



iSHARE

Version 1.10

Exported on 06/24/2019

Table of Contents

Introduction	5
Reader's guide.....	5
Goals and scope of the iSHARE Scheme	6
Guiding principles	7
Governance framework	11
Releases.....	13
Release notes	14
Release planning.....	19
Version history.....	20
Main scheme aspects.....	21
Key functionality	22
Technical overview	30
Framework and roles.....	31
Legal provisions	37
Operational provisions	38
Use cases	39
Use case: M2M interaction (with fine-grained authorization)	40
Use case: H2M interaction (with coarse-grained authorization).....	43
Use case: portable identity.....	47
Use case: delegation (and management of consent).....	52
Detailed descriptions.....	59
Functional.....	60
Technical	106
Operational	144
Legal	169
Glossary and legal notices.....	175
Glossary	176
Legal notices	187

Project history	188
Background information	189
Assumptions.....	190



iSHARE

This document provides a full overview of the current state of the iSHARE scheme (v1.10). iSHARE is a collaborative effort to improve conditions for data-sharing for organisations involved in the logistics sector. The functional scope of the iSHARE Scheme focuses on topics of identification, authentication and authorisation.

Introduction

The purpose of this document is to provide a complete overview of the current state of the iSHARE Scheme (v1.10).

iSHARE is a collaborative effort to improve conditions for data-sharing for organisations involved in the logistics sector. The functional scope of the iSHARE Scheme focuses on topics of identification, authentication and authorisation.

As of 2018, the iSHARE Scheme is publicly available to the market.

Reader's guide

- iSHARE's introductory section describes the scheme's starting points: its goals, the guiding principles and the iSHARE governance framework;
- The '[releases](#)' section describes the release notes, planning of future releases and version history of the iSHARE Scheme;
- The '[main scheme aspects](#)' section summarises the most important functionality of the iSHARE scheme, its framework and roles, and the technical, operational and legal provisions enabling it;
- The '[use cases](#)' section showcases the scheme's key functionalities in four use cases;
- The '[detailed descriptions](#)' section explains the in-depth Functional, Technical, Legal and Operational agreements that, together, improve data-sharing conditions for the logistics sector;
- The scheme concludes with the '[glossary and legal notices](#)' section;
- The project history provides some '[background information](#)' about the project and assumptions on the basis of which the scheme was co-created.

i Within the iSHARE Scheme documentation, the following notational conventions apply:

- The keywords 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'MAY', and 'OPTIONAL' in this document are to be interpreted as described in IETF [RFC 2119](#) whenever this note is at the top of the chapter:
 - *This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.*

Goals and scope of the iSHARE Scheme

The iSHARE Scheme is a collaborative effort to improve the exchange of data between organisations involved with the Dutch logistics sector. The iSHARE Scheme results in a set of agreements which improve circumstances for data exchange.

The ambition of iSHARE is to lower barriers for sharing data, to empower new forms of collaboration in chains and to help scale up existing initiatives that aim to improve conditions for data exchange. The underlying assumption is that if data can flow in a controlled and smart way, it will lead to a more efficient use of infrastructure, less carbon emissions and a more competitive logistics sector.

The iSHARE Scheme's scope focuses on three main topics that are crucial in any data exchange context:

1. Identification;
2. Authentication;
3. Authorisation.

iSHARE focuses on these three aspects as they are considered indispensable in any communication between parties, also in the context of exchanging logistical data. Within the iSHARE Scheme, agreements are made on the above three topics with the aim of to work towards a more uniform, straightforward and controlled way of exchanging data on a bigger scale than is possible right now*.

- Uniform: one uniform way of working across all types of modalities, small and large organisations, public and private organisations, suppliers and receivers of data or their softwarepartners, etc. iSHARE aims to create new possibilities for efficiency improvements, time gains and cost savings.
- Straightforward: Easy to connect with new, existing and third-party business partners throughout the sector, more certainty on trustworthiness of parties you exchange data with, a building block which is easy to implement by your software partners or your IT department and an addition that empowers your existing solutions.
- Controlled: The basic principle within iSHARE is that the owner of the data stays in control at all times; the owner decides with whom what data is exchanged on what terms.

These three aims can only be reached when a variety of perspectives are considered during the establishment of the scheme. To this end, a variety of organisations are involved in defining the agreements for iSHARE. During the co-creation phase of the iSHARE project, the involved organisations invest in the iSHARE Scheme in terms of expertise.

*Notes:

- The scope of the iSHARE Scheme does not include the specification of possible business models for sharing data and/or payments related to data exchange;
- The iSHARE Scheme can in some way be compared with the institute of the passport: the iSHARE Scheme will be usable by anyone who owns a digital identity compatible with the iSHARE Scheme. This will greatly simplify authentication and authorisation processes, also between different organisations (however: even though organisations can have valid certificates, it does not rule out possible malign intentions).

Guiding principles

To achieve the goals of the iSHARE Scheme, it is paramount to stay close to a set of guiding principles. As time progresses new principles can be defined, existing principles can be adapted or dropped if deemed necessary. The guiding principles were defined using the format as suggested* by [TOGAF 8.1.1 architectural principles](#).

The following principles define the iSHARE Scheme and must be kept in mind at all times during further development (see details of guiding principles below):

Principle #	Principle name
1	Generic building block to enable data exchange
2	Limited scope: identification, authentication, and authorisation
3	Leverage existing (international) building blocks
4	Agnostic towards nature and content of data
5	Benefits outweigh investment for all types of participants
6	International orientation

Guiding principles details:

Principle 1	Generic building block to enable data exchange
Statement	iSHARE is a generic identification, authentication and authorisation scheme to be used as enabler for data exchange in logistics
Rationale	In every exchange of data, identification, authentication and authorisation are fundamental factors. iSHARE aims to simplify processes of identification, authentication and authorisation as a generic solution to facilitate data exchange in the logistics sector.
Implications	<ul style="list-style-type: none"> The iSHARE Scheme will allow for extension or adaptability so it can be used in situation/sector specific cases; The iSHARE Scheme will not cater to a specific sector or market, it is applicable in an N amount of cases; The iSHARE Scheme will not be a point solution.

Principle 2	Limited scope: identification, authentication, and authorisation
Statement	The iSHARE Scheme's scope is limited to topics of identification, authentication and authorisation in the context of data exchange

Principle 2	Limited scope: identification, authentication, and authorisation
Rationale	iSHARE aims to improve the circumstances for data exchange throughout the logistics sector and provides focus on the topic of identification, authentication and authorisation. Identification, authentication and authorisation are a fundamental part of any data exchange, but are not solved in a scalable or standardised way at the moment.
Implications	<ul style="list-style-type: none"> Without this principle, there is a risk of 'scope creep': related topics could take away the focus off the intended topics

Principle 3	Leverage existing (international) building blocks
Statement	Where possible, iSHARE should be realised using existing and proven standards, technology or initiatives
Rationale	By reusing building blocks already available and in use, the impact on organisations to participate in iSHARE and the time to realise the iSHARE Scheme are lowered. Standards, technology and initiatives preferably have a broad (international) usage base and are backed by a professional organisation charged with maintenance of the standards, technology or initiatives.
Implications	<ul style="list-style-type: none"> the iSHARE Scheme will build on or use existing (international) standards, technology or initiatives where possible; the iSHARE Scheme will aim to use open standards, technology or initiatives; the iSHARE Scheme may use proprietary standards, technology or initiatives; if existing and/or proven standards, technology or initiatives do not provide what is needed, alternative solutions will be sought.

Principle 4	Agnostic towards nature and content of data
Statement	The iSHARE Scheme does not concern itself with the contents or nature of data
Rationale	Given the generic nature of the iSHARE Scheme and the aim to be applicable throughout the logistics sector, iSHARE needs to function with any type of possible data and/or any relevant data exchange interaction model. To this end, the contents of data are only considered where it concerns the facilities needed within iSHARE to adequately exchange various types of data (e.g. requirements to security, encryption, etc.). It is up to the participating organisations to ensure that iSHARE adequately fulfills requirements to the process of identification, authentication and authorisation in the context of data exchange.
Implications	<ul style="list-style-type: none"> the iSHARE Scheme will not specify the (allowed) content of data exchanges done within an iSHARE context; the iSHARE Scheme does not specify content specific data standards; the iSHARE Scheme should not have limitations connected to types of data or standards used.

Principle 5	Benefits outweigh investment for all types of participants
Statement	The iSHARE Scheme needs to be attractive to use and implement for all types of participants/roles.
Rationale	The iSHARE Scheme knows different roles with different responsibilities. When a potential participant considers taking a (or multiple) role(s) in the iSHARE Scheme, the iSHARE Scheme should aim to have the lowest possible threshold to participate for the potential participant. Depending on what the character of the potential participant is (e.g smaller size or larger size organisations) and which role the participant wants to take, this could mean that the impact of implementation needs to be small or that the implementation is kept relatively simple.
Implications	<ul style="list-style-type: none"> the iSHARE Scheme aims to keep thresholds to participate in the iSHARE Scheme (e.g. in terms of implementation impact or onboarding/certification effort) as low as possible for all possible roles; the iSHARE Scheme strives for the lowest possible impact for participants when changes occur in the future. Changes to used standards will take place; within the iSHARE Scheme and its specifications thought needs to be given to how change is dealt with in an efficient way.

Principle 6	International orientation
Statement	The iSHARE Scheme needs to look over geographic boundaries to foster international involvement and cooperation
Rationale	The logistics sector is per definition an international sector. The iSHARE Scheme needs to facilitate, to the extent that it is practical and possible, international involvement.
Implications	<ul style="list-style-type: none"> the iSHARE Scheme needs its participants to provide knowledge and experience on how the iSHARE Scheme can stay (and become) attractive in the international context

*Format used for defining guiding principles, based on TOGAF standard:

Principle name	Should both represent the essence of the rule as well as be easy to remember. Specific technology platforms should not be mentioned in the name or statement of a principle. Avoid ambiguous words in the Name and in the Statement such as: 'support', 'open', 'consider', and for lack of good measure the word 'avoid', itself, be careful with 'manage(ment)', and look for unnecessary adjectives and adverbs (fluff).
Statement	Should succinctly and unambiguously communicate the fundamental rule. For the most part, the principles statements for managing information are similar from one organisation to the next. It is vital that the principles statement be unambiguous.

Rationale	Should highlight the business benefits of adhering to the principle, using business terminology. Point to the similarity of information and technology principles to the principles governing business operations. Also describe the relationship to other principles, and the intentions regarding a balanced interpretation. Describe situations where one principle would be given precedence or carry more weight than another for making a decision.
Implications	Should highlight the requirements, both for the business and IT, for carrying out the principle - in terms of resources, costs, and activities/tasks. It will often be apparent that current systems, standards, or practices would be incongruent with the principle upon adoption. The impact to the business and consequences of adopting a principle should be clearly stated. The reader should readily discern the answer to: 'How does this affect me?' It is important not to oversimplify, trivialise, or judge the merit of the impact. Some of the implications will be identified as potential impacts only, and may be speculative rather than fully analysed.

Governance framework

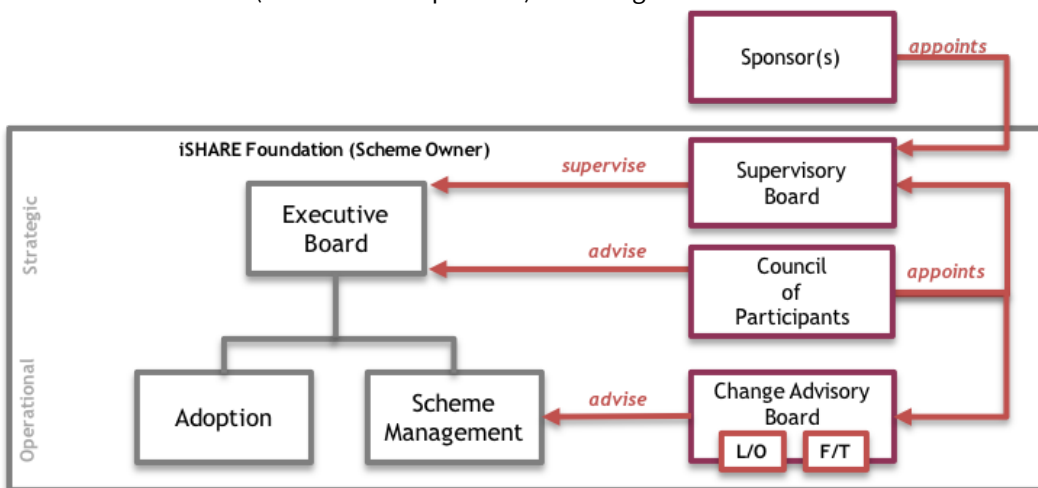
This page describes the governance framework of the iSHARE Scheme. The “iSHARE Foundation” has been established on the first of November, 2018. This marked the start of a transitional period in which both the iSHARE Foundation and the iSHARE Project co-exist.

The iSHARE governance framework consists of the following bodies. See below for detailed descriptions:

- the iSHARE Foundation, which is the Scheme Owner of iSHARE
- the Executive Board
- the Supervisory Board
- the Council of Participants
- the Change Advisory Board
- Sponsors of the iSHARE project

The governance of the iSHARE Scheme is organised as such that the iSHARE network can operate and grow in a sustainable way. At the same time its governance provides the appropriate checks and balances that will allow iSHARE Participants to provide input, supervise ongoing activities and collaboratively influence the growth and development of the iSHARE scheme.

Rules about the foundation’s organisation and its governance framework are captured in the statutes of the iSHARE Foundation. The iSHARE Foundation is listed in the Commercial Register (Handelsregister) maintained by the Chamber of Commerce (Kamer van Koophandel) under registration number 73058289.



Scheme Owner

The iSHARE Foundation is the Scheme Owner of the iSHARE Scheme and is responsible for all activities related to the iSHARE Scheme. The iSHARE Scheme Owner consists of:

- An **Executive Board** formed by (an) independent representative(s) of the iSHARE community. Executive board members are selected and chosen by the Supervisory Board. The Executive Board is the highest organ of the Scheme Owner. Executive board members are accountable to the Supervisory Board for the functioning of the iSHARE Foundation and the iSHARE Scheme.
- An **operational branch** responsible for day to day scheme management activities. These activities include (amongst others) the following responsibilities:
 - Management of the iSHARE network (participants);
 - Management of the iSHARE Scheme (specifications + brand management);

- Development and maintenance of tools.

The Scheme Owner's operational branch will be fulfilled by the iSHARE project organisation for the duration of the iSHARE Project. As such, during the transition period the operational activities of the iSHARE Foundation will be governed by both the Foundation itself as by the iSHARE Project Steering Committee. The operational activities will initially focus on carrying out legal matters, such as concluding accession agreements with Participants. After termination of the iSHARE Project all scheme management activities will be transferred to the iSHARE Foundation and governed solely and independently by the iSHARE Foundation and its governing bodies.

Sponsor(s)

The iSHARE Foundation can receive funding from government and semi-public organisations that wish to support the realisation of the goals of the Foundation. Upon request of such an organisation, the Supervisory Board can decide to grant the organisation the status of Sponsor of the iSHARE Foundation. The organisation maintains the status of Sponsor for the duration that it provides funding to the iSHARE Foundation. At the start of the iSHARE Foundation, [Stichting Connekt](#) is recognised as Sponsor, providing funding on behalf of [NLIP](#) and [Topsector Logistiek](#).

Supervisory Board

- The Supervisory Board is appointed by the Council of Participants and consists of three members
 - Until the end of 2020, two members of the Supervisory Board may be appointed by Sponsor(s), rather than the Council of Participants
 - From 2021, one member of the Supervisory Board may be appointed by the Sponsor(s), rather than the Council of Participants, if there are Sponsor(s) present
- The Supervisory Board supervises the correct functioning of the iSHARE Foundation's Executive Board and elects/dismisses the members of the Executive Board.

The Supervisory Board and the iSHARE Project Steering Committee can together decide on gradually transferring Scheme Owner activities from the iSHARE Project Organisation to the iSHARE Foundation.

Council of Participants

- The Council of Participants consists of all Parties (or representations of these parties) that have a iSHARE Scheme contract with the iSHARE Foundation and are willing to participate in the Council of Participants' activities.
- The Council of Participants advises the Board of the iSHARE Foundation and appoints members of the Supervisory Board.

Change Advisory Board

- The Change Advisory Board consists of subject matter experts (legal/ operational/ functional/ technical) delegated by the Participants.
- The Change Advisory Board advises the Scheme Owner on changes to the specifications of the iSHARE Scheme.

Releases

This chapter describes the release notes, planning of future releases and version history of the iSHARE Scheme.

- [Release notes](#)
- [Release planning](#)
- [Version history](#)

Release notes

The release notes show the release history and the main differences between releases.

