Gaia-X [IDM.AO] covers more details on identities, users, and roles. In the scope of this document, those aspects have to be considered.

The GX-DELS differentiated two different user classes to interact with the service:

- **SDE.LS.05 Contracted parties:** Data Provider and Data Consumer. Both parties send notifications to the DELS subsequently to a contract and related to a data exchange transaction. They act as the sender of notifications. Additionally, both can query the DELS for notifications from the inbox and can read notifications. They can act as a consumer.
- **SDE.LS.06 3rd eligible parties:** additional eligible parties may consume notifications from the DELS. Therefore, they must be entitled by the contracted parties. As it remains unclear for this version of this specification how 3rd parties can be entitled to access notifications, this user group is not specified here. In subsequent versions of the specification, the access of 3rd parties will be addressed. Therefore, 3rd party access to the DELS is currently not normative but covered here for information and to be considered in any architectural- or implementation-related decision.

2.5 Operating Environment

Please refer to [TDR].

2.6 User Documentation

User documentation must be provided along with the GX-DELS. The documentation must contain sufficient information to operate and use the GX-DELS. This includes:

- Administration Documentation
    - **SDE.LS.07** Deployment procedures: The developer must provide guidance documentation describing the deployment and installation procedures for the Federation Service
    - **SDE.LS.08** Modes of operation: The developer must provide documentation describing all modes of operation including operational error including instructions for secure operation and recovery
    - **SDE.LS.09** Software structure: The developer must provide an overview on the system architecture and the components and modules of the system
    - **SDE.LS.10** Security Concept: The developer must provide a security concept for the secure operation of the GX-DELS
- Gaia-X Participant Documentation
    - **SDE.LS.11** Software Description: The developer must provide a short description of the Federation Service, its purpose, and its intended usage
    - **SDE.LS.12** Interface Usage: The developer must provide descriptions of all GX-DELS interfaces and their usage for each Gaia-X participant, i.e., Data Providers and Data Consumers

Further requirements regarding the documentation can be found in [TDR].

2.7 Dependencies

GX-DELS and GX-DCS rely on the Log Token, as specified in the appendix B.

3. Requirements

3.1 External Interfaces

GX-DELS acts as a Linked Data Notification (LDN) inbox. LDN is embedded in well-known and described Linked Data Platform (LDP) and in fact, this, strongly bound to given REST-APIs as shown here.
Given Gaia-X Service Instance Data Provider (GX-SI Data Provider, Data Provider) and GX-SI Data Consumer (Data Consumer) behave as a sender of notifications to given receiver's inbox. Both senders are aware of LDNs inbox by requesting given target and get needed inbox-URL.

In the terminology of LDN there might be another party, the inbox-consumer, querying for present notifications.
So “translating” LDN-terminology to GX-DELS, the Data Provider and Data Consumer (as GX-SIs) send Log Entries as LDN-notifications to GX-DELS inbox. The LDN-target is a contract provided by Gaia-X Data Contract Service (GX-DCS), which itself is equipped with an URL of needed GX-DELS inbox endpoint.
Discovering an GX-DELS as an endpoint of given REST-API will be done by requesting given contract in GX Service Instance “Data Contract Service” (GX-DCS).

3.1.1 User Interfaces

ID & Title	Description	Verification Method
GX-DELS.IR.001 **Administrative GUI**	The GX-DELS must provide a basic administrative UI for the service operator.	Documentation Testing
GX-DELS.IR.002 **Administrative GUI display items**	• The GUI must be able to present to the administrator: ○ Show log retention period. ○ Show the storage capacity used by the service. ○ Show status of pruning of outdated entries. ○ Show status of backup mechanism.	Documentation Testing
GX-DELS.IR.003 **Administrative GUI configuration capabilities**	• The GUI must allow to configure some aspects of the GX-DELS: ○ Configure log retention period. ○ Configure backup behavior.	Documentation Testing

***Table 3**: Requirements User Interfaces*

3.1.2 Hardware Interfaces

The GX-DELS does rely on hardware trust anchors such as an HSM, a TPM, or the like. The interfacing must be present to store secrets in a non-manipulatable and protected way. For scalability, performance, and reliability reasons it SHOULD be realized in a virtualized environment.

3.1.3 Software Interfaces

ID & Title	Description	Verification Method
GX-DELS.LS.004 **Administrative GUI**	The GX-DELS is designed for a microservices architecture. It solely provides a REST interface as defined in section 3.1.4 Communications Interfaces as an external interface. It does not provide any software interfaces for source code level integration into other components.	Documentation Testing Code Review

***Table 4**: Requirements Software Interfaces*

3.1.4 Communications Interfaces

The main communication interface of given GX-DELS is based on the mechanics of Linked Data Notification (LDN, see: https://www.w3.org/TR/ldn/) and so it is heavily based on HTTP.

Example: GX-SI Data Provider **discovers inbox** by HEAD-request at given contract-endpoint provided **by GX-DCS**.

```
HEAD /contracts/1001 HTTP/1.1
Host: dcs.gaia-x.com

HTTP/1.1 200 OK
Link: <https://dels.gaia-x.com/inbox/>; rel="http://www.w3.org/ns/ldp#inbox"
```

Example: GX-SI Data Provider **discovers inbox** by GET-request at given contract-endpoint provided **by Data contract Service (GX-DCS, DCS)**.

```
GET /contracts/1001 HTTP/1.1
Host: dcs.gaia-x.com
Accept: application/ld+json

HTTP/1.1 200 OK
Content-Type: application/ld+json

{
  "@context": "https://www.w3.org/ns/ldp",
  "@id": "https://dcs.gaia-x.com/contracts/1001",
  "inbox": "https://dels.gaia-x.com/inbox/"
}
```