Release 1.10 (Current)	
Purpose	<ul style="list-style-type: none"> • Updated the process for verifying eIDAS certificates • Update the admission process for Certified Parties • Added levels of assurance for Certified Parties • Added support for the usecase where the Entitled Party is not an iSHARE Adhering Party, but is a customer of an iSHARE Service Provider • Small technical improvements
Release date	24 June 2019
Change log	<p>Functional</p> <ul style="list-style-type: none"> • Updated the functional requirements for Certified Parties (references to eHerkenning have been removed and replaced by relevant requirements) <p>Technical</p> <ul style="list-style-type: none"> • Updated the process for verifying eIDAS certificates (now uses the same process as PKI-Overheid certificates) • Added level of assurance to the Party Info endpoint at the Scheme Owner (for Certified Parties) • Clarified the use of acr_values in the H2M flow to request a minimal level of assurance • Clarified the headers used for JWE in the H2M flow <p>Operational</p> <ul style="list-style-type: none"> • Updated the admission process for Certified Parties, including the following changes: <ul style="list-style-type: none"> • Admission to eHerkenning is no longer required • Added the concept of Levels of Assurance for Certified Parties • Added the Assessment Framework for Certified Parties to determine the Level of Assurance of a Certified Party <p>Legal</p> <ul style="list-style-type: none"> • Updated the terms of use to reflect the usecase where the Entitled Party is not an iSHARE Adhering Party, but is a customer of an iSHARE Service Provider

Release 1.9	
Purpose	Enable Human to Machine (H2M) authorisations

Release 1.9	
Release date	5 April 2019
Change log	<p>Functional:</p> <ul style="list-style-type: none"> Updated primary Human to Machine use cases to reflect changes from RFC 010, which added the authorization flow to these cases <p>Technical:</p> <ul style="list-style-type: none"> Added technical specifications on the generic authorization flow for Human to Machine (H2M) interactions as facilitated by Identity Providers in the iSHARE Scheme, to reflect changes from RFC 010 Modified technical specifications for capabilities endpoint of all iSHARE Parties <p>Operational</p> <ul style="list-style-type: none"> No changes <p>Legal</p> <ul style="list-style-type: none"> Updated governance framework to reflect that Stichting iSHARE Foundation has been created <p>Miscellaneous</p> <ul style="list-style-type: none"> Rearranged pages and sections to improve readability Material has been moved from the Scheme to the Developer Portal, to improve readability and usability of both
Release 1.8	
Purpose	Enable Human to Machine (H2M) interactions and finalise contracts for signing between iSHARE Foundation and iSHARE Participants.
Release date	31 October 2018

Release 1.8	
Change log	<p>Functional:</p> <p>-</p> <p>Technical:</p> <ul style="list-style-type: none"> • Added technical specifications on the generic authentication flow for Human to Machine (H2M) interactions as facilitated by Identity Providers in the iSHARE Scheme. <p>Operational</p> <p>-</p> <p>Legal</p> <ul style="list-style-type: none"> • Minor modifications to Terms of Use and the Accession Agreements for Adhering Parties and Certified Parties • Moved GDPR Factsheet and templates for Data Exchange Agreement and Data Processor Agreements to appendix of the scheme, since they serve merely as an inspiration for participants who want to make additional bilateral arrangements with others they are sharing data with. <p>Miscellaneous</p> <ul style="list-style-type: none"> • The governance framework is updated • Rearranged pages and sections to improve readability

Release 1.7	
Purpose	Create a better overview of all API specs in a singular space, to make it easier for developers to implement iSHARE and to improve readability of the iSHARE Scheme
Release date	28 June 2018

Release 1.7	
Change log	<p>Functional:</p> <p>-</p> <p>Technical:</p> <ul style="list-style-type: none"> • The API technical specifications are moved to a dedicated developer portal. • Adjusted specification for JSON Web Token (JWT) to facilitate certificate validation under eIDAS. <p>Operational</p> <ul style="list-style-type: none"> • The service levels are now structured per participant type (Adhering Party/ Certified Party/ Scheme Owner) to give participants a better overview of their applicable service levels. <p>Legal</p> <p>-</p> <p>Miscellaneous</p> <ul style="list-style-type: none"> • Rearranged pages and sections to improve readability • The governance framework is updated • The project history is updated

Release 1.6	
Purpose	Lowering barriers for parties to start using iSHARE
Release date	11 May 2018
Change log	<p>Functional:</p> <p>-</p> <p>Technical:</p> <ul style="list-style-type: none"> • For authentication purposes the use of digital certificates within iSHARE will be limited initially to certificates issued under PKIOverheid. <p>Operational</p> <ul style="list-style-type: none"> • The admission process for Certified Parties and Adhering Parties are merged to one generic process. Role-specific requirements may apply. • The order of admission steps is changed to enable new iSHARE entrants to start testing before a contract is signed. <p>Legal</p> <p>-</p> <p>Miscellaneous</p> <ul style="list-style-type: none"> • Rearranged pages and sections to improve readability • The governance framework is updated

Release 1.5	
Purpose	First public version the iSHARE Scheme that can be used by launching customers
Release date	14 December 2017
Change log	<ul style="list-style-type: none"> • Updated specifications for all content of the iSHARE Scheme: <ul style="list-style-type: none"> • Functional • Technical • Operational • Legal • Significantly updated and rearranged sections for readability, including a main scheme aspects- and illustrative use cases chapter with new depictions; Integrated (technical) specifications, generic and per iSHARE role

Release planning

The release planning provides detailed information about changes that are planned for future releases of the scheme. See the [Operational Process Release Management](#) section for details about the release management process of the iSHARE Scheme.

Planned releases

There are not current releases planned.

The RFCs for a next release will be discussed in the CAB meetings, which can be found on the [agenda](#). If you want to participate, please contact us on info@ishareworks.org.

Suggestions

If you have any other suggestions to improve the iSHARE Scheme, please let us know. [Download the RFC request form](#) and send this form filled-in to info@ishareworks.org.

Version history

- [iSHARE v1.9](#), 5 April 2019
- [iSHARE v1.8](#), 31 October 2018
- [iSHARE v1.7](#), 28 June 2018
- [iSHARE v1.6](#), 11 May 2018
- [iSHARE v1.5](#), 14 December 2017
- [iSHARE v1.2](#), 25 October 2017
- [iSHARE v1.0](#), 23 June 2017
- [iSHARE v0.5](#), 24 March 2017
- [iSHARE v0.3](#), 27 February 2017
- [iSHARE v0.2](#), 13 February 2017
- [iSHARE v0.1](#) (start document)

Main scheme aspects

The iSHARE Scheme is a combination of Functional, Technical, Operational and Legal agreements to which participating parties adhere. This chapter provides a bird's eye view on the main aspects of iSHARE, and an introduction to more in depth details of the scheme.

This section describes the iSHARE Scheme's:

- Key functionality
 - Support Machine to Machine (M2M) interaction
 - Support Human to Machine (H2M) interaction
 - Facilitate portable identity(s) for parties and humans
 - Facilitate flexible authorizations, applicable in any context
 - Enable data exchange based on delegations - even between unknown parties
 - Enable control over own data through management of consent
 - Provide a trust framework
- Technical overview
- Framework and roles
- Legal provisions
- Operational provisions

Key functionality

The iSHARE Scheme aims to support the following key functionalities:

- Support Machine to Machine (M2M) interaction
- Support Human to Machine (H2M) interaction
- Facilitate portable identity(s) for parties and humans
- Facilitate flexible authorizations, applicable in any context
- Enable data exchange based on delegations - even between unknown parties
- Enable control over own data through management of consent
- Provide a trust framework

In line with iSHARE's [guiding principles](#), these key functionalities might be realised by (re)using existing standards or initiatives.

Support Machine to Machine (M2M) interaction

The iSHARE Scheme aims to support multiple interaction models, of which Machine to Machine (M2M) is one. M2M interaction can be characterised as communication between machines, without interference by a human. In contemporary data communication there is a heavy reliance on M2M interaction.

Example:

- Every day, the ERP system (machine) of party A requests a status update from the ERP system (machine) of party B. Party B's ERP system automatically responds with the requested status update. No humans are needed to interfere.

This example is detailed under [use cases](#).

The opposite of the M2M interaction model is the [Human to Machine interaction model](#).

Support Human to Machine (H2M) interaction

The iSHARE Scheme aims to support multiple interaction models, of which Human to Machine (H2M) is one. H2M interaction can be characterised as communication between a human and (a) machine(s). A user interface is necessary to enable H2M communication.

Example:

- Human X, working for Party A, requests a status update from the ERP system (machine) of Party B. It does so via a user interface.

This example is detailed under [use cases](#).

The opposite of the H2M interaction model is the [Machine to Machine interaction model](#).

Facilitate portable identity(s) for parties and humans

iSHARE aims to facilitate (but not impose) the use of one or more so called 'federated identity(s)'. A federated identity is an identity that is spread out and recognised, i.e. portable, across multiple, independent systems.

Within iSHARE, the use of federated identities would reduce costs by eliminating the need for proprietary, or newly issued identity solutions. In order for an identity to become part of iSHARE's federation, the legal entity providing the identity must be certified under the iSHARE Scheme.

Example:

- Human X, working for Party A, has a personal keycard issued by iSHARE certified Identity Provider Y. The card, and thus the identity of Human X, can be used to identify and authenticate Human X at party B.

This example is detailed under [use cases](#).

Facilitate flexible authorizations, applicable in any context

iSHARE aims to enable parties to grant other parties or persons access to (parts of) their data or services. Parties within the iSHARE Scheme have greatly varying backgrounds, however. Private and public, large and small, different value chains, different geographies, different modalities, etc. For that reason, iSHARE needs to have a flexible way of expressing authorizations.

Two examples can illustrate different levels of required flexibility:

1. Some parties or contexts require management of authorizations on a very detailed level, e.g. Party A's ERP system (machine) is ONLY allowed to request status updates concerning line X of bill of lading Y;
2. Some contexts require less detailed authorizations, e.g. Party A's ERP system (machine) is allowed to request ANY information about ANY (part of a) bill of lading.

Both examples are explained under use cases: [fine-grained](#); [coarse-grained](#).

The iSHARE Scheme envisions a world in which (access) authorizations are flexible in three ways:

- Flexible authorization scope;
iSHARE aims to provide a way to add a layer of authorization to any resource or any selection or combination of resources. The authorization scope refers to the objects or resources of a specific party, to which authorizations need to be assigned. The scope can include many or all resources (e.g. all data), or only some resources (e.g. specific data fields or services). Either way, the scope is always governed by a formal agreement and implemented by technical means.
- Granular authorizations, and;
iSHARE aims to provide a granular way to use authorizations for resources. The authorization granularity refers to the characteristics of both the requested resources and the rules (policies, conditions) that apply. Authorizations to resources can be coarse-grained (e.g. someone has access to all data in a certain data scope) or fine-grained (e.g. someone has access to only data with a low sensitivity level). The rules (policies, conditions) that control the authorizations can be fine-grained as well, meaning that many different types of rules can apply, such as time of day, location, organisation, role, and competence level.
- Flexible authorization source.
iSHARE aims to provide flexibility to where authorization rules are stored and can be retrieved. The authorization source refers to the location of the rules (policies, conditions) and the attributes (e.g. subject attributes, object attributes) that govern the authorizations. These can be located near the data, at a dedicated source, or a combination thereof. In the current version of the iSHARE Scheme, the flexibility in authorization source is described as 'Policy Information Point' or PIP in the [detailed functional descriptions](#).

Enable data exchange based on delegations - even between unknown parties

One of the barriers to exchanging data is often that parties do not know each other sufficiently, and therefore are not able to share data. Often this can only be done after some form of contract has been established.

Within iSHARE it is the explicit aim to make it possible to exchange data for parties that are unknown to each other based on delegations. A delegation within iSHARE functions as evidence that a party is directly or indirectly operating in name of a known party. Based on the delegation a certain (unknown) party has given, a party can decide if this party may receive certain data or not.

Example:

- Party A hires Trucking Company B to deliver Container X to Party C. Trucking Company B's ERP system asks Party C's ERP system at what time it should deliver the container. Party C's ERP system does not know Trucking Company B, but can check the delegation to Trucking Company B that Party A has registered at Authorisation Registry D. Because this delegation is in order, Party C's ERP system shares a time slot with Trucking Company B's ERP.

This example is detailed under [use cases](#).

Enable control over own data through management of consent

As described under key functionalities '[facilitate flexible authorisations](#)' and '[enable data exchange based on delegations](#)', iSHARE aims to enable parties to grant other parties or persons access to (parts of) their data or services. At least as important is iSHARE's aim to allow parties to modify or withdraw these access rights, to their data or services, whenever they wish. This is called management of consent, and enables full control over own data at any moment in time.

Example:

- In the example described under key functionality '[enable data exchange based on delegations](#)', Party A hires Trucking Company B to deliver Container X to Party C. Trucking Company B's ERP system asks Party C's ERP system at what time it should deliver the container. Party C's ERP system does not know Trucking Company B, but can check the delegation to Trucking Company B that Party A has registered at Authorisation Registry D. Because this delegation is in order, Party C's ERP system shares a time slot with Trucking Company B's ERP.
- Now imagine that moments before Trucking Company B's ERP system asks Party C's ERP system for a time slot, Party C revokes Party A's access to requesting a time slot. Consequently, Trucking Company B's request for a time slot gets an access forbidden message; Trucking Company B's request is NOT accepted because Party A, and therewith delegated Trucking Company B, is no longer authorised to ask for a time slot.

Party C, as showcased, remains in full control over its own data and services at any moment in time. This example is detailed under [use cases](#).

Provide a trust framework

Within iSHARE, it is the explicit aim to define a trust framework based on a synthesis between technological and legal aspects. In practical terms the aim is to let iSHARE participants sign one contract with the Scheme Owner, on the basis of which they have a contract with all participants within iSHARE. In other words, participants within iSHARE do not need to sign separate contracts with each other to share data with each other (although they are free to define additional contracts that do not conflict with the iSHARE framework).

An important tool within the trust framework are licenses which define the conditions under which data can be exchanged or services can be consumed. For functional details on licenses, see the [Licenses page](#).

The trust framework is depicted under [Primary use cases](#) and needs appropriate technological underpinning so that parties can authenticate each other in a reliable way.

Technical overview

The iSHARE Scheme can be characterised as an API (Application Programming Interface) architecture for identification, authentication and authorisation based on a modified version of the widely used OAuth and OpenID Connect standards. The APIs specified for every role within iSHARE enable standardised interaction between computer systems.

Important

APIs manage access to services of an organisation, services that can be consumed by other parties. Services accessible through APIs can let those (machines or humans) that access the service do anything between reading simple data, to receiving complex instructions, to adding information to a database. If a truck's systems send a time and location to another party's 'Estimated Time of Arrival'-service, for example, this service might respond with an optimal route to take and an Estimated Time of Arrival. Within iSHARE, the terms 'service consumption' and 'service provision' are used to specify how parties interact with each other (with, in this example, the truck's owner the Service Consumer, and the other party the Service Provider). Note that while the word data exchange is not literally in these terms, API service provision and consumption ALWAYS entails data exchange.

The API architecture of iSHARE also builds upon the following components:

- **PKI and digital certificates;**
For the authentication of parties and machines, iSHARE uses PKI and digital certificates.
- **HTTP over TLS (HTTPS);**
iSHARE uses the commonly used HTTP protocol for its communications, including TLS to encrypt the communications.
- **RESTful architectural style;**
iSHARE uses the RESTful architectural style to structure APIs and HTTP calls.
- **JSON/JWT;**
Data exchanged in the iSHARE context is structured using the JSON standard. Where non-repudiation is required, JWT's are used;
- **XACML.**
Delegations are structured according to a JSON port of the XACML standard.

The combination of the above standards and protocols leads to a certain dynamic between the [roles in the iSHARE framework](#). In essence, Service Consumers acquire a token which allows them to access certain services from certain Service Providers. The roles specified in the iSHARE framework are loosely based on the OAuth standard.

For a full explanation and description of all APIs, standards and protocols, please refer to the [Developer Portal](#).

Framework and roles

iSHARE aims to provide a generic building block for service provision, widely applicable in the logistics sector. This requires a framework that can be applied to the wide variety of use cases possible in practice. This chapter explains the iSHARE framework, its roles, and its relations, step-by-step.

i Important (and as under technical overview)

APIs manage access to services of an organisation, services that can be consumed by other parties. Services accessible through APIs can let those (machines or humans) that access the service do anything between reading simple data, to receiving complex instructions, to adding information to a database. If a truck's system sends a time and location to another party's 'Estimated Time of Arrival'-service, for example, this service might respond with an optimal route to take and an Estimated Time of Arrival. Within iSHARE, the terms 'service consumption' and 'service provision' are used to specify how parties interact with each other (with, in this example, the truck's owner the Service Consumer, and the other party the Service Provider). Note that while the word data exchange is not literally in these terms, API service provision and consumption ALWAYS entails data exchange.

The iSHARE framework consists of six roles that, depending on the situation, interact with each other based on the iSHARE scheme agreements. Each role has a certain function in the scheme and bears certain responsibilities, as described below:



Any party fulfilling a role in the iSHARE framework must be iSHARE adhering or iSHARE certified:

- Parties fulfilling **adhering roles**, depicted in purple, provide and consume services under iSHARE. These parties adhere to the iSHARE terms of use;
 - Note: as it is the responsibility of the Service Provider to determine the Entitled Party, the Service Provider can choose to provide services where the Entitled Party is not admitted to iSHARE. In this event, the responsibilities of the Entitled Party are shifted to the Service Provider in question. This is particularly useful for Service Providers who have existing (smaller) customers, who do not have own systems, or are only an Entitled Party for services at a single Service Provider.
- Parties fulfilling **certified roles**, depicted in grey, facilitate functions that Adhering Parties can rely upon when providing or consuming services. To become certified, these parties must not only prove adherence to the iSHARE terms of use, but also meet several role-specific criteria.

Adhering roles

In any iSHARE use case, the three adhering roles appear: a Service Consumer always consumes a Service Provider's service on the basis of the Entitled Party's entitlements.

Adhering role:	Role description:
Service Consumer	<p>The Service Consumer-role is fulfilled by a legal entity that consumes a service, such as data, as provided by a Service Provider. This legal entity is in need of the result of a service; for example, a trucking company that needs to know its optimal route and Estimated Time of Arrival.</p> <p>A Service Consumer can be represented by a machine (its system) or a human (e.g. the trucker), fittingly called the Machine Service Consumer and the Human Service Consumer.</p>
Service Provider	<p>The Service Provider-role is fulfilled by a legal entity that provides a service, such as data, for consumption by a Service Consumer. This legal entity provides the result of a service that Service Consumer(s) need; for example the party that uses a truck's a time and location to calculate and communicate the truck's optimal route and Estimated Time of Arrival.</p>

Adhering role:	Role description:
Entitled Party	<p>The Entitled Party-role is fulfilled by a legal entity that has one or more rights to a service provided by a Service Provider, for example to data. These rights, or entitlements, are established in a legal relation between the Entitled Party and the Service Provider.</p> <p>The Entitled Party- and Service Consumer-roles can be fulfilled by the same entity - i.e. a legal entity that consumes a service based on its own entitlements to this service (for example, the trucking company's entitlement to request Estimated Time of Arrival- and optimal route information) - but this is not necessary. Legal entities that are entitled to a service can delegate other entities to consume this service on its behalf: the legal entity consuming the service, then, does so on the basis of <i>another entity's</i> entitlements. In such use cases, as always, the Service Consumer consumes a Service Provider's service on the basis of the Entitled Party's entitlements, but the Service Consumer-role is fulfilled by another entity than the Entitled Party-role.</p> <p>Our trucking company, for example, could have been delegated the right to request Estimated Time of Arrival- and optimal route information by an Entitled Party, that had originally planned to transport its goods itself but instead hired the trucking company to do so. It therefore delegated its own right to request Estimated Time of Arrival- and optimal route information to the trucking company.</p>

Certified roles

For the controlled provision and consumption of services, Adhering Parties (and specifically, the humans and machines representing them) must be identified, authenticated, and authorised. The tooling necessary for these processes *can* be implemented by Adhering Parties. Such tooling is expensive, however, and must be constantly updated to keep in check with the latest security standards. To make sure no such tooling needs to be implemented by Adhering Parties before they start providing or consuming services under iSHARE (and therefore, to improve iSHARE's scalability), iSHARE recognises several certified roles fulfilled by legal entities that offer outsourced identification, authentication, and authorisation tooling to Adhering Parties.

Certified role:	Role description:
Identity Provider	<p>The Identity Provider-role is fulfilled by a legal entity whose tooling identifies and authenticates humans (and specifically, Human Service Consumers representing Service Consumers). An Identity Provider:</p> <ul style="list-style-type: none"> • Provides identifiers for humans; • Issues credentials (i.e. a password or electronic keycard) to humans; • On the basis of this identification information, identifies and authenticates humans for Service Providers. • Holds information on authorisations of humans representing a Service Consumer; i.e. information indicating which humans are authorised to act on a Service Consumer's behalf. • Can check, on the basis of this information, whether a human representing a legal entity is authorised to take delivery of a service; • Can confirm whether this is the case to the Service Provider. <p>As a result, Service Providers can outsource identification and authentication of humans, as well as tasks concerning the management of authorisation and delegation information of humans, to an Identity Provider instead of implementing their own tooling.</p>
Identity Broker	<p>Different humans might hold identifiers at different Identity Providers. Also, Service Providers might need to connect to several Identity Providers. To make sure Service Providers do not need a relation with each Identity Provider individually, an Identity Broker is introduced. The Identity Broker-role is fulfilled by a legal entity that provides Service Providers access to different Identity Providers, and that offers humans the option to choose with which Identity Provider to identify and authenticate themselves throughout the iSHARE Scheme.</p> <p>As a result, if Service Providers choose to outsource identification and authentication to more than one Identity Provider, they can connect to an Identity Broker instead of to several Identity Providers.</p>
Authorisation Registry	<p>The Authorisation Registry-role is fulfilled by a legal entity who provides solutions for Adhering Parties for the storage of delegation- and authorisation information. An Authorisation Registry:</p> <ul style="list-style-type: none"> • Can hold information on delegations to Service Consumers; i.e. information indicating what parts of the rights of an Entitled Party are delegated to a Service Consumer. • Can check, on the basis of this information, whether a machine representing a legal entity is authorised to take delivery of a service; • Can confirm whether this is the case to the Service Provider. <p>As a result, Adhering Parties can outsource tasks concerning the management of authorisation and delegation information to an Authorisation Registry instead of implementing their own tooling.</p>

As detailed under [functional requirements per role](#), to become an iSHARE Certified Party, a legal entity must (first) be admitted as a participant by the Scheme Owner (in the relevant role).

iSHARE compatible software

Next to iSHARE adherence and certification, the concept of iSHARE compatibility exists. This concept is reserved for software that technically adheres to the iSHARE Scheme (i.e. is iSHARE compatible), and can be sold to parties fulfilling adhering- and certified roles. Note that parties using iSHARE compatible software within an iSHARE

context must be adhering or certified, whereas a party that delivers iSHARE compatible software does not need to be so.

Role of the Scheme Owner

One last role, not part of the basic iSHARE framework, is that of the Scheme Owner. The Scheme Owner-role is fulfilled by the legal entity that keeps the scheme, and its network of participants, operating properly. How exactly is found under the [detailed Operational descriptions](#). It is this Scheme Owner that decides whether a party is admitted to the iSHARE network (and whether this is as an Adhering- or Certified Party).

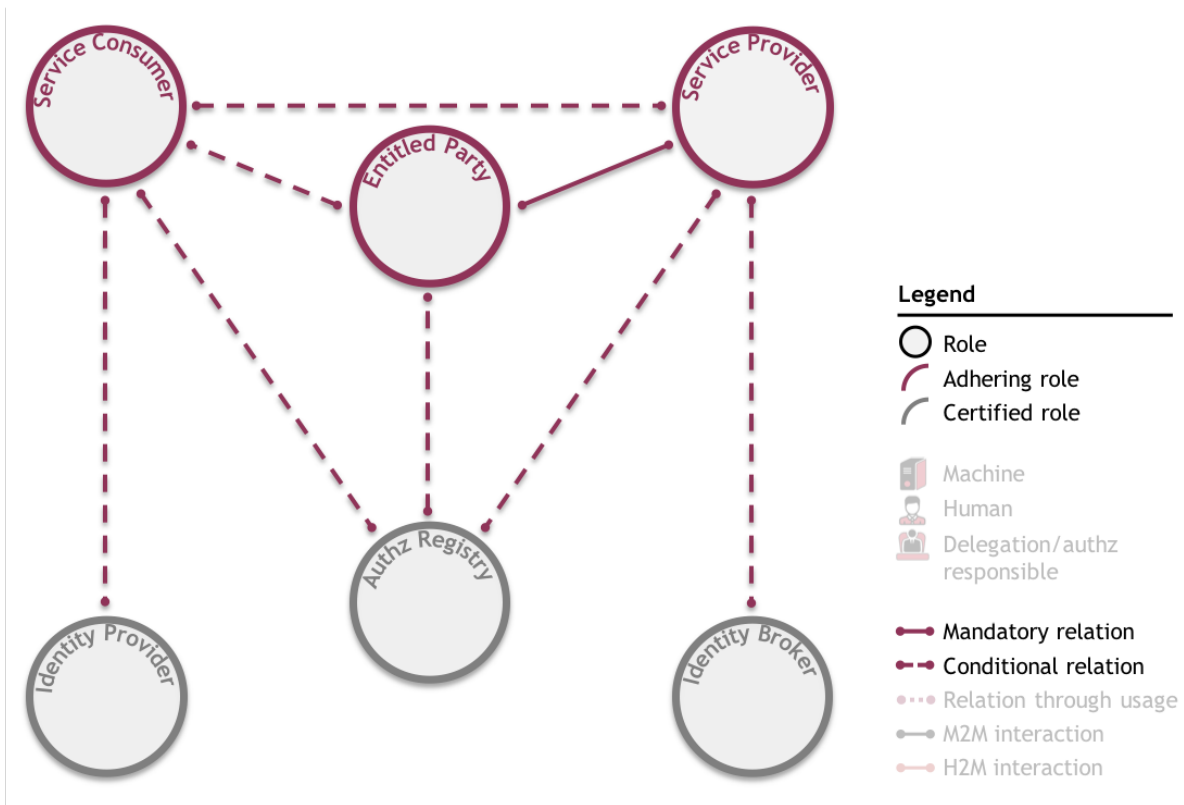
The Scheme Owner plays a fundamental role in any iSHARE use case. Every participant to the iSHARE Scheme must have a relation with the Scheme Owner, and can check at the Scheme Owner whether other parties participate in iSHARE. These are prerequisites, however, which is why the Scheme Owner does not play a direct role (and is not depicted) in any of the use cases. Please refer to the [detailed Functional descriptions](#) for details on how the Scheme Owner facilitates trust in the iSHARE Scheme.

Framework and roles in use cases

All of iSHARE's use cases can be depicted in the iSHARE framework. Their complexity is dependent on:

- The interaction model (Machine to Machine or Human to Machine);
i.e. whether the Service Consumer is represented by a machine or a human.
- Whether delegation takes place;
i.e. whether the Service Consumer-role is fulfilled by another entity than the Entitled Party-role. How delegations work exactly is explained [here](#).
- Whether parties fulfilling adhering roles use their own tooling for identification, authentication, and authorisation or outsource these processes and the information necessary for these processes to certified roles instead.

Hypothetically, and dependent on the above, a use case could include all of the following relations between roles:



Note that the only relation mandatory in all use cases is the relation between the Entitled Party and the Service Provider, which establishes the entitlements of the Entitled Party. In the depiction of iSHARE's use cases, all legal relations are shown before the actual interaction is plotted in the framework.

Legal provisions

The legal underpinning of iSHARE and its trust framework consists of a contract between all iSHARE participants and the iSHARE Scheme Owner (the so called Accession Agreement). Based on this one contract with the Scheme Owner, all participants are bound to the common iSHARE terms of use and can appeal to each other to abide by these rules (in legal terms this is called perfection (Dutch: *derdenwerking*)).

Two main documents make up iSHARE's legal provisions:

- 1. The Accession Agreement**

The main contract between the participant and the iSHARE Scheme Owner. This contract refers to the terms of use, including all iSHARE specifications, to which all participants must abide. After signing the Accession Agreement, a party becomes a participant of the iSHARE Scheme either as an Adhering Party or a Certified Party. There are two separate Accession Agreements: one for Adhering Parties and one for Certified Parties.

- 2. The Terms of Use**

The Terms of Use further define the rights and obligations of every iSHARE Participant and the Scheme Owner. The Terms of Use apply to any party that has signed the Accession Agreement. The Terms of Use also state that participants fully abide by the iSHARE scheme specifications.

For the details of the Accession Agreements, the full version of the Terms of Use and the information available on legal context, please refer to the [detailed Legal descriptions](#).

Licenses

Within iSHARE it is possible to explicitly provide instructions on how a service may be consumed or under which conditions data is exchanged. These instructions or conditions are called 'licenses'. Licenses are a crucial part of iSHARE, because they provide its participants the possibility to clearly state what is and what is not allowed. Since all iSHARE participants are bound to the same contract and underlying scheme rules, participants can appeal to each other to follow the provided licenses. Please refer to the iSHARE Terms of Use for a detailed legal explanation.

Operational provisions

The iSHARE Scheme is constantly improved in collaboration with its stakeholders. Keeping the scheme, and its network of participants operating properly is facilitated by the iSHARE Scheme Owner.

The main responsibilities of the Scheme Owner include:

- Management of the iSHARE Scheme (specifications);
- Management of the iSHARE network (participants);
- Management of the iSHARE brand.

To fulfil its responsibilities, the Scheme Owner facilitates the correct operation of the iSHARE scheme and -network through administering several aspects:

- [Operational processes](#)
- [Service levels](#)
- [Communication](#)

The Scheme Owner is part of a wider governance framework, which can be found in the [introduction of the scheme](#).

Use cases

This chapter builds on the [iSHARE framework](#) to showcase the scheme's [key functionalities](#) in four use cases:

1. **Use case: M2M interaction (with fine-grained authorisation)** showcases:
 - Support Machine to Machine (M2M) interaction;
 - Facilitate flexible authorizations, applicable in any context.
2. **Use case: H2M interaction (with coarse-grained authorisation)** showcases:
 - Support Human to Machine (H2M) interaction;
 - Facilitate flexible authorizations, applicable in any context.
3. **Use case: portable identity** showcases:
 - Facilitate portable identity(s) for parties and humans.
4. **Use case: delegation (and management of consent)** showcases:
 - Enable data exchange based on delegations - even between unknown parties;
 - Enable control over own data through management of consent.

Structure

Each use case includes:

- A description and depiction of the roles and relations;
- A description of the prerequisites, and a depiction of prerequisite registration;
- A description and depiction of the use case;
- A sequence diagram;
- A reference to what needs to be technically implemented for this use case.

The depicted use cases are only a selection of iSHARE's use case scope. For the full scope, please refer to the [detailed Functional descriptions](#).

Use case: M2M interaction (with fine-grained authorization)

This use case showcases iSHARE's key functionality '[support Machine to Machine \(M2M\) interaction](#)'.

The example described in the linked chapter is as follows:

- Every day, the ERP system (machine) of Party A requests a status update from the ERP system (machine) of Party B. Party B's ERP system automatically responds with the requested status update. No humans are needed to interfere.

To also showcase iSHARE's key functionality '[facilitate flexible authorizations](#)', Party A's ERP system (machine) is ONLY allowed to request status updates concerning line X of bill of lading Y. This can be considered a fine-grained authorization.

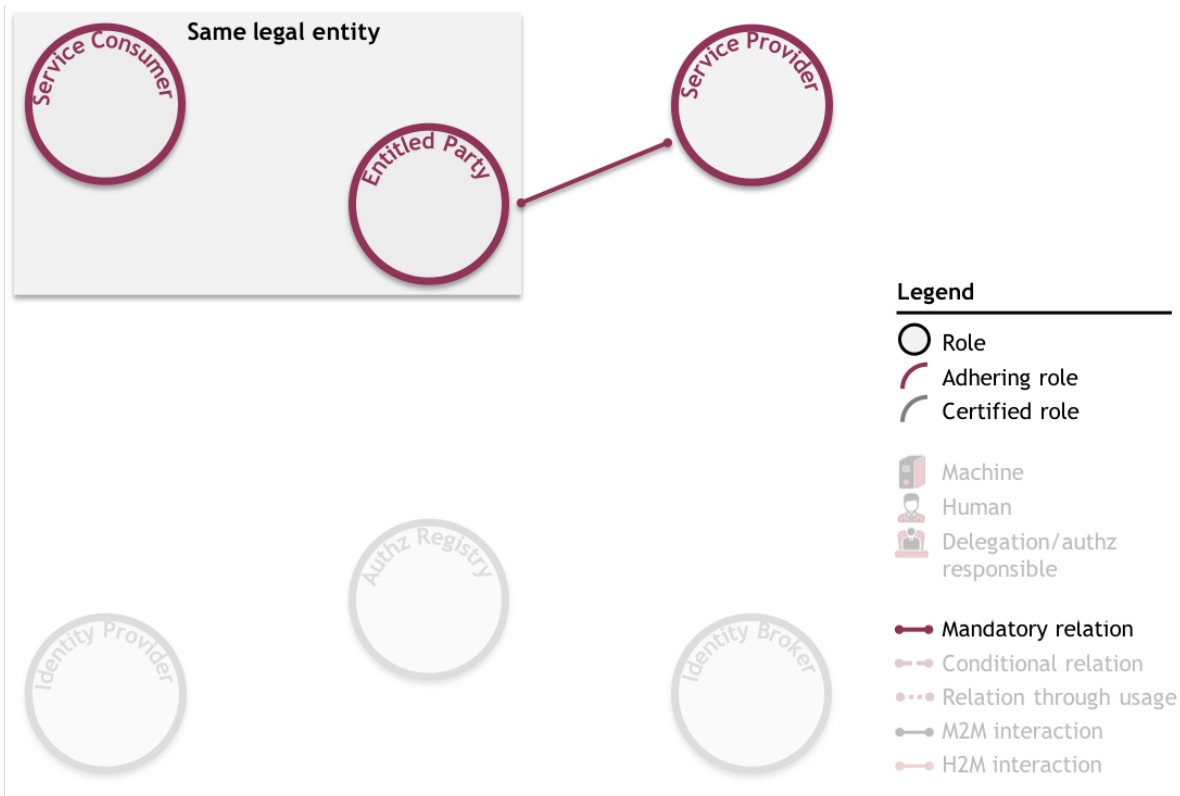
The following explains this example in detail, utilising the iSHARE framework.

Roles and Relations

The following roles are fulfilled in this use case:

- Party A requests a status update, so it is the legal entity fulfilling the **Service Consumer**-role;
- Party B responds with the status update, so it is the legal entity fulfilling the **Service Provider**-role;
- No delegation takes place, so Party A also fulfils the **Entitled Party**-role;
- As this is a M2M use case, a **Machine Service Consumer** represents Party A.

The only **legal relation** is the mandatory relation between the Entitled Party (Party A) and the Service Provider (Party B), which establishes the entitlements of the Entitled Party (Party A). As depicted:



Prerequisites

It is prerequisite of this use case that:

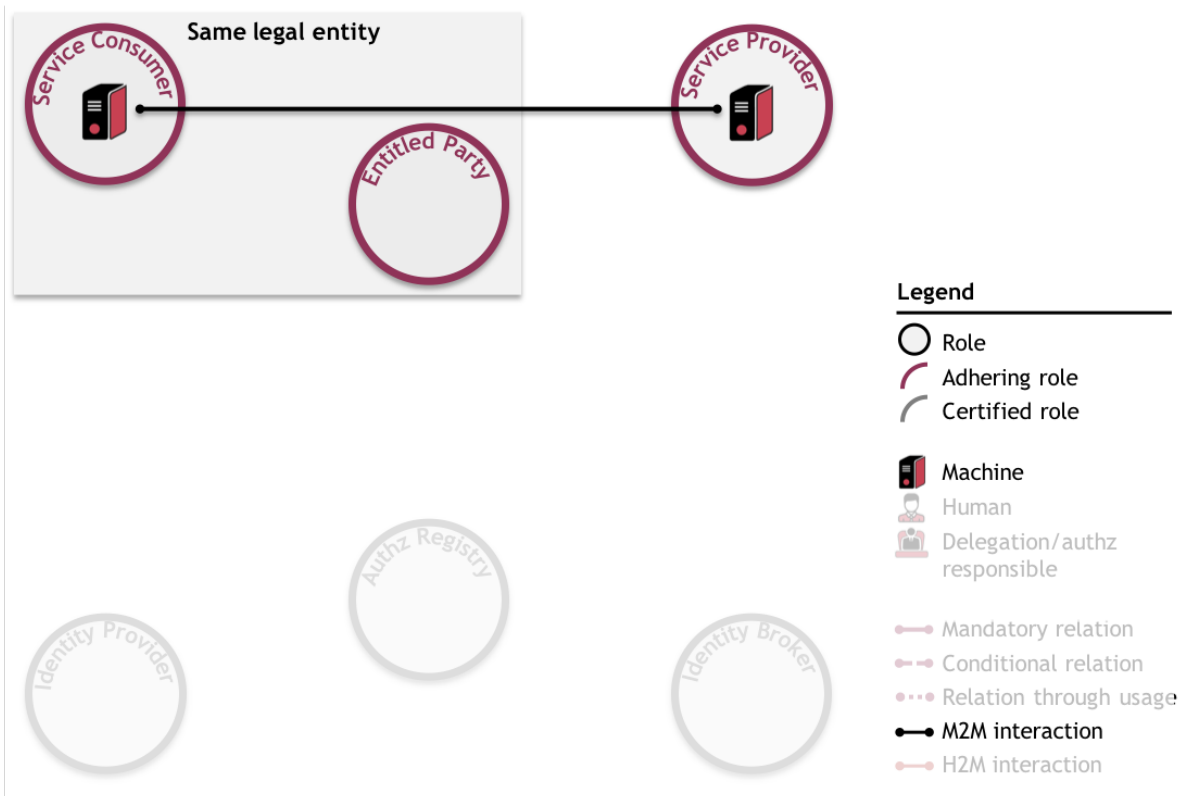
- The Service Provider (Party B) has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services, i.e. Party B has information indicating that Party A is allowed to request status updates concerning line X of bill of lading Y from its ERP system;
- The Service Consumer (Party A) is able to authenticate the Service Provider (Party B);
- The Service Provider (Party B) is able to authenticate the Service Consumer (Party A).

Use case

The use case consists of the following steps:

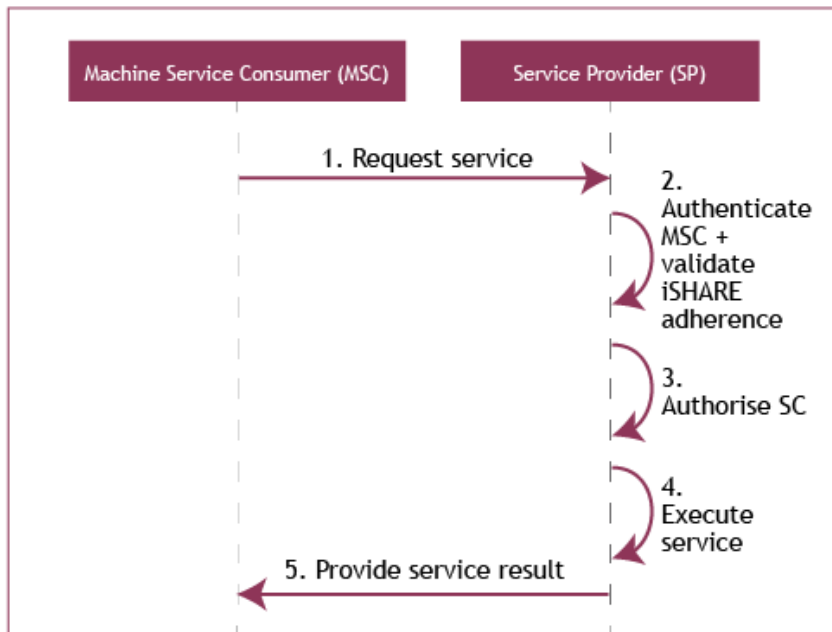
1. The Machine Service Consumer (of Party A) requests a service from the Service Provider (Party B);
2. The Service Provider (Party B) authenticates the Machine Service Consumer (of Party A) and validates the iSHARE adherence of the Service Consumer (Party A);
3. The Service Provider (Party B) authorizes the Machine Service Consumer of the Service Consumer (Party A) based on the entitlement information registered with the Service Provider (Party B);
4. The Service Provider (Party B) executes the requested service;
5. The Service Provider (Party B) provides the service result to the Machine Service Consumer (of Party A).