Example: GX-SI Data Provider **asks for options** by OPTIONS-request to inbox-endpoint of GX-DELS Service Instance “dels.gaia-x.com” and gets its response with granted HTTP-methods and accepted formats (like ‘*ld+json*’ or ‘*turtle*’, for example).

```
OPTIONS /inbox/ HTTP/1.1
Host: dels.gaia-x.com

HTTP/1.1 200 OK
Allow: GET, HEAD, OPTIONS, POST
```

```
Accept-Post: application/ld+json, text/turtle
```

Example: GX-SI Data Provider **hands out a notification "SendDataNotification"** by POST-request to inbox-endpoint of GX-DELS Service Instance "dels.gaia-x.com" and gets its response with the location of resulting notification URL.

```
POST /inbox/ HTTP/1.1
Host: dels.gaia-x.com
Content-Type:     application/ld+json;profile="https://www.gax.org/ns/datacon-
tracts"
Content-Language: en

{
  "@context": "https://www.gax.org/ns/datacontracts",
  "@type": "gax-dels:SendDataNotification",
  "id": "https://www.gx-si-provider.net/1a2s3d4f",
  "gax-dels:issued": "2021-03-30T02:02:02Z",
  "gax-dcs:contract": "https://dcs.gaia-x.com/contracts/1001",
  "gax-dels:sender": "",
  "gax-dels:dataContract": "",
  "gax-dels:receiver": ""
}

HTTP/1.1 201 Created
Location: http://dels.gaia-x.com/inbox/42-42-42-42
```

Example: A consumer (as standalone GX-SI, *not* a Data Provider or Data Consumer, as qouted in this specification**) queries a special notification**.

```
GET /inbox/ HTTP/1.1
Host: dels.gaia-x.com
Accept: application/ld+json
Accept-Language: en-GB,en;q=0.8, en-US;q=0.6

HTTP/1.1 200 OK
Content-Type: application/ld+json
Content-Language: en

{
  "@context": "http://www.w3.org/ns/ldp",
  "@id": "http://dels.gaia-x.com/inbox/",
  "contains": [
    "https://dels.gaia-x.com/inbox/42-42-42-42"
  ]
}
```

3.2 Functional

ID & Title	Description	Verification Method
GX-DELS.IR.005 **Inbox**	The GX-DELS MUST provide an inbox to store notifications permanently. The inbox MUST be accessed via defined interfaces GX-DLES.IR.006, GX-DLES.IR.007, GX-DLES.IR.008, GX-DLES.IR.009 . The notifications stored in the inbox MUST comply with the message format defined in GX-DLES.IR.009. Notifications MUST be validated before entering the inbox. All notifications in the inbox MUST be stored permanently. The minimal and maximal storage duration of notifications MUST be configurable by the operator of the GX-DELS. The inbox discovery MUST follow the W3C Linked Data Notification protocol, the URI of the inbox is provided by the notification target. Valid HTTPs endpoints are available and can consume and provide notifications and response messages.	Documentation Testing
GX-DELS.IR.006 **Receive Notifications**	The GX-DELS can receive notification, i.e., events send by Data Provider or Data Consumer to the inbox of the GX-DELS. Therefore, a proper interface based on W3C-LDN MUST be implemented to provide an inbox. The target notification must comply with Gaia-X self-descriptions. The inbox MUST be able to process GET and POST requests on the INBOX URL. Upon receipt of a POST request, if the notification resource was processed successfully, GX-DELS MUST respond with status code 201 Created and the Location header set to the URL from which the notification data can be retrieved. If the request was queued to be processed asynchronously, the receiver must respond with a status code of 202 Accepted and include information about the status of the request in the body of the response. If the constraints for the notification are not met, X-DELS MUST return the appropriate 4xx error code. GX-DELS must accept notifications where the request body is JSON-LD, with the Content-Type: application/ld+json, which may include a profile URI.	Documentation Testing

GX-DELS.IR.007 **Query Notifications**	The GX-DELS MUST provide an interface to query notification from the GX_DELS inbox. This includes querying a list of notifications based on filters and to query one notification. The access to the inbox and to the notification MUST be access controlled. The notification MUST be directly available via its identifier (URI). The query of the inbox MUST support filters for the notifications stored in the inbox: • the requester is a contracted party • where the requester is the sender of the notification • by contract identifier A successful GET request on the Inbox must return a HTTP 200 OK with the URIs of notifications, subject to the requester's access (returning 4xx error codes as applicable). GX-DELS may list only URIs of notifications in the Inbox that the consumer is able to access. The Inbox URL must use the http://www.w3.org/ns/ldp#contains predicate to refer to the notifications. Each notification must be an RDF source. If non-RDF resources are returned, the consumer may ignore them. A successful GET request on the notification URI must return a HTTP 200 OK subject to the requester's access (returning 4xx error codes as applicable). The JSON-LD content type must be available for all resources, but clients may send Accept headers preferring other content types (RFC7231 Section 3.4 - Content Negotiation). If the client sends no Accept header, the server may send the data in JSON-LD or any format which faithfully conveys the same information (e.g., Turtle). Any additional description about the Inbox itself may also be returned.	Documentation Testing

GX-DELS.IR.008 **Query Notifications**	The GX-DELS MUST provide an interface to query notification from the GX_DELS inbox. This includes querying a list of notifications based on filters and to query one notification. The access to the inbox and to the notification MUST be access controlled. The notification MUST be directly available via its identifier (URI). The query of the inbox MUST support filters for the notifications stored in the inbox: • the requester is a contracted party • where the requester is the sender of the notification • by contract identifier A successful GET request on the Inbox must return a HTTP 200 OK with the URIs of notifications, subject to the requester's access (returning 4xx error codes as applicable). GX-DELS may list only URIs of notifications in the Inbox that the consumer is able to access. The Inbox URL must use the http://www.w3.org/ns/ldp#contains predicate to refer to the notifications. Each notification must be an RDF source. If non-RDF resources are returned, the consumer may ignore them. A successful GET request on the notification URI must return a HTTP 200 OK subject to the requester's access (returning 4xx error codes as applicable). The JSON-LD content type must be available for all resources, but clients may send Accept headers preferring other content types (RFC7231 Section 3.4 - Content Negotiation). If the client sends no Accept header, the server may send the data in JSON-LD or any format which faithfully conveys the same information (e.g., Turtle). Any additional description about the Inbox itself may also be returned.	Documentation Testing
GX-DELS.IR.009 **Message Structure**	The message structure must comply with Gaia-X Self-Description (see GX-SD) attributes and have the subsequent structure and content. The Message can be validated against Gaia-X Self-Description schema.	Testing