As depicted:



Note that this use case is exactly the same as primary use case 1, as found under [detailed Functional descriptions](#).

Sequence diagram



What needs to be implemented technically for this use case is described [generically](#), and specifically per role in the iSHARE Developer Portal.

Use case: H2M interaction (with coarse-grained authorization)

This use case showcases iSHARE's key functionality '[support Human to Machine \(H2M\) interaction](#)'.

The example described in the linked chapter is as follows:

- Human X, working for Party A, requests a status update from the ERP system (machine) of Party B. It does so via a user interface.

To also showcase iSHARE's key functionality '[facilitate flexible authorizations](#)', Party A's ERP system (machine) is allowed to request ANY information about ANY (part of a) bill of lading. This can be considered a coarse-grained authorization.

The following explains this example in detail, utilising the iSHARE framework.

Roles and Relations

The following roles are fulfilled in this use case:

- Party A requests a status update, so it is the legal entity fulfilling the **Service Consumer**-role;
- Party B responds with the status update, so it is the legal entity fulfilling the **Service Provider**-role;
- No delegation takes place, so Party A also fulfils the **Entitled Party**-role;
- Human X is the **Human Service Consumer** that represents Party A.

The only **legal relation** is the mandatory relation between the Entitled Party (Party A) and the Service Provider (Party B), which establishes the entitlements of the Entitled Party (Party A). As depicted:



Prerequisites

It is prerequisite of this use case that:

- The Service Provider (Party B) has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services, i.e. Party B has information indicating that Party A is allowed to request ANY information about ANY (part of a) bill of lading from its ERP system;
- The Service Consumer (Party A) has and manages its own authorization information indicating which Human Service Consumers are authorized to act on its behalf;
- **The delegation/authorization responsible at the the Service Consumer (Party A) registers the authorization information at the Service Provider (Party B);**
- The Human Service Consumer (Human X) is able to authenticate the Service Provider (Party B);
- The Service Provider (Party B) is able to authenticate the Human Service Consumer (Human X);
- **The Human Service Consumer (Human X) has been issued identity credentials by the Service Provider (Party B).**

The prerequisites in bold are depicted as follows:



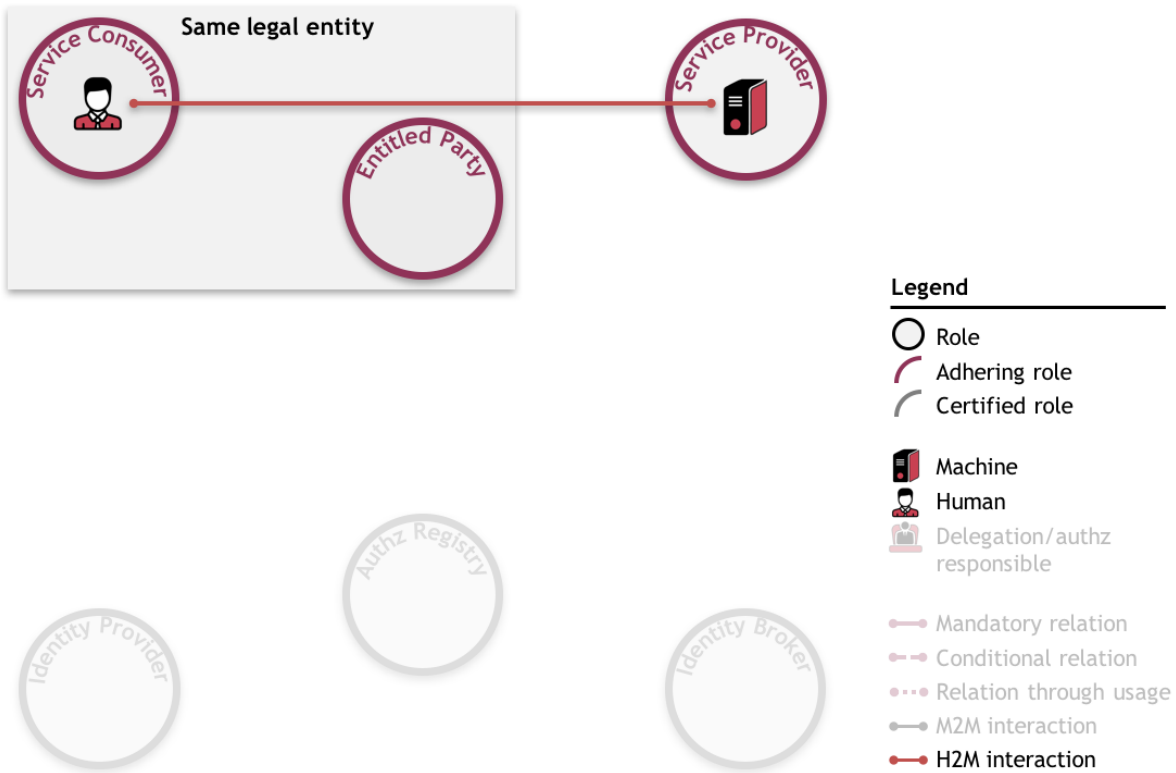
Use case

The use case consists of the following steps:

1. The Human Service Consumer (Human X) requests a service from the Service Provider (Party B);
2. The Service Provider (Party B) authenticates the Human Service Consumer (Human X), and validates the iSHARE adherence of the Service Consumer (Party A);

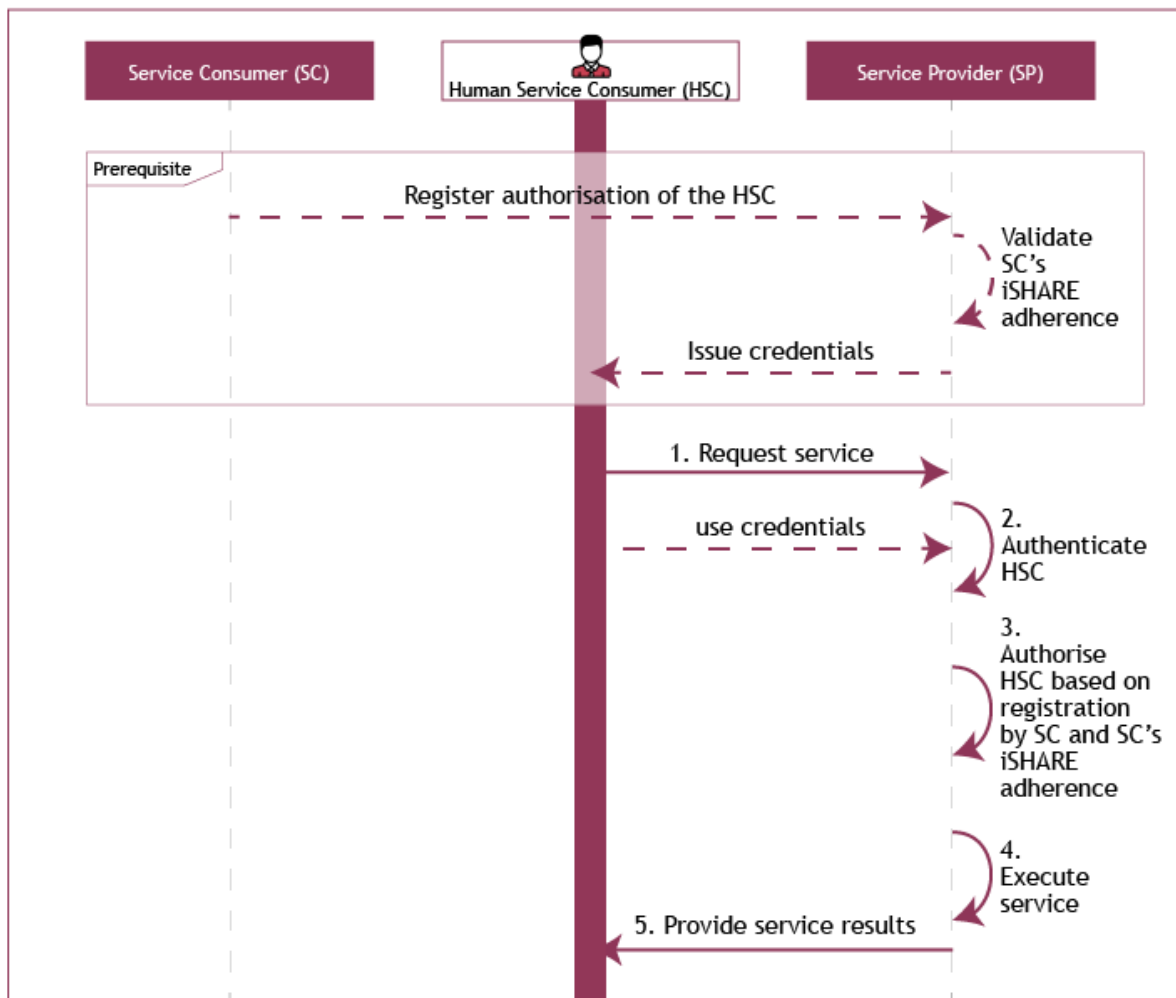
3. The Service Provider (Party B) authorizes the Human Service Consumer (Human X) of the Service Consumer (Party A) based on the entitlement- and authorization information registered with the Service Provider (Party B);
4. The Service Provider (Party B) executes the requested service;
5. The Service Provider (Party B) provides the service result to the Human Service Consumer (Human X).

As depicted:



Note that this use case is exactly the same as primary use case 2, as found under [detailed Functional descriptions](#).

Sequence diagram



What needs to be implemented technically for this use case is described [generically](#), and specifically per role in the iSHARE Developer Portal.

Use case: portable identity

This use case showcases iSHARE's key functionality '[facilitate portable identity\(s\) for parties and humans](#)'.

The example described in the linked chapter is as follows:

- Human X, working for Party A, has credentials issued by iSHARE certified Identity Provider Y. The credentials, and thus the identity of Human X, can be used to identify and authenticate Human X at party B.

Human X will now use its Identity Provider Y credentials to request a status update from the ERP system (machine) of Party B.

The following explains this example in detail, utilising the iSHARE framework.

Roles and Relations

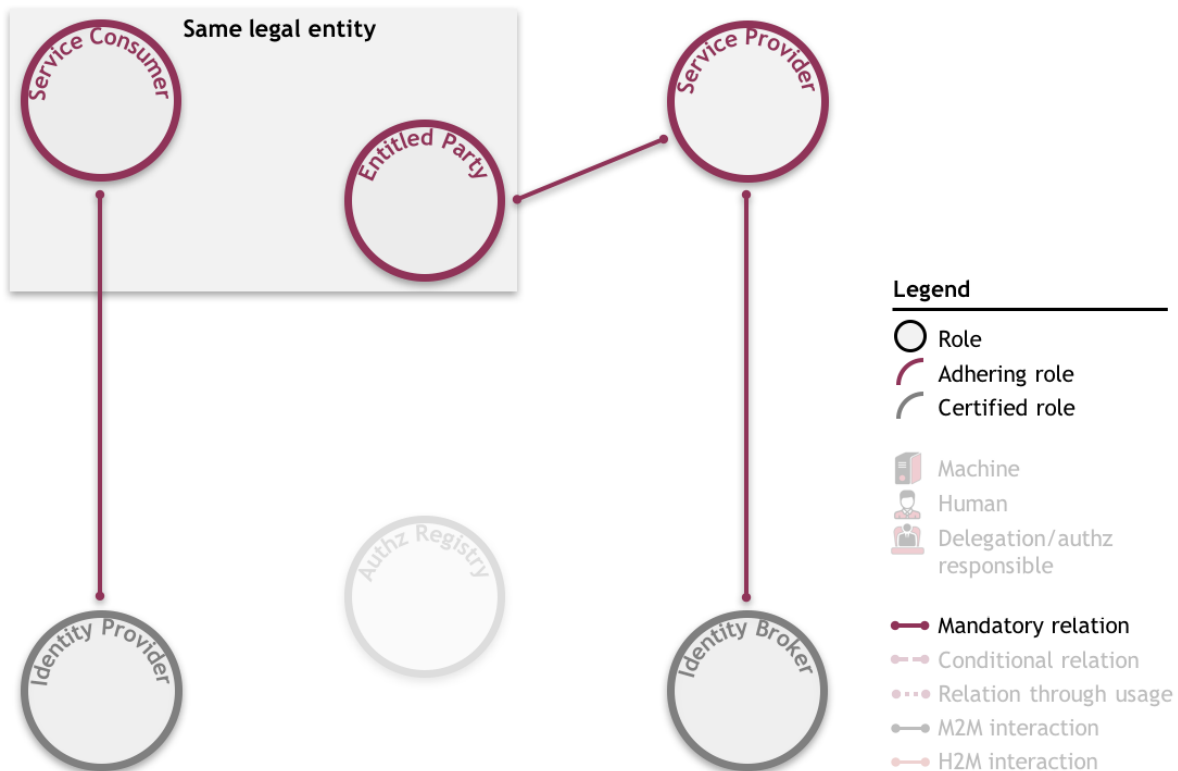
The following roles are fulfilled in this use case:

- Party A requests a status update, so it is the legal entity fulfilling the **Service Consumer**-role;
- Party B responds with the status update, so it is the legal entity fulfilling the **Service Provider**-role;
- No delegation takes place, so Party A also fulfils the **Entitled Party**-role.
- Human X is the **Human Service Consumer** that represents Party A;
- Identity Provider Y is one of the **Identity Providers** to which Party B has outsourced identification, authentication and authorisation of humans, and the party that has given Human X his keycard.
- Optionally (and shown in this case), Identity Broker Z is the **Identity Broker** that provides Party B access to different Identity Providers, and that offers Human X the option to choose with which Identity Provider to identify and authenticate itself.

Legal relations

- As always, a mandatory relation between the Entitled Party (Party A) and the Service Provider (Party B) establishes the entitlements of the Entitled Party (Party A);
- A mandatory relation between the Service Provider and the Identity Broker covers the use of Identity Broker Z's services, including a connection to several Identity Providers, by the Service Provider (Party B);
- A mandatory relation between the Service Consumer (Party A) and Identity Provider Y covers the use of Identity Provider Y's keycards by the the Service Consumer's (Party A's) humans, including Human X.

As depicted:



Prerequisites

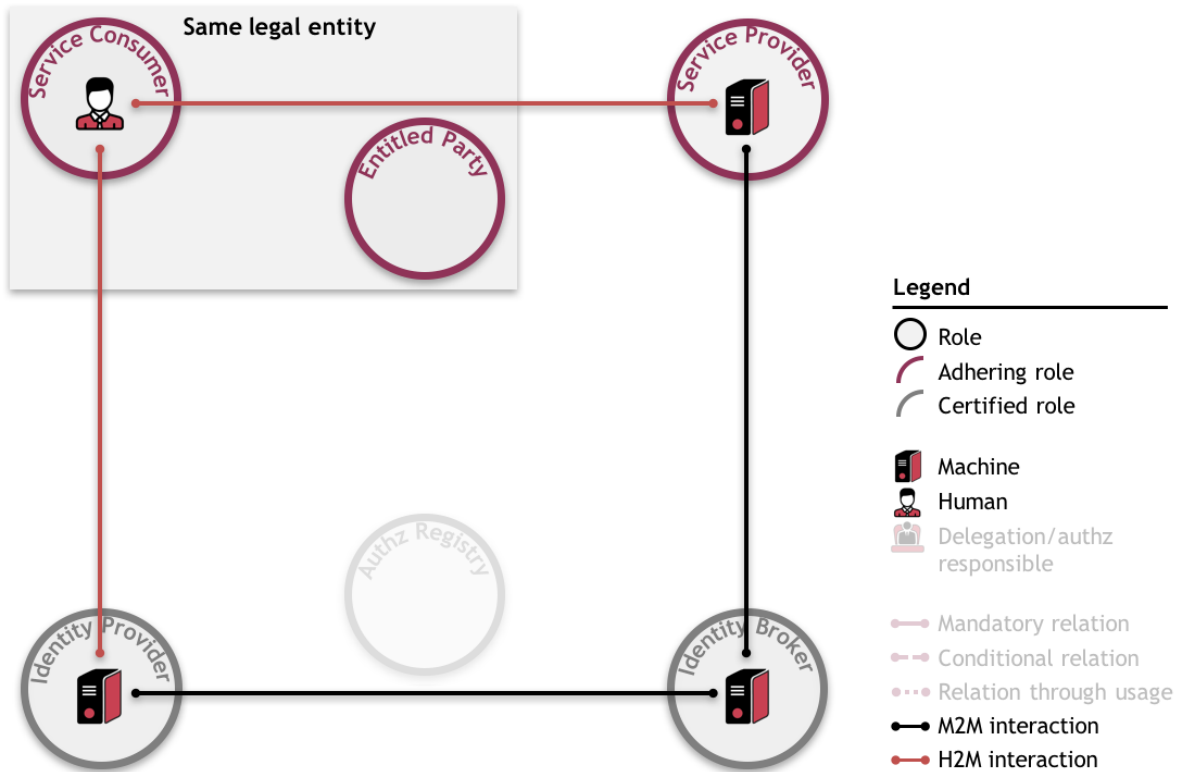
It is prerequisite of this use case that:

- The Service Provider (Party B) has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services, i.e. Party B has information indicating that Party A is entitled to status updates from its ERP system;
- The Service Consumer (Party A) has and manages its own authorization information indicating which Human Service Consumers are authorized to act on its behalf;
- **The delegation/authorization responsible at the the Service Consumer (Party A) registers the authorization information at the Identity Provider (Y);**
- The Human Service Consumer (Human X) is able to authenticate the Service Provider (Party B);
- The Service Provider (Party B) is able to authenticate the Human Service Consumer (Human X);
- The Identity Provider (Y) is able to authenticate the Service Provider (Party B);
- The Service Provider (Party B) is able to authenticate the Identity Provider (Y);
- The Identity Broker (Z) is able to authenticate the Service Provider (Party B);
- The Service Provider (Party B) is able to authenticate the Identity Broker (Z);
- **The Human Service Consumer (Human X) has been issued identity credentials by the Identity Provider (Y).**

The prerequisites in bold are depicted as follows:

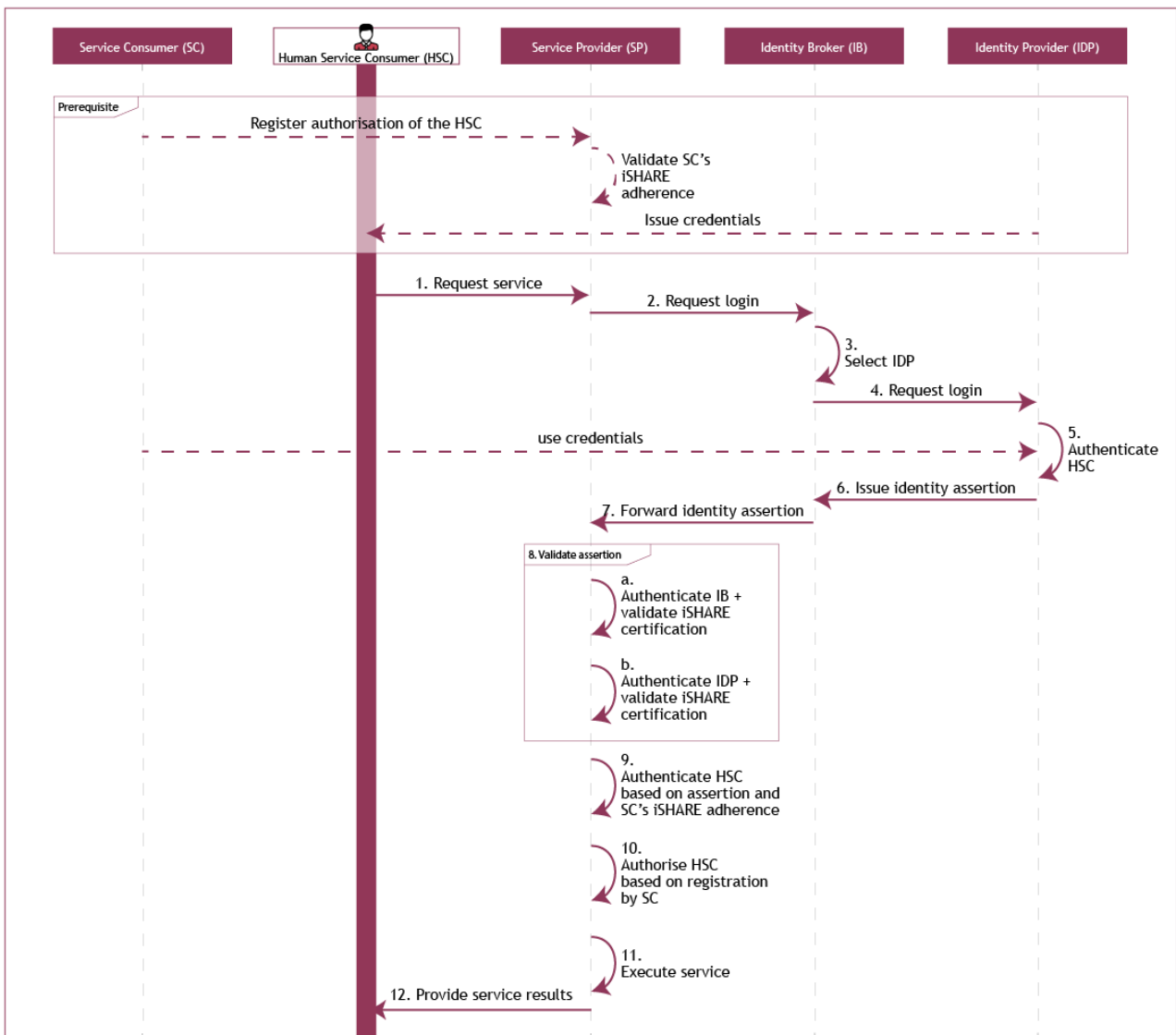
12. The Service Provider (Party B) provides the service result to the Human Service Consumer (Human X).

As depicted:



Note that this use case is exactly the same as primary use case 3, as found under [detailed Functional descriptions](#). In this section, the same use case is also explained without an Identity Broker.

Sequence diagram



What needs to be implemented technically for this use case is described [generically](#), and specifically per role in the iSHARE Developer Portal.

Use case: delegation (and management of consent)

This use case showcases iSHARE's key functionality '[enable data exchange based on delegations - even between unknown parties](#)'.

The example described in the linked chapter is as follows:

- Party A hires Trucking Company B to deliver Container X to Party C. Trucking Company B's ERP system asks Party C's ERP system at what time it should deliver the container. Party C's ERP system does not know Trucking Company B, but can check the delegation to Trucking Company B that Party A has registered at Authorization Registry D. Because this delegation is in order, Party C's ERP system shares a time slot with Trucking Company B's ERP.

The following explains this example in detail, utilising the iSHARE framework.

After explanation of the delegation use case, a scenario is introduced that showcases key functionality '[enable control over own data through management of consent](#)'. In this [Alternative scenario on management of consent](#), Party C decides to revoke Party A's access to requesting a time slot.

Roles and Relations

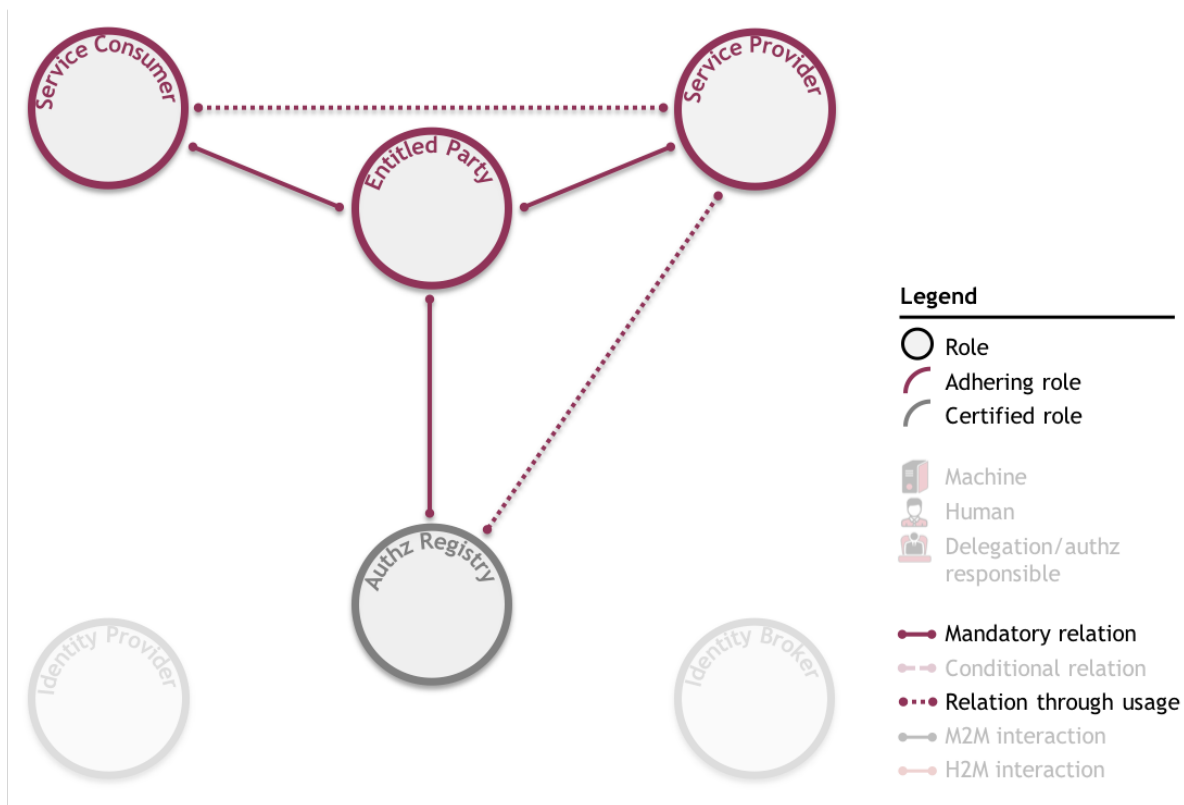
The following roles are fulfilled in this use case:

- Delegation takes place, with Party A the party originally entitled to request a time slot. Party A therefore fulfils the **Entitled Party**-role;
 - Trucking Company B is delegated the right to request a time slot, so it is the legal entity fulfilling the **Service Consumer**-role;
 - Party C responds with the time slot, so it is the legal entity fulfilling the **Service Provider**-role;
 - Authorization Registry D is the **Authorization Registry** to which Party A has outsourced managing delegation information.
- As this is a M2M use case, a **Machine Service Consumer** represents Trucking Company B.

Legal relations

- As always, a mandatory relation between the Entitled Party (Party A) and the Service Provider (Party C) establishes the entitlements of the Entitled Party (Party A);
 - A mandatory relation between the Entitled Party (Party A) and the Service Consumer (Trucking Company B) covers the delegation of the right to request a time slot;
 - A mandatory relation between the Entitled Party (Party A) and the Authorization Registry (D) covers the outsourcing of managing delegation information.
- No relation between the Service Consumer (Trucking Company B) and the Service Provider (Party C) is mandatory before service consumption, i.e. the Service Consumer and the Service Provider do not need to know each other. This relation only commences through usage;
 - No relation between the Service Provider (Party C) and the Authorization Registry (D) is mandatory before communication. This relation also commences through usage.

As depicted:

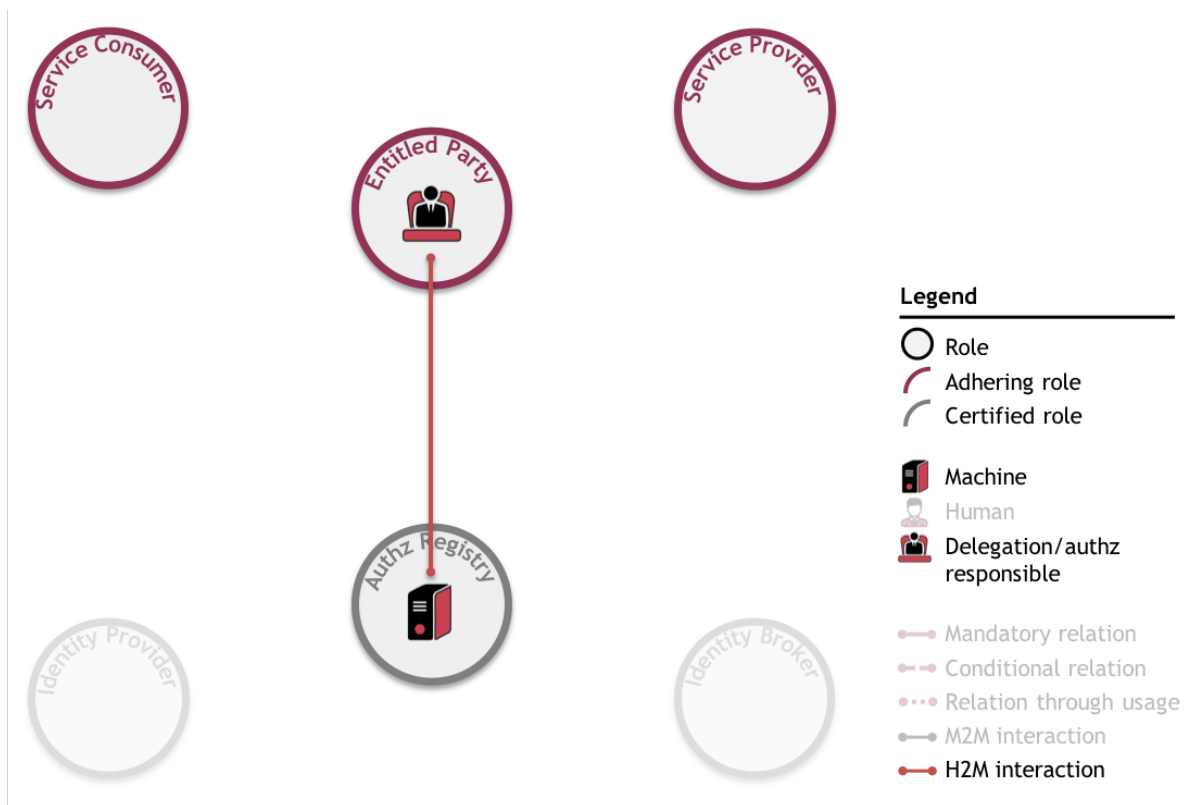


Prerequisites

It is prerequisite of this use case that:

- The Service Provider (Party C) has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services, i.e. Party C has information indicating that Party A is entitled to request a time slot;
- The Service Consumer (Trucking Company B) is able to authenticate the Service Provider (Party C);
- The Service Provider (Party C) is able to authenticate the Service Consumer (Trucking Company B);
- **The delegation/authorization responsible at the Entitled Party (Party A) delegates (part of) the Entitled Party's (Party A's) rights (as registered at the Service Provider (Party C)) to the Service Consumer (Trucking Company B). He registers this delegation in an Authorization Registry (D);**
- The Service Provider (Party C) knows which Authorization Registry (D) to request the delegation evidence from;
- The Service Provider (Party C) is able to authenticate the Authorization Registry (D);
- The Authorization Registry (D) is able to authenticate the Service Provider (Party C);
- It is clear, through scheme agreements, under what conditions an Authorization Registry can provide delegation information to a Service Provider.

The prerequisites in bold are depicted as follows:

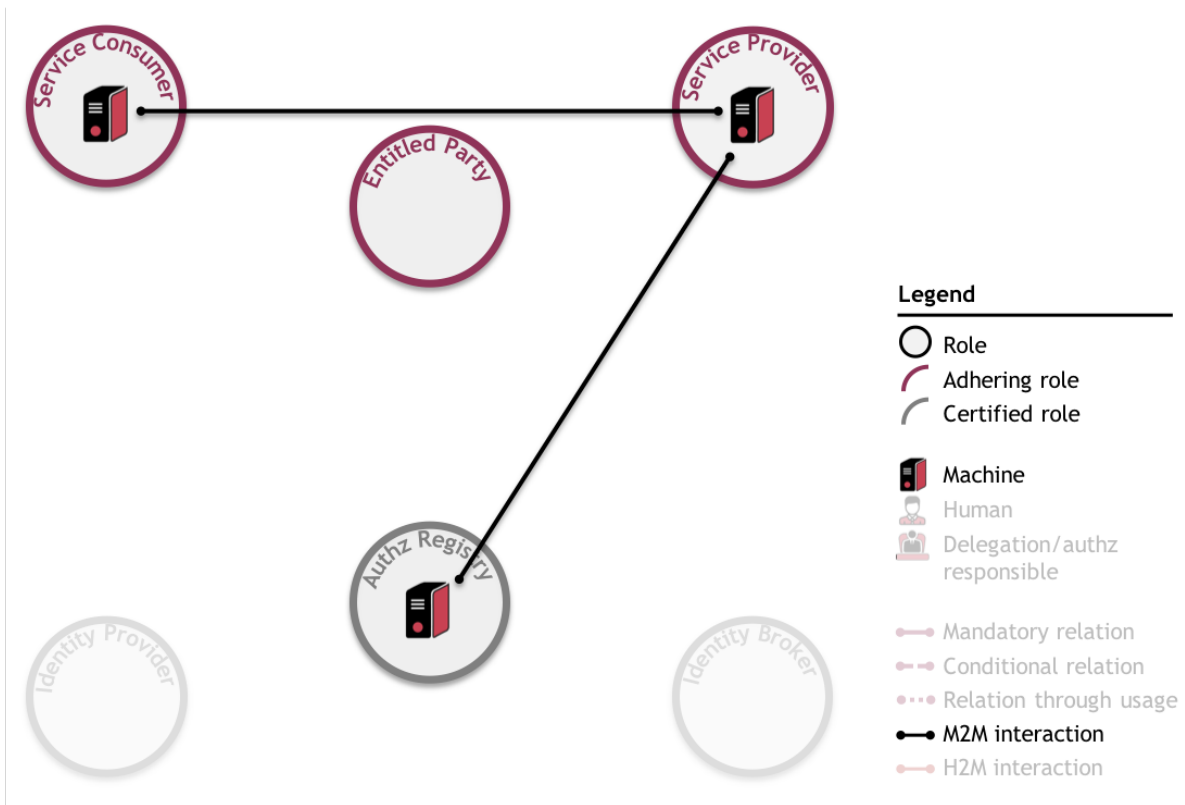


Use case

The use case consists of the following steps:

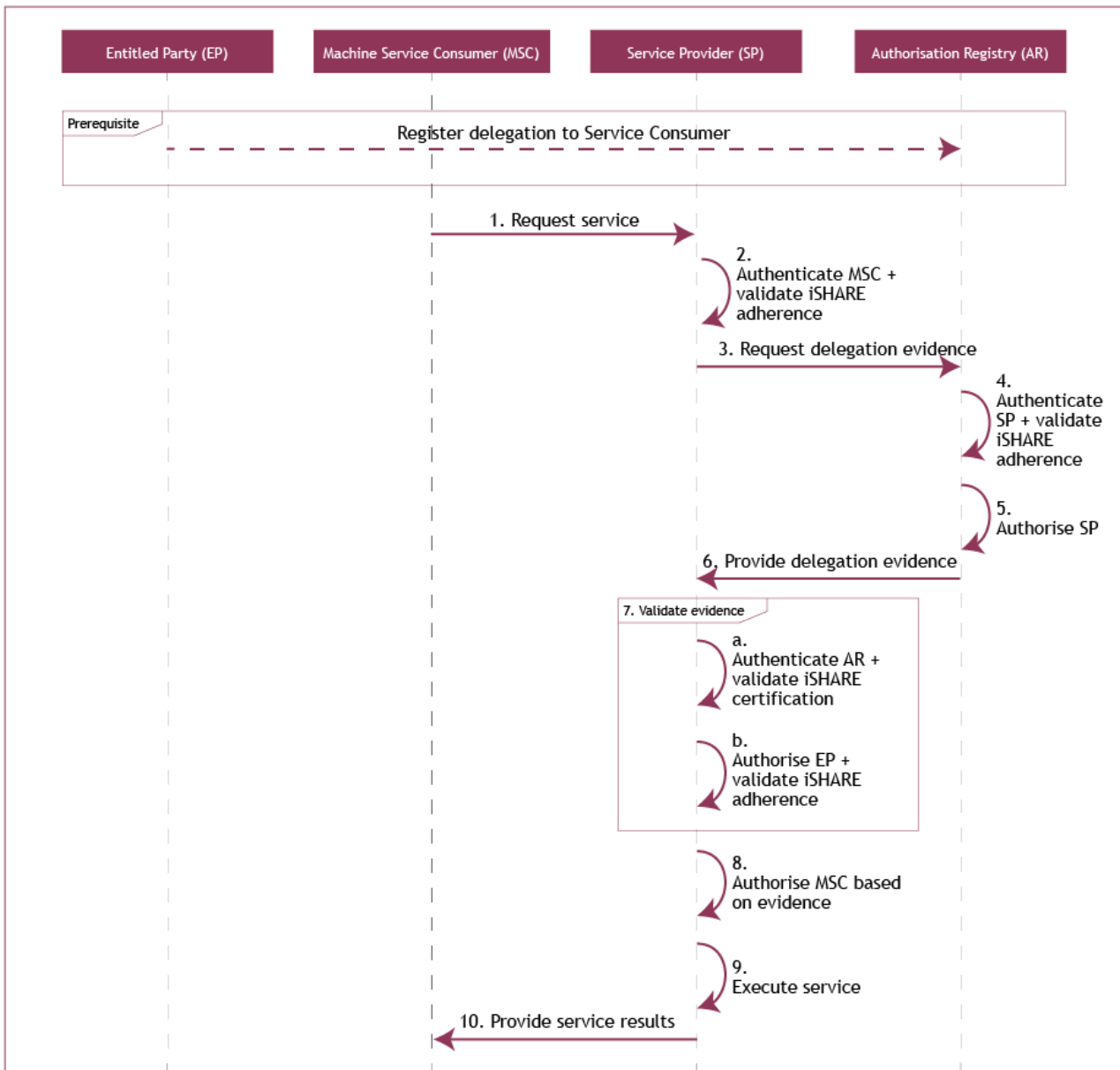
1. The Machine Service Consumer (of Trucking Company B) requests a service from the Service Provider (Party C);
2. The Service Provider (Party C) authenticates the Machine Service Consumer (of Trucking Company B) and validates the iSHARE adherence of the Service Consumer (Trucking Company B);
3. The Service Provider (Party C) requests delegation evidence from the Authorization Registry (D);
4. The Authorization Registry (D) authenticates the Service Provider (Party C) and validates its iSHARE adherence;
5. The Authorization Registry (D) authorizes the Service Provider (Party C) based on the scheme agreements for providing delegation information;
6. The Authorization Registry (D) provides the delegation evidence;
7. The Service Provider (Party C) validates the received delegation evidence through the following steps:
 - a. The Service Provider (Party C) authenticates the Authorization Registry (D) and validates its iSHARE certification;
 - b. The Service Provider (Party C) authorizes the Entitled Party (Party A) based on the entitlement information registered with the Service Provider (Party C), and validates its iSHARE adherence.
8. The Service Provider (Party C) authorizes the Machine Service Consumer of the Service Consumer (Trucking Company B) based on the validity of the delegation evidence;
9. The Service Provider (Party C) executes the requested service;
10. The Service Provider (Party C) provides the service result to the Machine Service Consumer (of Trucking Company B).