GX-DELS.IR.010 **Monitoring**	The DELS must support monitoring by compliance monitoring services. These need to be in line with the GX-CAM service and SHOULD be able to satisfy these metric collections: SystemComponentsIntegrity, CyberSecurityCertification, TlsVersion, OAuthGrantTypes, TlsCipherSuite, AtRestEncryption.	Documentation Testing
GX-DELS.IR.011 **Forced Logging Callback**	For forced confirmation logging mode, the Provider MUST be able to register a callback at the logging service. The DELS must send logging confirmation to the registered callback to notify the Provider that the Consumer logged the transaction.	Documentation Testing
GX-DELS.IR.012 **Log Entry Persistence Integrity Protection**	The GX-DELS MUST provide a mechanism to enforce the integrity of the overall Log Entry storage. The GX-DELS MUST provide a possibility to verify the integrity in regular intervals by the GX-CAM (see above in GX-DLES.IR.010).	Documentation Code Review Testing
GX-DELS.IR.012 **Log Entry Integrity Protection**	The single Log Entries MUST be integrity protected and validated by the participant's signature. The signature MUST be based on the private key belonging to the participant's Gaia-X identity.	Documentation Code Review Testing
GX-DELS.IR.013 **Log Entry Storage period**	The DELS must store the Log Entries for a configurable time period according to applying legal obligations.	Documentation Testing
GX-DELS.IR.014 **Log Entry Storage Encryption**	The DELS must store the Log Entries encrypted so it is inaccessible by raw access to the persistence layer. The persisted data must be encrypted using state-of-the-art methods, using key lengths as stated in the general requirements.	Documentation Code Review

***Table 5**: Functional Requirements*

3.3 Other Nonfunctional Requirements

This section states additional, quality-related property requirements that the functional effects of the software should present

3.3.1 Performance Requirements

ID & Title	Description	Verification Method
GX-DELS.NFR.001 **Performance by design**	The component SHOULD be designed and implemented with performance in mind. In particular, it MUST be implemented in a non-blocking way.	Performance Testing Code Review
GX-DELS.NFR.002 **Scalability**	The component MUST be scalable and able to handle multiple requests. It MUST be possible to run multiple instances of the component simultaneously.	Deployment procedures review

Table 6: *Non-functional Requirements Performance Requirements*

3.3.2 Safety Requirements

ID & Title	Description	Verification Method
GX-DELS.NFR.003 **Availability**	The component SHOULD be designed and implemented with performance in mind. In particular, it MUST be implemented in a non-blocking way.	Performance Evaluation Code Review
GX-DELS.NFR.004 **Scalability**	The GX-DELS must be designed in a way to avoid outage in case of failures, i.e., the Recovery Time Objective (RTO) must be 0. A failover mechanism must be in place to hand over operations in case of failure. This is purely related to the service.	Documentation
GX-DELS.NFR.005 **Storage Redundancy and Backup**	The GX-DELS log entry storage must be designed in a way to avoid data loss in case of failures. The Recovery Point Objective (RPO) and the Recovery Time Objective (RTO) must be 0.	Documentation

Table 7*: Non-functional Requirements Safety Requirements*

3.3.3 Security Requirements

3.3.3.1 General Security Requirements

Each Gaia-X Federation Service must fulfill the requirements stated in the document “Specification of non-functional Requirements Security and Privacy by Design” [NF.SPBD].

Federation Services specific requirements will be documented in the next chapter.

3.3.3.2 Service Specific Security Requirements

This chapter describes the service-specific requirements, which will extend the requirements defined in the chapter above.

ID & Title	Description	Verification Method
GX-DELS.SEC.001 **Transport Layer Security**	Each communication with an interface of the GX-DCS MUST utilize TLS of at least version 1.2. It SHALL use TLS in version 1.3.	Documentation Code Review
GX-DELS.SEC.002 **Remote administration**	If the component can be remotely administrated by the Federator, the communication MUST utilize a secure communication channel such as SSH or VPN.	Documentation Code Review
GX-DELS.SEC.003 **State-of-the-art cryptography**	Cryptographic algorithms and cipher suites MUST be state-of-the-art and chosen in accordance with official recommendations. Those recommendations MAY be those of the German Federal Office for Information Security (BSI) [TR02102] or SOG-IS [SOG-IS].	Documentation Code Review
GX-DELS.SEC.004 **Authentication and Authorization**	Authentication and Authorization: The product MUST grant access to its services only to authenticated and authorized Gaia-X participants. The authentication MUST be based on a valid Gaia-X Identity as defined in [IDM.AA].	Documentation Code Review
GX-DELS.SEC.005 **Data Confidentiality**	The product MUST encrypt any sensitive data that is stored persistently, e.g., the Log Entry Storage (see above). For the storage of secrets further, see GX-DLES.SEC.006 Storage of Secrets.	Documentation Code Review
GX-DELS.SEC.006 **Storage of Secrets**	Storage of Secrets: Secrets such as keys and other cryptography material MUST be stored in a secure and protected environment, e.g., a TPM, HSM, or TEE to ensure their confidentiality and integrity.	Documentation Code Review