As depicted:



Note that this use case is exactly the same as derived use case 1c, as found under [detailed Functional descriptions](#). This section also includes delegation use cases with delegation information held by other roles than an Authorization Registry.

Sequence diagram



Alternative scenario on management of consent

This alternative scenario showcases key functionality 'enable control over own data through management of consent'.

The example detailed in the above is as follows:

- Party A hires Trucking Company B to deliver Container X to Party C. Trucking Company B's ERP system asks Party C's ERP system at what time it should deliver the container. Party C's ERP system does not know Trucking Company B, but can check the delegation to Trucking Company B that Party A has registered at Authorization Registry D. Because this delegation is in order, Party C's ERP system shares a time slot with Trucking Company B's ERP.

Now imagine:

- Moments before Trucking Company B's ERP system asks Party C's ERP system for a time slot, Party C decides to revoke Party A's access to requesting a time slot. Consequently, Trucking Company B's request for a time slot gets an access forbidden message; Trucking Company B's request is NOT accepted because Party A, and therewith delegated Trucking Company B, is no longer authorised to ask for a time slot.

Prerequisites

To the prerequisites, ONLY the following changes:

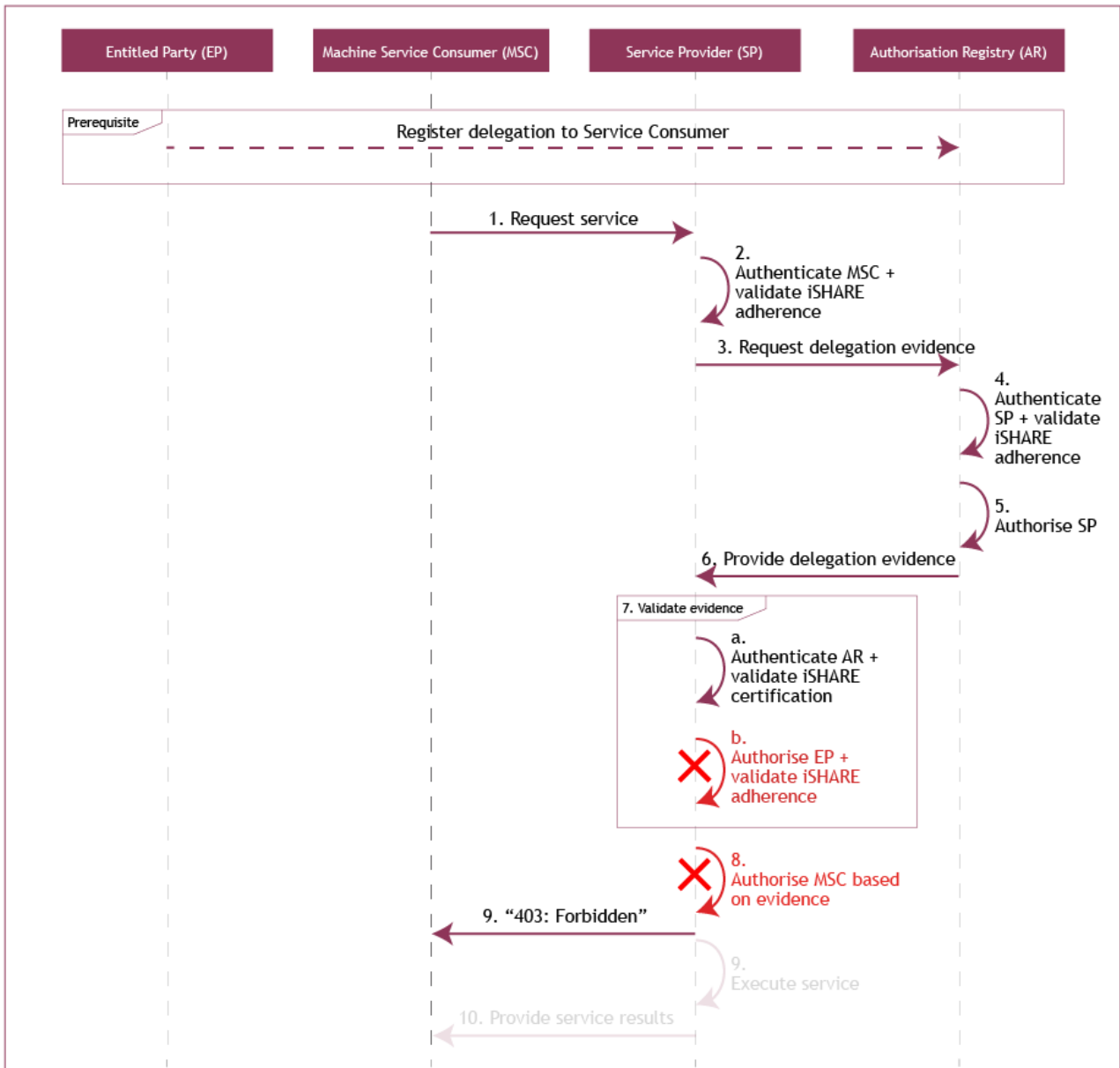
- The Service Provider (Party C) changes its entitlement information indicating what Entitled Parties are entitled to what (parts of) services, i.e. Party C deletes the information indicating that Party A is entitled to request a time slot.

Use case

The alternative use case consists of the following steps, with changes to the above use case in bold:

1. The Machine Service Consumer (of Trucking Company B) requests a service from the Service Provider (Party C);
2. The Service Provider (Party C) authenticates the Machine Service Consumer (of Trucking Company B) and validates the iSHARE adherence of the Service Consumer (Trucking Company B);
3. The Service Provider (Party C) requests delegation evidence from the Authorization Registry (D);
4. The Authorization Registry (D) authenticates the Service Provider (Party C) and validates its iSHARE adherence;
5. The Authorization Registry (D) authorizes the Service Provider (Party C) based on the scheme agreements for providing delegation information;
6. The Authorization Registry (D) provides the delegation evidence;
7. The Service Provider (Party C) validates the received delegation evidence through the following steps:
 - a. The Service Provider (Party C) authenticates the Authorization Registry (D) and validates its iSHARE certification;
 - b. **The Service Provider (Party C) CANNOT authorize the Entitled Party (Party A) based on the entitlement information registered with the Service Provider (Party C)**
8. **The Service Provider (Party C) CANNOT authorize the Machine Service Consumer of the Service Consumer (Trucking Company B) based on the validity of the delegation evidence;**
9. **The Service Provider (Party C) communicates an access forbidden message to the Machine Service Consumer (of Trucking Company B).**

Sequence diagram



What needs to be implemented technically for this use case (and the alternative scenario) is described [generally](#), and specifically per role in the iSHARE Developer Portal.

Detailed descriptions

This chapter provides an in depth overview of all the Functional, Technical, Operational and Legal details of the iSHARE Scheme. The following chapters are present in this section:

- Functional
 - Primary use cases
 - Secondary use cases
 - Licenses
 - Delegation paths
 - Functional requirements per role
- Technical
 - Generic technical standards
 - Structure of delegation evidence
- Operational
 - Operational processes
 - Service levels
 - Communication
- Legal
 - Legal context

Functional

This section details iSHARE's functionality.

The [use cases depicted in earlier chapters](#) are only a selection of iSHARE's full use case scope. This scope is based on [three 'primary' use cases](#):

1. Machine to Machine service provision;
2. Human to Machine service provision with authorization and identity info held at the Service Provider;
3. Human to Machine service provision with identity info held at the Identity Provider.

These primary use cases have several 'derived' use cases which cover all possible uses.

The primary use cases are supported by '[secondary' use cases](#), that include processes related to registration, and processes that recur in primary use cases. This section is concluded by functional requirements - those [per role in the scheme](#) and those to the iSHARE [user interface](#) in H2M use cases.

Primary use cases

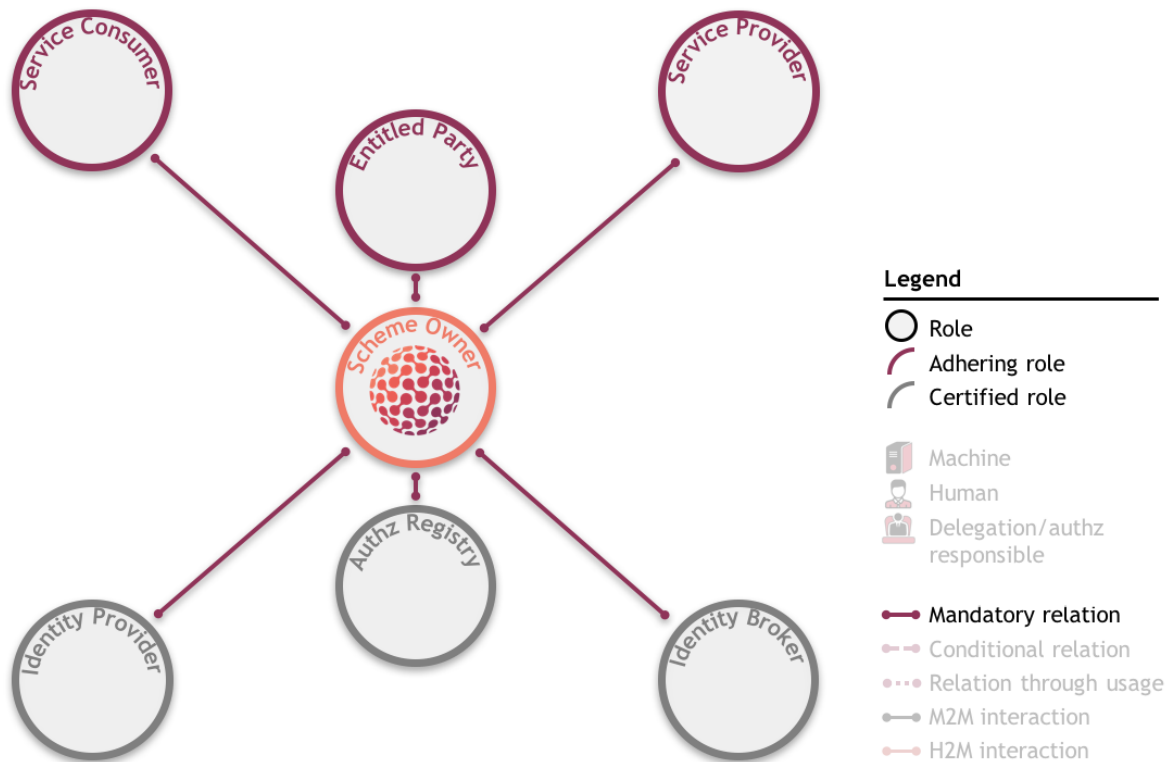
This most important part of the Functional descriptions explains the following in detail:

- The iSHARE framework, including the Scheme Owner and what role can hold what types of information;
- The three primary use cases: Machine to Machine, Human to Machine with authorization info and identity info held at the Service Provider, and Human to Machine with identity info held at an Identity Provider;
- The possible variations to the three primary use cases, depending on where identity information, authorization information and/or delegation information is held.

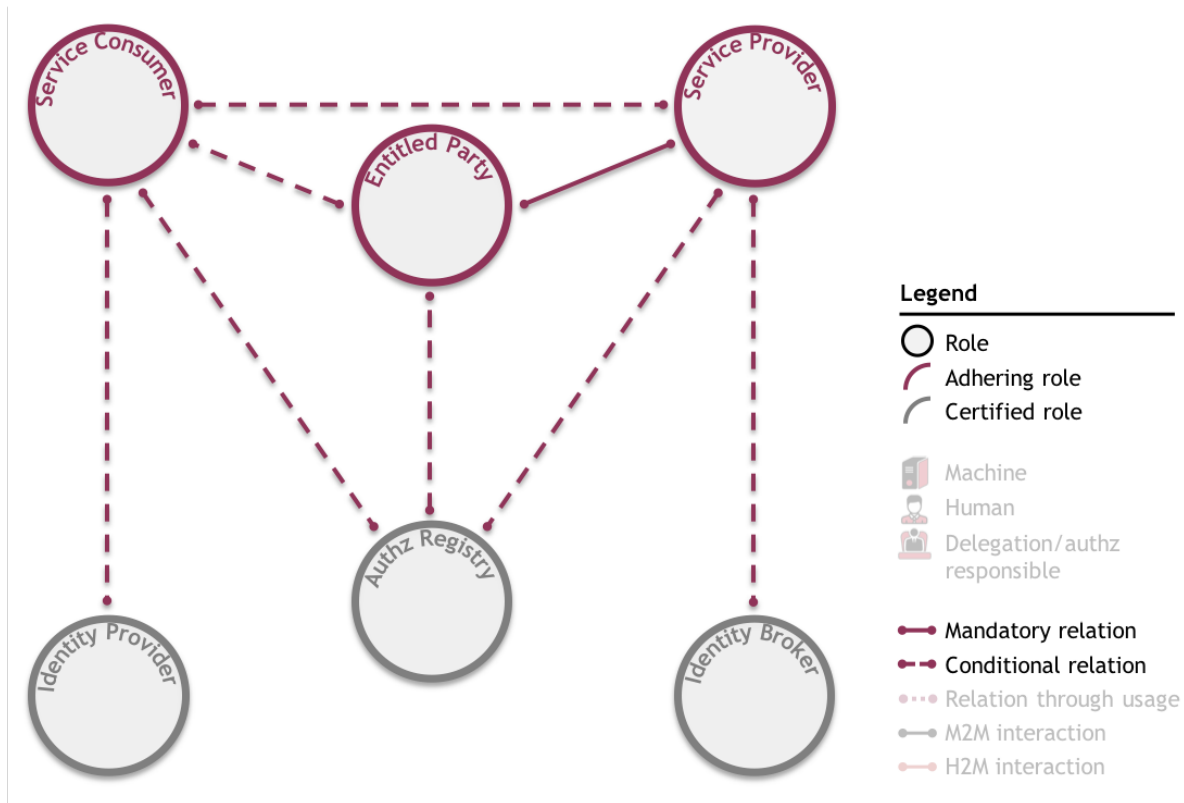
iSHARE framework

The iSHARE framework was explained under [use cases](#). It consists of six roles that, depending on the situation, interact with each other based on the iSHARE Scheme agreements. Each role has a certain function in the scheme and bears certain responsibilities. To fulfil an other role in the framework, a party must fulfil specific admittance criteria, as explained.

What was not explained under use cases was how the iSHARE and the Scheme Owner-role provide a trust framework. The Scheme Owner-role is fulfilled by the legal entity that governs the iSHARE Scheme and its participant network. It is this Scheme Owner that decides whether a party is admitted to the iSHARE network. To be admitted, this party must sign an agreement with the Scheme Owner. The fact that every legal entity fulfilling a role in the iSHARE Scheme agrees to the scheme rules - as proven by its agreement with the Scheme Owner - creates trust between parties in the iSHARE network. This is why the following depiction of the iSHARE framework, showing the mandatory relation between the Scheme Owner and every other role, can be called the **trust framework**:



In order to know whether a party is an iSHARE participant before sharing data with it, the Scheme Owner can be asked about this party's adherence/certification (as detailed in [secondary use case 5a](#)). This and the trust framework as a whole are not reflected in the primary use cases because *every relation or interaction* within iSHARE is build upon the trust framework. The framework used to depict use cases was already presented as follows:



It was stated that all of iSHARE's use cases can be depicted in the above framework.

Their complexity is dependent on:

- The interaction model (Machine to Machine or Human to Machine); i.e. whether the Service Consumer is represented by a machine or a human.
- Whether delegation takes place, and; i.e. whether the Service Consumer-role is fulfilled by another entity than the Entitled Party-role. How delegations work exactly is explained [here](#).
- Whether parties fulfilling adhering roles use their own tooling for identification, authentication, and authorization or outsource these processes and the information necessary for these processes to certified roles instead.

Zooming in on the latter, four types of information are recognised that are needed to facilitate [identification, authentication and authorization](#):

- **Entitlement info:** information indicating what Entitled Parties are entitled to what (parts of) services;
- **Delegation info:** information indicating which (parts of) an Entitled Party's rights (as registered at the Service Provider or the Authorization Registry) are delegated to a Service Consumer;

- **Authorization info:** information indicating which Human Service Consumers are authorized to act on a Service Consumer's behalf;
- **Identity info:** information about a Human Service Consumer's identity (only applicable in H2M use cases).

All complexity can be brought back to three primary use cases, with 21 variations.

Three primary use cases

1. Machine to Machine service provision;
Primary use case 1 caters to all Machine to Machine cases.
2. Human to Machine service provision with authorization and identity info held at the Service Provider;
Primary use case 2 caters to all Human to Machine cases where the Service Provider resides over both identity information and authorization information. He has not outsourced identification, authentication and authorization, and therefore does not need to consult certified parties
3. Human to Machine service provision with identity info held at the Identity Provider.
Primary use case 3 caters to all Human to Machine cases where identity information is held at an Identity Provider. The Service Provider has outsourced identification and authentication, and therefore needs to consult the Identity Provider.

Derived use cases

The primary use cases all know a variety of derived use cases. Derived use cases are variations of the primary use cases in which delegation- and authorization information required by the Service Provider is held by (i.e. outsourced to) and retrieved from different parties. In technical terms, we call the party holding information a **Policy Information Point (PIP)**. This PIP, as in [XACML 3.0](#), acts as the source of the information. There are different use case variations for different PIPs for delegation- and/or authorization information, as presented in the use case tables below. Note that entitlement info is always held by the Service Provider which is (consequently) not depicted in the tables below.

The Service Provider requests (from the PIP(s)) and evaluates the information required to decide whether or not to grant a Service Consumer access to a service. After making its decision based on the received information, it grants this access (or not) to the Service Consumer. Technically, the Service Provider therefore acts as **Policy Enforcement Point (PEP)** and **Policy Decision Point (PDP)** in all use cases.

Primary use case 1 (and derived use cases)*: M2M service provision

Use case initiated by the Machine Service Consumer

	Delegation info PIP			
	<i>No delegation</i>	Service Provider	Entitled Party	Authorization Reg
Derived use cases**	1	1a	1b	1c

*Use case 1 and its variations can also be initiated by a Human Service Consumer through an app. In such case, the Machine Service Consumer acts as a proxy between the Human Service Consumer and the Service Provider's machine as described [here](#).

**Primary use case 1 assumes that authorization information is always present in a valid token used by the Machine Service Consumer. Therefore primary use case 1 has no derived use cases where authorization information is retrieved from other parties.

Note that interaction sequences are not described in the table above. In derived use cases 1b and 1c, several interaction sequences are possible depending on who requests delegation info from the PIP. If the Entitled Party is the delegation info PIP:

1. The Service Provider can request delegation info after a service request from the Service Consumer;
2. The Machine Service Consumer can request delegation info and include it in its service request to the Service Provider;
3. The Entitled Party can push delegation info to the Machine Service Consumer, so it can include it in its service request to the Service Provider.

If the Authorization Registry is the delegation info PIP:

1. The Service Provider can request delegation info after a service request from the Service Consumer;
2. The Machine Service Consumer can request delegation info and include it in its service request to the Service Provider.

Use case 1 only has one interaction pattern as there is no delegation info PIP. Derived use case 1a also has one interaction pattern as the Service Provider is the Delegation info PIP and therefore already has the delegation info it needs.

Primary use case 2 (and derived use cases): H2M service provision with authorization info and identity info held at the SP

Use case initiated by the Human Service Consumer

		Delegation info PIP			
		<i>No delegation</i>	S e r v i c e P r o v i d e r	E n t i t l e d P a r t y	A u t h o r i z a t i o n R e g i s t r y
Auth info PIP	Service Provider	2	2a	2b	2c

Primary use case 3 (and derived use cases): H2M service provision with identity info held at the IDP

Use case initiated by the Human Service Consumer

		Delegation info PIP			
		<i>No delegation</i>	S e r v i c e P r o v i d e r	E n t i t l e P r o t y	A u t h o r i z a t i o n R e g i s t r y
Auth info PIP	Identity Provider	3	3 a	3 b	3 c

Note again that interaction sequences are not described in the tables above. A Human Service Consumer cannot include delegation (or authorization) info in its service request to the Service Provider. In use cases 2 and 3 (and derived use cases), therefore, the Service Provider will always request delegation- and/or authorization info from the respective PIP(s) after a service request from the Human Service Consumer.

Several interaction sequences are still theoretically possible depending on who requests a login from the Identity Provider. During the Functional working groups, however, it appeared that in practice, a Human Service Consumer will never request login from an Identity Provider before requesting a service from the Service Provider. Until proven otherwise, therefore, the only interaction sequence in scope for use cases 2 and 3 (and derived use cases) is the one in which the Service Provider (also) requests login from the Identity Provider after a service request from the Human Service Consumer.

In use case 3 (and derived use cases), an Identity Broker can be introduced to broker the relation between the Service Provider and the Identity Provider(s) and/or the Service Provider and the Authorization Registry(s). This is optional and useful in situations with several Identity Providers and/or Authorization Registries. [Use case 3](#) is detailed both without an Identity Broker and with one.

Rest of this section

Please note that all use cases that contain a hyperlink (in their respective tables) are detailed on their own page - as follows:

- Roles;
- Depiction of legal relations, prerequisite registration and use case interaction;
- Description of prerequisites and use case interaction;
- Sequence diagram.

For both use case 2 and 3 (and derived use cases), an interface is required. Requirements to this interface are summarised [here](#).

1. M2M service provision

In use case 1, a service is provided by the Service Provider to the Machine Service Consumer.

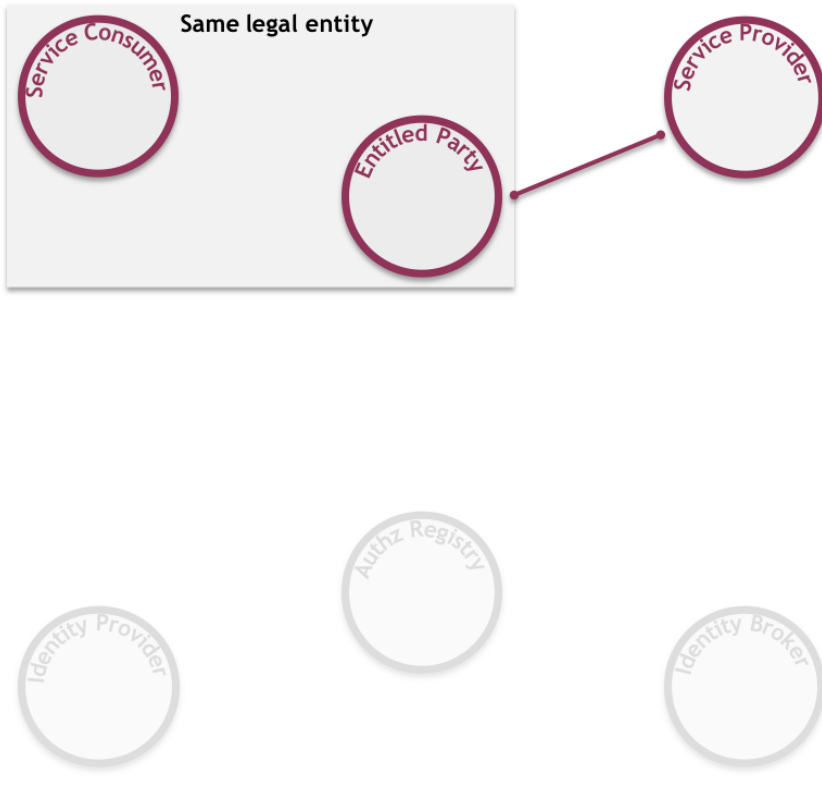
Roles

	<p>Delegation info PIP</p> <p><i>No delegation</i></p>	S e r v i c e P r o v i d e r	E n t i t l e P a r t y	A u t h o r i z a t i o n R e g
Use case variation	1. M2M service provision	1 a	1 b	1 c

As no delegation takes place, the legal entity fulfilling the Entitled Party-role also fulfils the Service Consumer-role.

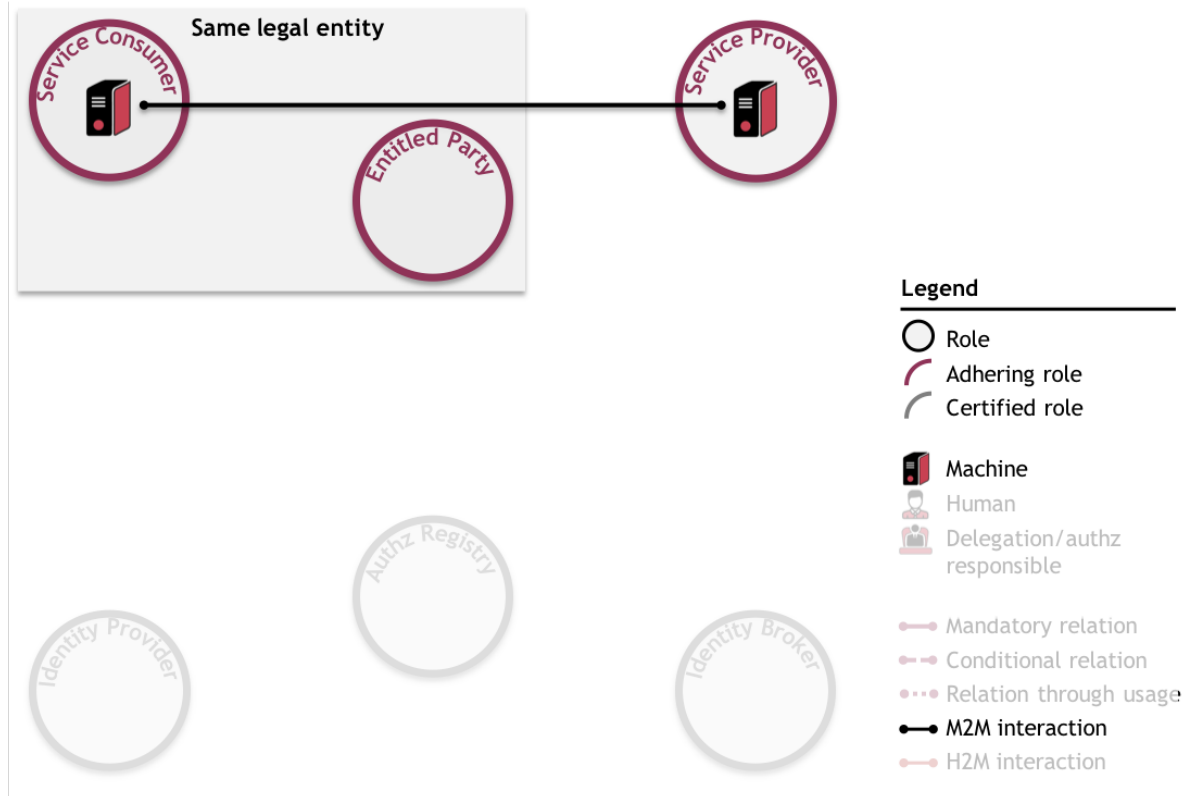
Depiction

Legal relations



Legend

- Role
- ◌ Adhering role
- ◌ Certified role
- Machine
- Human
- Delegation/authz responsible
- Mandatory relation
- - - Conditional relation
- ... Relation through usage
- M2M interaction
- H2M interaction

Use case interaction**Description****It is prerequisite of this use case that:**

- The Service Provider has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services*;
- The Service Consumer is able to authenticate the Service Provider;
- The Service Provider is able to authenticate the Service Consumer.

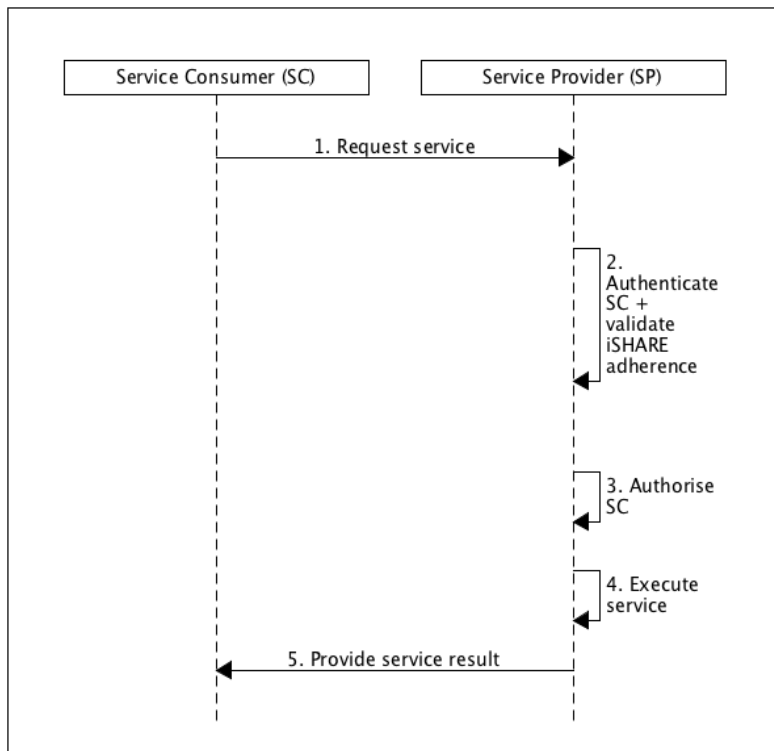
- In this use case the Entitled Party is also the Service Consumer.

*The Service Provider can outsource this function to a third party

The use case consists of the following steps:

1. The Machine Service Consumer requests a service from the Service Provider;
2. The Service Provider authenticates the Machine Service Consumer and validates the iSHARE adherence of the Service Consumer;
3. The Service Provider authorizes the Machine Service Consumer of the Service Consumer based on the entitlement information registered with the Service Provider;
4. The Service Provider executes the requested service;
5. The Service Provider provides the service result to the Machine Service Consumer.

Sequence diagram



1b. M2M service provision with the EP as the delegation info PIP

In use case 1b, a service is provided by the Service Provider to the Machine Service Consumer. The Service Consumer has been delegated by the Entitled Party.

Roles

	Delegation info PIP			
	<i>No delegation</i>	Service Provider	Entitled Party	Authorization Reg
Use case variation	1	1a	1b	1c

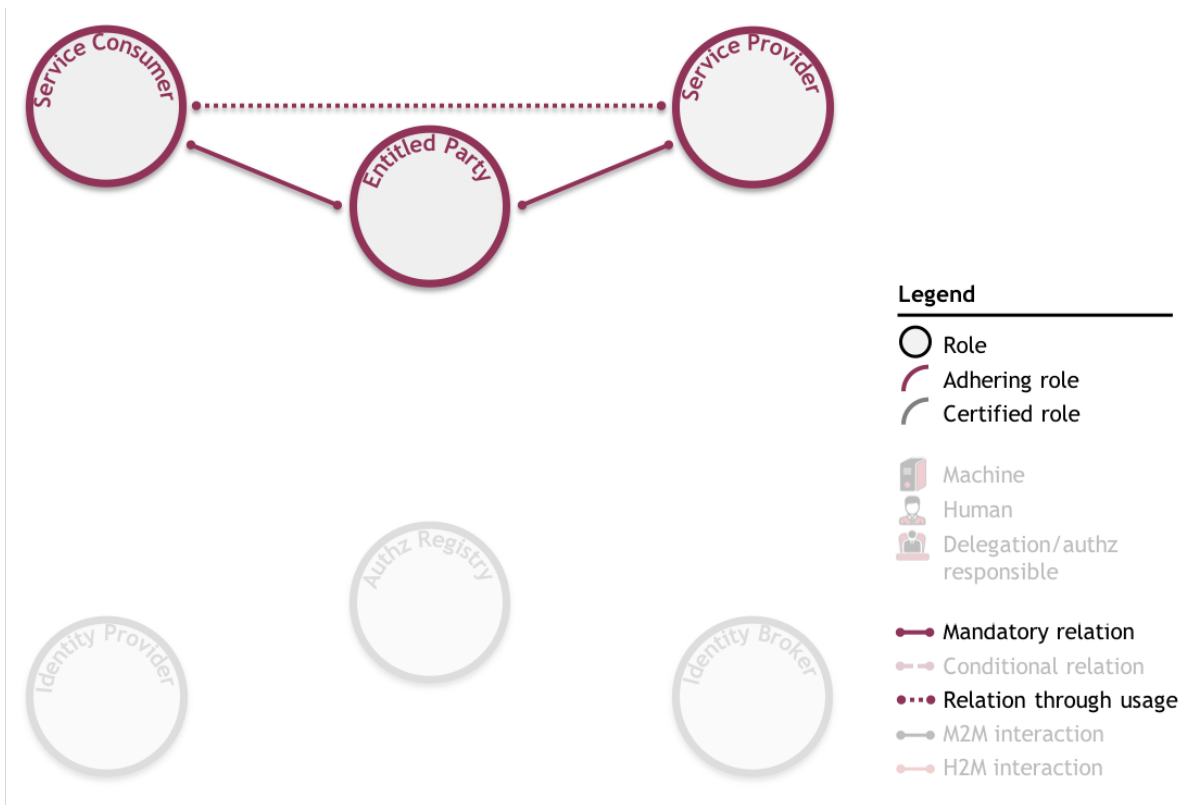
Note that interaction sequences are not described in the table above. In derived use case 1b, three interaction sequences are possible depending on who requests delegation info from the PIP:

1. The Service Provider can request delegation info after a service request from the Service Consumer;
2. The Machine Service Consumer can request delegation info and include it in its service request to the Service Provider;
3. The Entitled Party can push delegation info to the Machine Service Consumer, so it can include it in its service request to the Service Provider.

Interaction sequence 3 is detailed below.

Depiction

Legal relations



Note that no prior legal relation exists between the Service Consumer and the Service Provider. Which services can be consumed by the Service Consumer, as delegated by the Entitled Party, is set out in the mandatory relation between this Entitled Party and the Service Provider.

Prerequisite registration



Legend

- Role
- ◡ Adhering role
- ◡ Certified role
- 🖨 Machine
- 👤 Human
- 👤 Delegation/ authz responsible
- Mandatory relation
- - - Conditional relation
- ⋯ Relation through usage
- M2M interaction
- H2M interaction

Use case interaction



Legend

- Role
- ◡ Adhering role
- ◡ Certified role
- 🖨 Machine
- 👤 Human
- 👤 Delegation/ authz responsible
- Mandatory relation
- - - Conditional relation
- ⋯ Relation through usage
- M2M interaction
- H2M interaction

Description

It is prerequisite of this use case that:

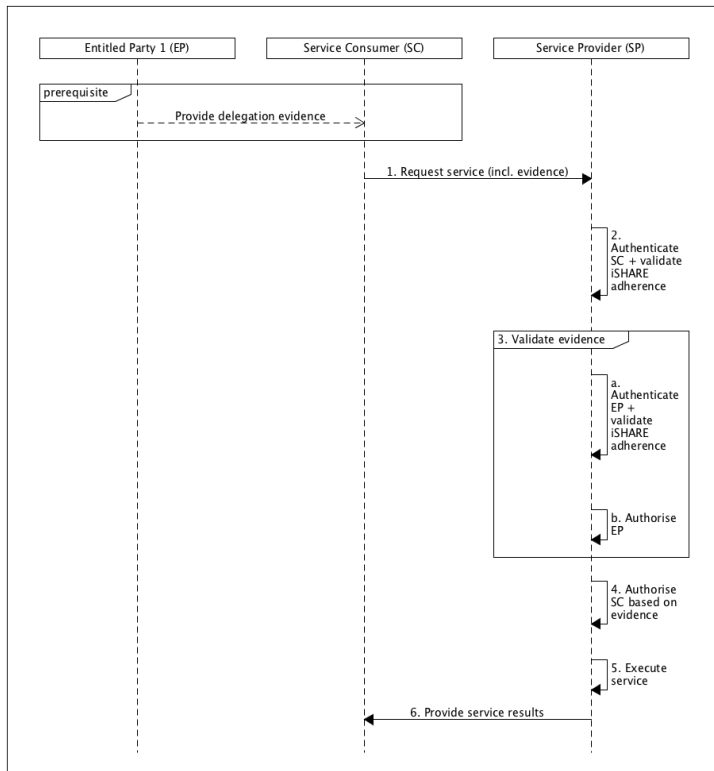
- The Service Provider has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services*;
- The Service Consumer is able to authenticate the Service Provider;
- The Service Provider is able to authenticate the Service Consumer;
- The delegation/authorization responsible at the Entitled Party delegates (part of) the Entitled Party's rights (as registered at the Service Provider) to the Service Consumer. He provides the Machine Service Consumer of the Service Consumer with evidence of this delegation.