GX-DELS.SEC.007 **Integrity Protection for Configuration**	Where the functionality of the Service is based on configuration files, those files MUST be authenticated, and integrity protected.	Documentation Code Review
GX-DELS.SEC.008 **Integrity Protection for the Service**	The Federator MUST utilize security measures to ensure the integrity of the GX-DELS. It MAY support proof of the integrity to remote parties using an additional interface (Remote Attestation).	Documentation Code Review
GX-DELS.SEC.009 **Integration into Gaia-X IAM DID architecture**	Access to interfaces must only be granted to parties with a valid Gaia-X DID as specified in WP1[2] [IDM.AO].	Documentation Code Review
GX-DELS.SEC.010 **Integration into Gaia-X IAM role model**	The service MUST support roles or credentials specified in [IDM.AO].	Documentation Code Review

***Table 8**: Non-functional Requirements Service Specific Security Requirements*

3.3.4 Software Quality Attributes

ID & Title	**Description**	**Verification Method**
GX-DELS.QR.001 **Programming Style**	The implementation SHOULD follow best practices and a consistent style for coding, e.g., the source code shall be clearly structured and modularized; there should be no dead code; function and variables shall be clear and self-explaining. The code MUST be well documented to support adaptability, maintainability, and usability of the component.	Documentation Code Review
GX-DELS.QR.002 **Testing**	The development of the component MUST include functional and security testing, source code audits and penetration testing as described in the document [NR.SPBD].	Testing

***Table 9**: Non-functional Requirements Software Quality Attributes*

[2] Please refer to appendix D for an overview and explanation of the Work Packages (WP).

3.3.5 Business Rules

Only Data Providers and Data Consumers with a valid Gaia-X Identity [IDM.AO] possessing a valid Log Token from the Contract Service [SDE.DCS] may write and read Log Entries. However, all participants must only be allowed to access those log entries they are involved in.

3.4 Compliance

The Gaia-X Federation Service shall fulfill the cybersecurity control set of the EUCS [EUCS] Annex A according to its assigned Assurance Level as described in the document [NF.SPBD] “Specification of non-functional Requirements Security and Privacy by Design.

GX-DELS must be compliant with the metric collection of GX-CAM (see above GX-DLES.IR.010).

3.5 Design and Implementation

Please refer to [TDR].

4. System Features

4.1 Inbox notifications

4.1.1 Description and Priority

The major functionality of the GX-DELS is the realization of an inbox that can receive notifications from senders, as explained in section 2. Priority: HIGH.

4.1.2 Stimulus/Response Sequences

Stimulus/Response sequence is described in section 2.

4.1.3 Functional Requirements

- GX-DELS.IR.005 Inbox
- GX-DELS.IR.006 Receive Notifications
- GX-DELS.IR.007 Query Notifications
- GX-DELS.IR.008 Query Notifications
- GX-DELS.IR.009 Message Structure
- GX-DELS.IR.011 Forced Logging Callback
- GX-DELS.IR.012 Log Entry Persistence Integrity Protection
- GX-DELS.IR.012 Log Entry Integrity Protection

- GX-DELS.IR.013 Log Entry Storage period
- GX-DELS.IR.014 Log Entry Storage Encryption
- GX-DELS.NFR.001 Performance by design
- GX-DELS.NFR.002 Scalability
- GX-DELS.NFR.003 Availability
- GX-DELS.NFR.004 Scalability
- GX-DELS.NFR.005 Storage Redundancy and Backup
- GX-DELS.SEC.001 Transport Layer Security
- GX-DELS.SEC.003 State-of-the-art cryptography
- GX-DELS.SEC.004Authentication and Authorization
- GX-DELS.SEC.005 Data Confidentiality
- GX-DELS.SEC.006 Storage of Secrets
- GX-DELS.SEC.007 Integrity Protection for Configuration
- GX-DELS.SEC.008 Integrity Protection for the Service
- GX-DELS.SEC.009 Integration into Gaia-X IAM DID architecture
- GX-DELS.SEC.010 Integration into Gaia-X IAM role model
- GX-DELS.QR.002 Programming Style
- GX-DELS.QR.003 Testing

4.2 Query Inbox

4.2.1 Description and Priority

Notification sent to the inbox must be queried by an eligible party. Therefore, access control is mandatory and filtering as explained above. Priority: HIGH.

4.2.2 Stimulus/Response Sequences

Stimulus/Response sequence is described in section 2.

4.2.3 Functional Requirements

- GX-DELS.IR.005 Inbox
- GX-DELS.IR.006 Receive Notifications
- GX-DELS.IR.007 Query Notifications
- GX-DELS.IR.008 Query Notifications

- GX-DELS.IR.009 Message Structure
- GX-DELS.IR.010 Monitoring
- GX-DELS.IR.011 Forced Logging Callback
- GX-DELS.IR.012 Log Entry Persistence Integrity Protection
- GX-DELS.IR.012 Log Entry Integrity Protection
- GX-DELS.IR.013 Log Entry Storage period
- GX-DELS.IR.014 Log Entry Storage Encryption
- GX-DELS.NFR.001 Performance by design
- GX-DELS.NFR.002 Scalability
- GX-DELS.NFR.003 Availability
- GX-DELS.NFR.004 Scalability
- GX-DELS.NFR.005 Storage Redundancy and Backup
- GX-DELS.SEC.001 Transport Layer Security
- GX-DELS.SEC.002 Remote administration
- GX-DELS.SEC.003 State-of-the-art cryptography
- GX-DELS.SEC.004Authentication and Authorization
- GX-DELS.SEC.005 Data Confidentiality
- GX-DELS.SEC.006 Storage of Secrets
- GX-DELS.SEC.007 Integrity Protection for Configuration
- GX-DELS.SEC.008 Integrity Protection for the Service
- GX-DELS.SEC.009 Integration into Gaia-X IAM DID architecture
- GX-DELS.SEC.010 Integration into Gaia-X IAM role model
- GX-DELS.QR.002 Programming Style
- GX-DELS.QR.003 Testing

4.3 Response types

4.3.1 Description and Priority

The GX-DELS must send a response to a notification sent to its service interface stating if the operation was executed successfully or if any error occurred. In case of an error or exception, the DELS shall add helpful information to resolve the error or exception, if possible.

4.3.2 Stimulus/Response Sequences

The GX-DELS responds to messages sent to the Service Interfaces. It includes Header information and Status Information. The existence of a valid contract is a precondition for sending notifications to the GX-DELS. Additionally, only valid and verified Gaia-X identities can send or read notifications. After a notification is received by the GX-DELS, it must verify the content and shall send a proper response.

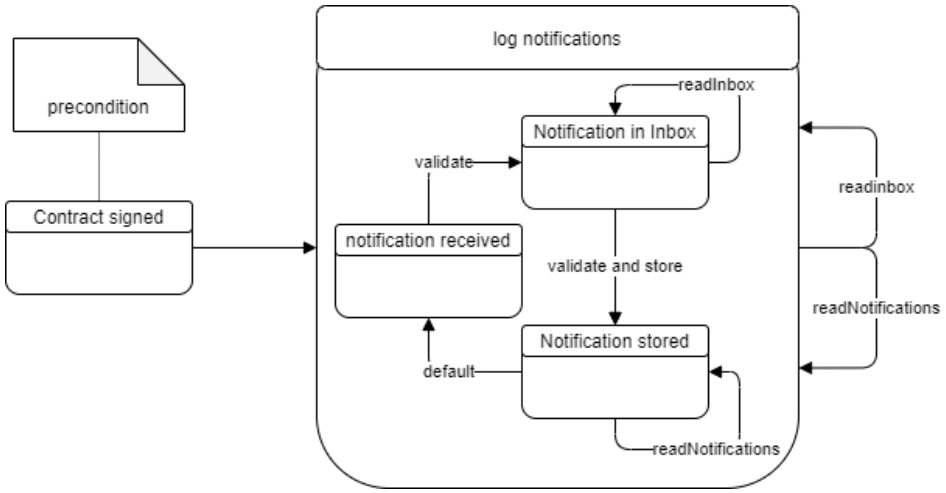


***Figure 4**: Stimulus/Response Sequences*

Response Code	**Meaning**	**Message**
1xx	informational response – the request was received, continuing process. The response may be used as appropriate and specified in RFC 7231	refer to
200	Standard response to serve GET request. Header information and body must be included as appropriate.	HTTP/1.1 OK
201	Notification Created, and the Location header set to the URL from which the notification data can be retrieved.	HTTP/1.1 201 Created
202	Accepted, the message was received and will be processed asynchronously, GX-DELS includes information about the status of the request in the body of the response.	HTTP/1.1. 202 Accepted
3xx	All responses with status code 3xx shall be avoided.	

400	Bad request Must be used for all client errors that are not specified afterwards. Additional information shall be provided via header or body		
401	Unauthorized Must be used for unauthorized request.		
403	Forbidden Must be responded for access to a notification or an inbox in case the consumer is not a contracted party or entitled to access. This status code must be responded when a notification or inbox does not exist.		
404	Not Found for all resources that are not specified if a notification or inbox does not exist, the DELS must answer with 403.		
405	Method not allowed		
500	internal server error		
Response Header Key	**Header Value**	**Cardinality**	**Description**
Location	http://gx-dels.gaia-x.com/inbox/5c6ca040		URL of given (new) notification

***Table 10**: Response Types*

4.3.3 Functional Requirements

- GX-DELS.IR.001 Administrative GUI
- GX-DELS.IR.002 Administrative GUI display items
- GX-DELS.IR.003 Administrative GUI configuration capabilities
- GX-DELS.LS.004 Administrative GUI

- GX-DELS.IR.005 Inbox
- GX-DELS.IR.006 Receive Notifications
- GX-DELS.IR.007 Query Notifications
- GX-DELS.IR.008 Query Notifications
- GX-DELS.IR.009 Message Structure
- GX-DELS.IR.010 Monitoring
- GX-DELS.IR.011 Forced Logging Callback
- GX-DELS.IR.012 Log Entry Persistence Integrity Protection
- GX-DELS.IR.012 Log Entry Integrity Protection
- GX-DELS.IR.013 Log Entry Storage period
- GX-DELS.IR.014 Log Entry Storage Encryption
- GX-DELS.NFR.001 Performance by design
- GX-DELS.NFR.002 Scalability
- GX-DELS.NFR.003 Availability
- GX-DELS.NFR.004 Scalability
- GX-DELS.NFR.005 Storage Redundancy and Backup
- GX-DELS.SEC.001 Transport Layer Security
- GX-DELS.SEC.002 Remote administration
- GX-DELS.SEC.003 State-of-the-art cryptography
- GX-DELS.SEC.004Authentication and Authorization
- GX-DELS.SEC.005 Data Confidentiality
- GX-DELS.SEC.006 Storage of Secrets
- GX-DELS.SEC.007 Integrity Protection for Configuration
- GX-DELS.SEC.008 Integrity Protection for the Service
- GX-DELS.SEC.009 Integration into Gaia-X IAM DID architecture
- GX-DELS.SEC.010 Integration into Gaia-X IAM role model
- GX-DELS.QR.002 Programming Style
- GX-DELS.QR.003 Testing

Appendix A: Glossary

The glossary is part of the Gaia-X Technical Architecture Document [GX-TAD].

Appendix B: Log Token specification

The rationale behind the log token is motivated by the fundamental need of logging events in an immutable and verifiable way. The GX-DELS enables the logging of data transactions to make data transactions reproducible and to enable future billing services. The log token is issued by GX-DCS since every logging event needs to be related to a valid data contract.

The log token is compliant with the JSON Web Token Format (https://tools.ietf.org/html/rfc7519). It consists of a header, which specifies the token type and signing algorithm. The token body contains a number of claims, specified in the table below. The signature can be used to verify the authenticity and integrity of the token.

Token format

```
LogToken Header:
{
  "alg": "HS256",
  "typ": "JWT"
}
LogToken Body: {
  gax-dcs:logID: "(128-bit UUID)",
  gax-dcs:dataTransactionID: "(Transaction ID)",
  gax-dcs:contractID: "(contract ID)",
  iss: "(Logging Service ID)"
  sub: "(Participant ID)",
  aud: "(GX-DELS identifier)",
  exp: Token lifetime
}
LogToken Signature:
// DCS signs whole token, compliant to https://tools.ietf.org/html/rfc7519
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  secret)
```

Claim	Description
gax-dcs:logID	Identifier of the log token
gax-dcs:dataTransactionID	Identifier of the overarching data transaction. Used to reference all linked logging events.

gax-dcs:contractID OR gax-dcs:contract / gax-dcs:dataContract	Identifier of the contract that is the baseline for the data transaction associated with the logging event.
iss	The identifier of the GX-DCS issuing the token.
sub	Identifier if the participant that requested the log token.
aud	The identifier of the logging service the token is intended for.
exp	The expiration timestamp, specifying the lifetime of the token.

Token verification

Tokens are signed with a private key of GX-DCS. Token validation and verification must be performed compliant to DID/VC validation mechanisms specified in [IDM.AO].

Appendix C: Ontology

Gaia-X Data Exchange Logging Service (GX-DELS) ontology, separated by an own namespace (here: 'gax-dels', so given classes and properties will be discussed before merging it into main Gaia-X ontology (here: 'gax'). 'gax-dels' also uses Data Contract Service (GX-DCS) ontology (here: 'gax-dcs'), at least referring to data contracts.

```
@prefix cc:        <http://creativecommons.org/ns#> .
@prefix dct:       <http://purl.org/dc/terms/> .
@prefix foaf:      <http://xmlns.com/foaf/0.1/> .
@prefix ldp:       <http://www.w3.org/ns/ldp#> .
@prefix owl:       <http://www.w3.org/2002/07/owl#> .
@prefix rdf:       <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:      <http://www.w3.org/2000/01/rdf-schema#> .
@prefix vann:      <http://purl.org/vocab/vann/> .
@prefix voaf:      <http://purl.org/vocommons/voaf#> .
@prefix void:      <http://rdfs.org/ns/void#> .
@prefix xsd:       <http://www.w3.org/2001/XMLSchema#> .

# REM: Gaia-X main ontology, as provided for Gaia-X Self Description
@prefix gax:       <http://w3id.org/gaia-x/core#> .

# REM: as long we didn't know if it's aligned with <gax:> we leave it this
way...
# REM: dcs = 'Data Contract Service'
@prefix gax-dcs:   <http://w3id.org/gaia-x/data-contract-service/v1#> .

# REM: this Ontology Gaia-X Data Exchange Logging Service
```

```
# REM: as long we didn't know if it's aligned with <gax:> we leave it this
way...
@prefix gax-dels:  <http://w3id.org/gaia-x/logging-service/v1#> .

gax-dels:
    a                              voaf:Vocabulary, owl:Ontology ;
    rdfs:label                     "Gaia-X DELS"@en ;
    dct:title                       "Gaia-X Ontology for Data Exchange Logging
Service"@en ;
    cc:license                     <http://www.apache.org/licenses/LICENSE-2.0>
;
    dct:creator                   "Gaia-X Federation Service Specification WP3"
;
    dct:contributor                <https://github.com/jlangkau> ;
    dct:created                   "2021-03-09T12:00:00+01:00"^^xsd:dateTimeStamp
;
    dct:modified                  "2020-03-29T12:00:00+01:00"^^xsd:dateTimeStamp
;
    owl:versionInfo                "0-0-1" ;
    owl:versionIRI                 "http://gaia-x.eu/gaiaxOntology/1.0.0>" ;
    vann:preferredNamespaceUri      "http://w3id.org/gaia-x/logging-service/v0-
0-1#" ;
    vann:preferredNamespacePrefix "gax-dels" ;
    rdfs:seeAlso                   <http://w3id.org/gaia-x/core#> ;
    rdfs:seeAlso                                    <https://partnerspace.atlas-
sian.net/wiki/spaces/GXFS/pages/edit-v2/1716093513> ;
    void:vocabulary                vann:, void:, voaf:, dct:, foaf: ;
.

<https://github.com/jlangkau>
    a         dct:Agent, foaf:Person ;
    foaf:name "Jörg Langkau" ;
.

gax-dels:Notification
    a                 owl:Class ;
    rdfs:label        "Notification"@en ;
    dct:description   "Abstract class of all Data Exchange Logging Service No-
tifications."@en ;
    rdfs:subClassOf   ldp:Resource ;
    rdfs:isDefinedBy gax-dels: ;
.

gax-dels:DataExchangeNotification
    a                 owl:Class ;
    rdfs:label        "Data Exchange Notification"@en ;
    dct:description   "Abstract class of all Data Exchange Logging Service Data
Exchange Notifications."@en ;
    rdfs:subClassOf   gax-dels:Notification ;
    rdfs:isDefinedBy gax-dels: ;
.

gax-dels:SendDataNotification
    a                 owl:Class ;
    rdfs:label        "Send Data Notification"@en ;
    dct:description
```

```
                    "Class of Notification send by given provider to figure
out 'data will be send immediatly to given Data Consumer'."@en ;
    rdfs:subClassOf  gax-dels:DataExchangeNotification ;
    rdfs:isDefinedBy gax-dels: ;
.

gax-dels:DataReceivedNotification
    a                owl:Class ;
    rdfs:label       "Data Received Notification"@en ;
    dct:description
                    "Class of Notification send by given consumer to figure
out 'data was received from given Data Provider'."@en ;
    rdfs:subClassOf  gax-dels:DataExchangeNotification ;
    rdfs:isDefinedBy gax-dels: ;
.

gax-dels:issued
    a                  owl:DatatypeProperty ;
    rdfs:subPropertyOf dct:issued ;
    rdfs:label         "issued"@en ;
    dct:description
                    "Date and time given Notification was issued to inbox of
given Data Exchange Logging Service."@en ;
    rdfs:domain        gax-dels:Notification ;
    rdfs:range         xsd:dateTimeStamp ;
    rdfs:isDefinedBy   gax-dels: ;
.

gax-dels:dataAsset
    a                owl:ObjectProperty ;
    rdfs:label       "Data Asset"@en ;
    dct:description  "The main piece of data this Notification is about."@en ;
    rdfs:domain      gax-dels:Notification ;
    rdfs:range       gax:DataAsset ;
    rdfs:isDefinedBy gax-dels: ;
.

gax-dels:sender
    a                owl:ObjectProperty ;
    rdfs:label       "Sender"@en ;
    dct:description
                    "Sender of given Notification. Also acts as given Gaia-X
Service Instance (GX-SI), called 'Data Provider'."@en ;
    rdfs:domain      gax-dels:Notification ;
    rdfs:range       gax:ServiceInstance ;
    rdfs:comment     "'sender' is also sender of given Data Asset as shown in
'gax-dels:dataAsset'."@en ;
    rdfs:isDefinedBy gax-dels: ;
.

gax-dels:dataContract
    a                owl:ObjectProperty ;
    rdfs:label       "Data Contract"@en ;
    dct:description   "Contract to given Data Asset as subject of data-
excahnge."@en ;
    rdfs:domain      gax-dels:Notification ;
    rdfs:range       gax-dcs:DataContract ;
```

```
    rdfs:isDefinedBy gax-dels: ;
.

gax-dels:receiver
    a                owl:ObjectProperty ;
    rdfs:label       "Receiver"@en ;
    dct:description
                     "Receiver of given Data Asset. Also acts as given Gaia-X
Service Instance (GX-SI), called 'Data Consumer'."@en ;
    rdfs:domain      gax-dels:Notification ;
    rdfs:range       gax:ServiceInstance ;
    rdfs:isDefinedBy gax-dels: ;
.


gax-dels:logToken
    a                owl:ObjectProperty ;
    rdfs:label       "Log Token"@en ;
    dct:description  "Log Token description"@en ;
    rdfs:domain      gax-dels:Notification ;
    rdfs:range       xsd:string ;
    rdfs:isDefinedBy gax-dels: ;
.

## EOF
```

Code: "Gaia-X Data Exchange Logging Service (GX-DELS) Ontology."

Appendix D: Overview GXFS Work Packages

The project "Gaia-X Federation Services" (GXFS) is an initiative funded by the German Federal Ministry of Economic Affairs and Energy (BMWi) to develop the first set of Gaia-X Federation Services, which form the technical basis for the operational implementation of Gaia-X.

The project is structured in five Working Groups, focusing on different functional areas as follows:

Work Package 1 (WP1): Identity & Trust
Identity &Trust covers authentication and authorization, credential management, decentral Identity management as well as the verification of analogue credentials.

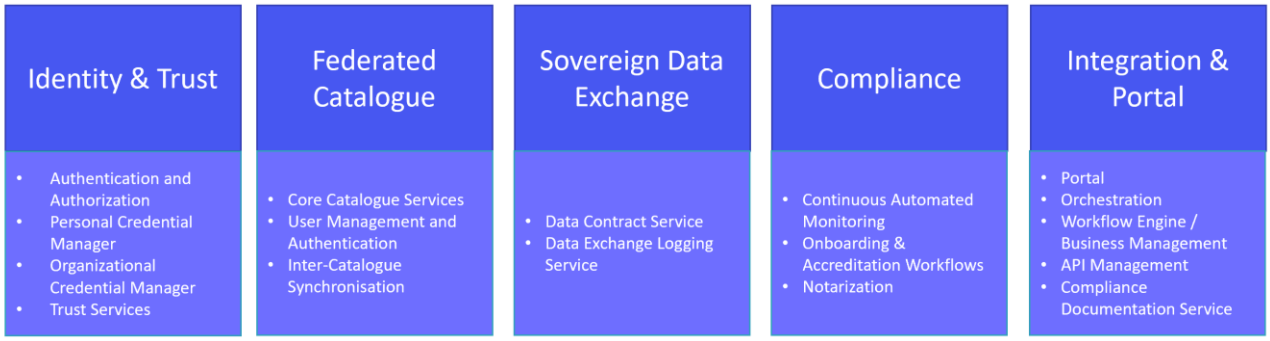
Work Package 2 (WP2): Federated Catalogue
The Federated Catalogue constitutes the central repository for Gaia-X Self-Descriptions to enable the discovery and selection of Providers and their Service Offerings. The Self-Description as expression of properties and Claims of Participants and Assets represents a key element for transparency and trust in Gaia-X.

Work Package 3 (WP3): Sovereign Data Exchange
Data Sovereignty Services enable the sovereign data exchange of Participants by providing a Data Agreement Service and a Data Logging Service to enable the enforcement of Policies. Further, usage constraints for data exchange can be expressed by Provider Policies as part of the Self-Description

Work Package 4 (WP4): Compliance
Compliance includes mechanisms to ensure a Participant's adherence to the Policy Rules in areas such as security, privacy transparency and interoperability during onboarding and service delivery.

Work Package 5 (WP5): Portal & Integration
Gaia-X Portals and API will support onboarding and Accreditation of Participants, demonstrate service discovery, orchestration and provisioning of sample services.

All together the deliverables of the first GXFS project phase are specifications for 17 lots, that are being awarded in EU-wide tenders:



Further general information on the Federation Services can be found in [GX-TAD].

Appendix E: ADR-XXX

ADR-XXX: GAIA-X Identifier Format
==========
:adr-id: XXX
:revnumber: 2.1
:revdate: 2021-03-19
:status: proposed
:author: GAIA-X Catalogue and IAM Community
:stakeholder: IAM WG, Self-Description WG, Catalogue WG
Summary

ADR-001 defines the use of JSON-LD for the Self-Descriptions. JSON-LD requires that Identifiers used for cross-referencing between self-descriptions are IRIs (Internationalized Resource Identifiers [RFC3987]). This ADR upholds this definition and further refines it.
Identifiers used in GAIA-X shall be URIs following the [RFC3986].
Identifier must re-use existing schemas or define their own private URI schema(s) eu.gaia-x. where necessary [BCP35]. The schema shall define additional semantics to indicate the underlying protocol.

Context

The generic structure of the identifier takes the form:
:
For protocols requiring a new URI schema a private schema should be defined following the pattern:
eu.gaia-x.

The following Identifier schemas have a defined mechanism to ensure uniqueness of the Identifier. More schemas may be added in the future.
Protocol Schema Protocol_Specific_Id

TAG [RFC4151] urn:tag ,:
OpenID Connect eu.gaia-x.openid ;
DID did :

Tag URI Schema
~~~~~~~~~~~~~~~
Identifiers used in Self-Descriptions may follow the conventions of RFC 4151 for the 'tag' URI scheme. Identifiers of this format contain the DNS domain name or an email of the issuing organization as well as a date at which the organization was in possession of the DNS domain. That way, the organization in possession of the DNS domain at that time is responsible to issue only unique Identifiers.

Some examples of Identifiers:
urn:tag:provider-name.com,2020:my-service:v1
urn:tag:subdomain.foobar.com,2020-01:org1/data-asset5/element20
urn:tag:foobar@acme.org,2020-01-29:e51a9f18273718445f0c016f23b2bc05919f7433

By the convention that only the organization owning the domain-name may use it for Identifiers, GAIA-X Participants can themselves issue new Identifiers and ensure that Identifiers are unique without the need for a central identifier registry for all GAIA-X Participants.

OpenID Connect URI Schema
~~~~~~~~~~~~~~~~~~~~~~~~~~
OpenID Connect eu.gaia-x.openid ;
Example:
eu.gaia-x.openid:https://example-idp.org/auth/realms/master;YWxpY2VAZm9vLmNvbQ
Companies have to host an endpoint that is part of the Identifier.
To ensure uniqueness, endpoints might need to change after a domain changes ownership and uniqueness of identifiers cannot be otherwise guaranteed.

DID URI Schema
~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~

DID identifiers according to W3C "Decentralized Identifiers", Candidate Recommendation: https://www.w3.org/TR/did-core refer to a "method".
1. The method refers to a "Verifiable Data Registry" where the DID can be resolved to a document.
2. The Verifiable Data Registry ensures uniqueness of the Identifiers.
Examples for such Verifiable Data registries include "distributed ledgers, decentralized file systems, databases of any kind, peer-to-peer networks, and other forms of trusted data storage" as described in https://www.w3.org/TR/did-core/#architecture-overview

Decision Statements

GAIA-X Identifiers uniquely identify an entity in GAIA-X.
Informational: Entities can refer to (non-exhaustive)
- Entities with a Self-Description (contains Participant)
- Principals (user accounts)
- Abstract Concepts (for example "European Economic Area" or "ISO 27001").

GAIA-X Identifiers are unique in the sense that an Identifier must never refer to more than one entity. There can be several GAIA-X Identifiers refering to the same entity.

Informational: As a policy, multiple Identifiers for the same entity should be avoided in the Catalogue.
All Identifiers used in GAIA-X are URIs following the [RFC3986] specification.

Informational: JSON-LD allows IRIs. URIs are a strict subset of that.
The lifetime of an Identifier is permanent. That is, the Identifier has to be unique forever, and may be used as a reference to an entity well beyond the lifetime of the entity it identifies or of any naming authority involved in the assignment of its name [RFC1737]. Reuse of an Identifier for a different entity, also at a later time, is forbidden.
There are multiple valid URI schemas defined, each associated with a technical mechanism to ensure uniquenes. The structure of an identifier has to ensure the uniqueness of the Identifier.

Informational:
GAIA-X Participants can self-issue Identifiers. It is solely the responsibility of a Participant to determine the conditions under which the Identifier will be issued. A self-issued Identifier can be used without publicly registering or announcing the Identifier first.
Not all URI schemas are usable for self-issuing.

Informational:
Identifiers shall be derived from the native identifiers of an Identity System without any separate attribute needed. The Identifier shall provide a clear

reference to the Identity System technology used. OpenID Connect and DID shall be supported. Any scheme for Identifiers must permit future extensions to the scheme.

Informational:
The Identifier shall be comparable in the raw form. It shall not be needed to make any transformation to compare two identifiers and tell whether they are the same.

Informational:
Identifiers should not contain more information than necessary (including Personal Identifiable Information).
Consequences

GAIA-X Participants Identity Systems can self-issue valid Principal Identifiers.
Based on the identifier it is possible to determine the technology and the unique reference to the Identity.

ADR References
---
* ADR-001

External References
---
* [BCP35] Guidelines and Registration Procedures for URI Schemes. https://tools.ietf.org/html/bcp35
* [DID-Core] Decentralized Identifiers (DIDs) v1.0. https://www.w3.org/TR/did-core/
* [IAM Framework] GAIA-X IAM Framework v1.01. https://docs.google.com/document/d/1XCjIVRul_w_6runDn_Rh-8nVdMhSFmMxZTXoQAhtISA/
* [RFC1737] Functional Requirements for Uniform Resource Names. https://tools.ietf.org/html/rfc1737
* [RFC3986] Uniform Resource Identifier (URI): Generic Syntax. https://tools.ietf.org/html/rfc3986