*The Service Provider can outsource this function to a third party

The use case consists of the following steps:

1. The Machine Service Consumer requests a service from the Service Provider. With this requests it includes the evidence obtained from the Entitled Party;
2. The Service Provider authenticates the Machine Service Consumer and validates the iSHARE adherence of the Service Consumer;
3. The Service Provider validates the received delegation evidence through the following steps:
 - a. The Service Provider authenticates the Entitled Party and validates its iSHARE adherence based on the delegation evidence;
 - b. The Service Provider authorizes the Entitled Party based on the entitlement information registered with the Service Provider.
4. The Service Provider authorizes the Machine Service Consumer of the Service Consumer based on the validity of the delegation evidence;
5. The Service Provider executes the requested service;
6. The Service Provider provides the service result to the Machine Service Consumer.

Sequence diagram



1c. M2M service provision with the AR as the delegation info PIP

In use case 1c, a service is provided by the Service Provider to the Service Consumer. The Service Consumer has been delegated by the Entitled Party, and delegation evidence is registered at an Authorization Registry.

Roles

	Delegation info PIP			
	<i>No delegation</i>	Service Provider	Entitled Party	Authorization Reg
Use case variation	1	1a	1b	1c

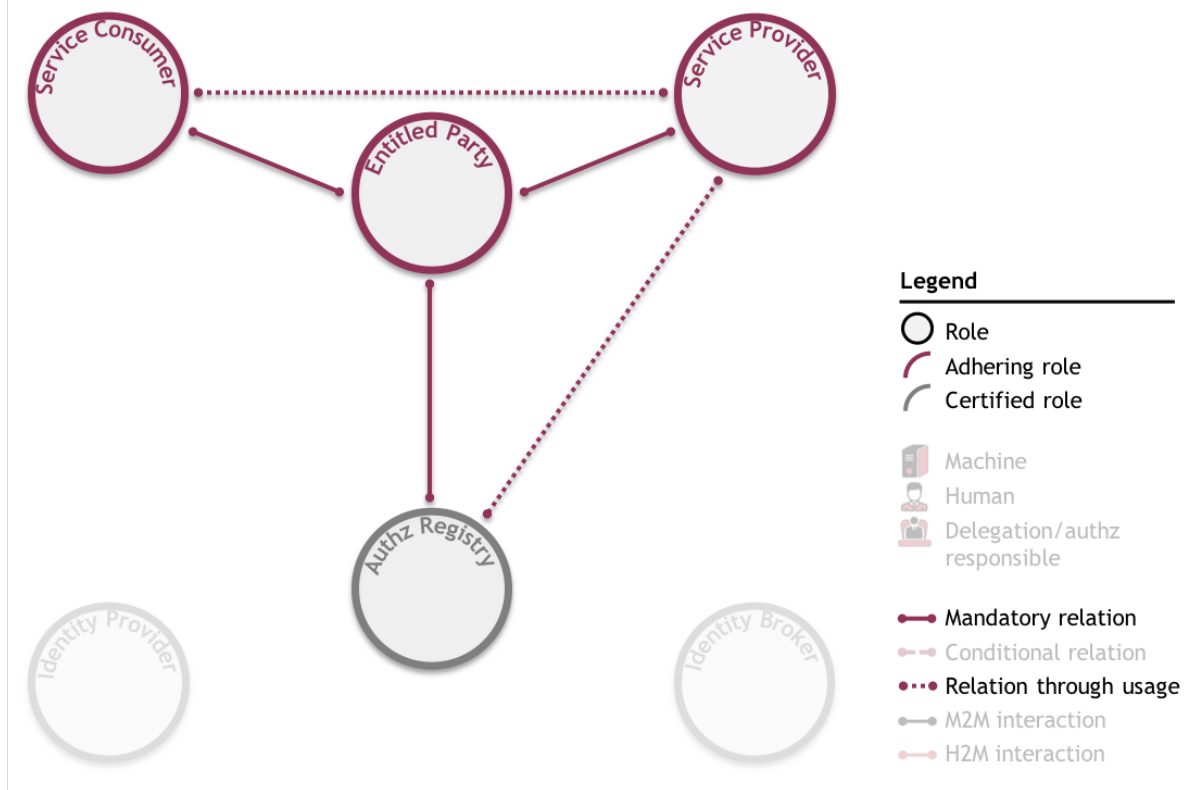
Note that interaction sequences are not described in the table above. In derived use case 1c, two interaction sequences are possible depending on who requests delegation info from the PIP:

1. The Service Provider can request delegation info after a service request from the Service Consumer;
2. The Machine Service Consumer can request delegation info and include it in its service request to the Service Provider.

Interaction sequence 1 is detailed below.

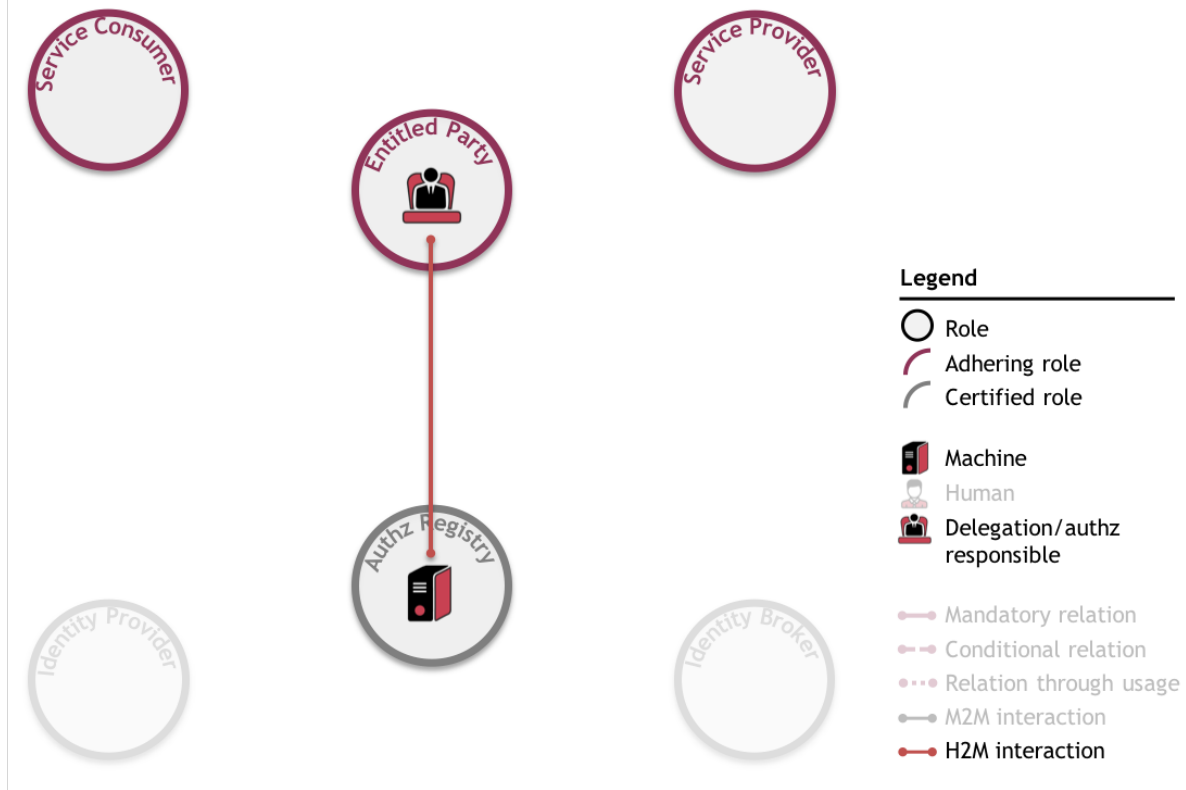
Depiction

Legal relations

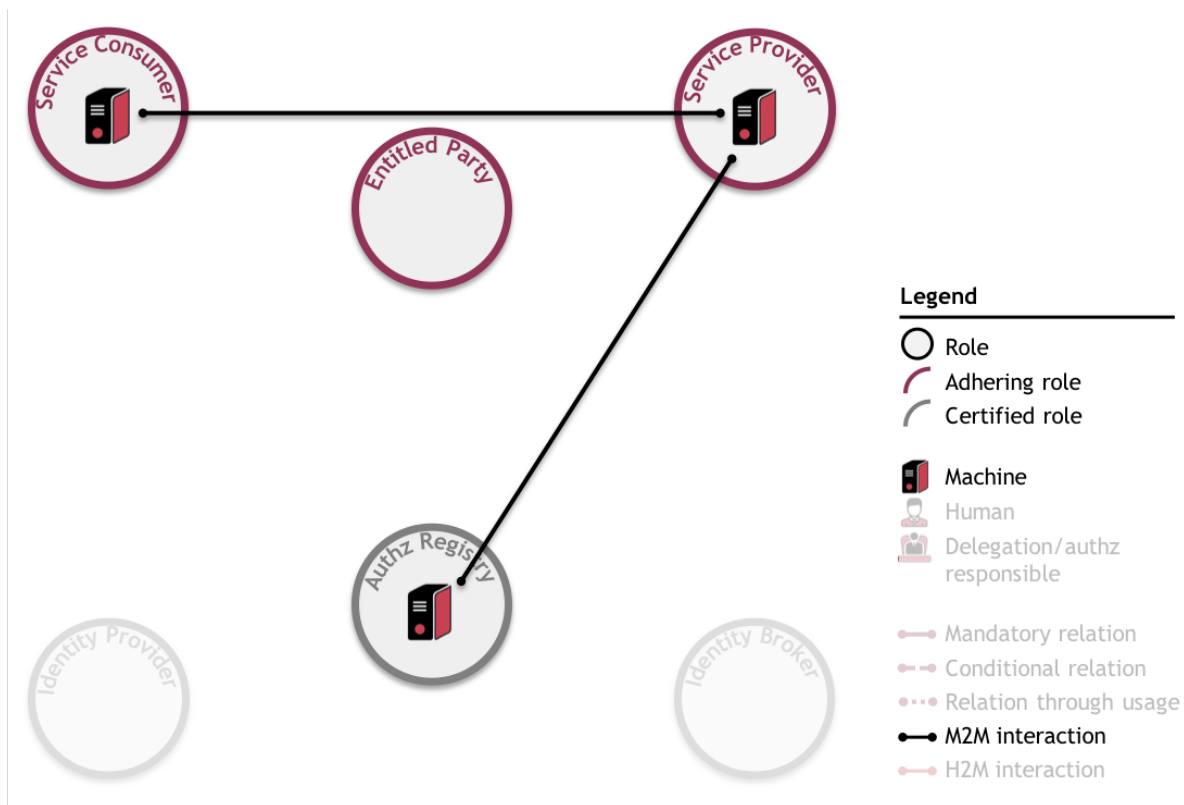


Note that no prior legal relation exists between the Service Consumer and the Service Provider. Which services can be consumed by the Service Consumer, as delegated by the Entitled Party, is set out in the mandatory relation between this Entitled Party and the Service Provider.

Prerequisite registration



Use case interaction



Description

It is prerequisite of this use case that:

- The Service Provider has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services*;
- The Service Consumer is able to authenticate the Service Provider;
- The Service Provider is able to authenticate the Service Consumer;
- The delegation/authorization responsible at the Entitled Party delegates (part of) the Entitled Party's rights (as registered at the Service Provider) to the Service Consumer. He registers this delegation in an Authorization Registry;
- The Service Provider knows which Authorization Registry to request the delegation evidence from;
- The Service Provider is able to authenticate the Authorization Registry;
- The Authorization Registry is able to authenticate the Service Provider;
- It is clear, through scheme agreements, under what conditions an Authorization Registry can provide delegation information to a Service Provider.

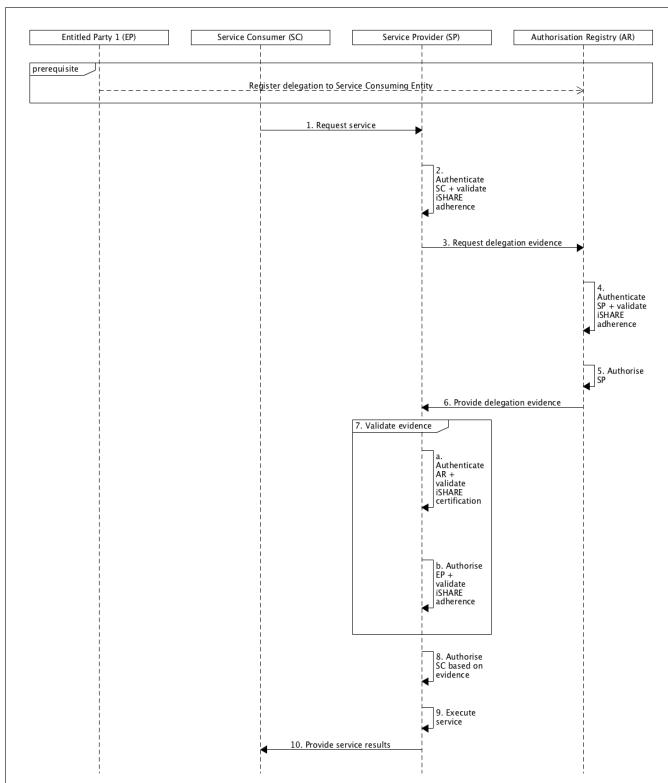
*The Service Provider can outsource this function to a third party

The use case consists of the following steps:

1. The Machine Service Consumer requests a service from the Service Provider;
2. The Service Provider authenticates the Machine Service Consumer and validates the iSHARE adherence of the Service Consumer;
3. The Service Provider requests delegation evidence from the Authorization Registry;
4. The Authorization Registry authenticates the Service Provider and validates its iSHARE adherence;
5. The Authorization Registry authorizes the Service Provider based on the scheme agreements for providing delegation information;

6. The Authorization Registry provides the delegation evidence;
7. The Service Provider validates the received delegation evidence through the following steps:
 - a. The Service Provider authenticates the Authorization Registry and validates its iSHARE certification;
 - b. The Service Provider authorizes the Entitled Party based on the entitlement information registered with the Service Provider, and validates its iSHARE adherence.
8. The Service Provider authorizes the Machine Service Consumer of the Service Consumer based on the validity of the delegation evidence;
9. The Service Provider executes the requested service;
10. The Service Provider provides the service result to the Machine Service Consumer.

Sequence diagram



M2M service provision including an app

Use case 1 and its variations can be initiated by a Human Service Consumer through an app. In such case, the Machine Service Consumer acts as a proxy between the Human Service Consumer and the Service Provider's machine.

Roles

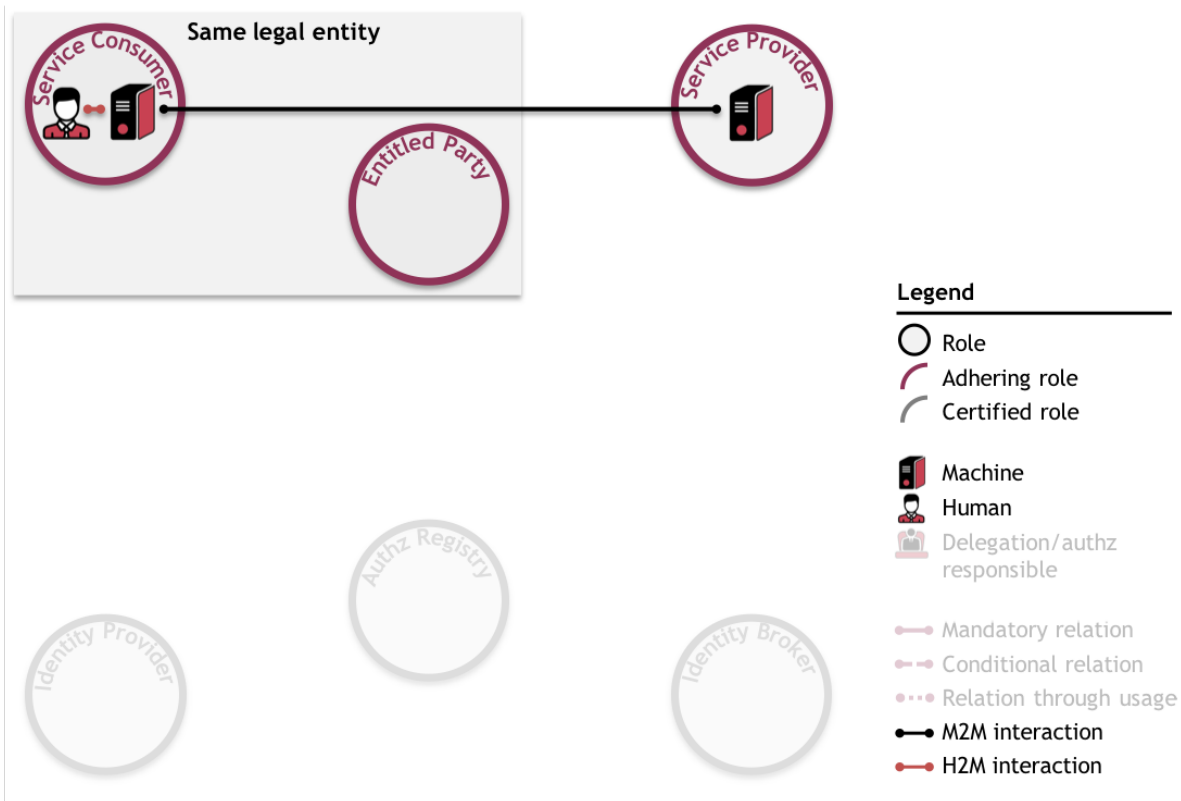
	Delegation info PIP			
	No delegation	Service Provider	Entitled Party	Authorization Reg
Use case variation	1	1a	1b	1c

Depiction

Legal relations



Use case interaction



Description

As to use case 1, it is prerequisite of this use case that:

- The Service Provider has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services*;
- The Service Consumer is able to authenticate the Service Provider;
- The Service Provider is able to authenticate the Service Consumer.

- In this use case the Entitled Party is also the Service Consumer.

*The Service Provider can outsource this function to a third party

The use case consists of the following steps:

- The Human Service Consumer uses an app to request a service at the Machine Service Consumer - the Human Service Consumer's identity is included in the request;
 - The request is mapped to a service request;
1. The Machine Service Consumer requests a service from the Service Provider;
 2. The Service Provider authenticates the Machine Service Consumer and validates the iSHARE adherence of the Service Consumer;
 3. The Service Provider authorizes the Machine Service Consumer of the Service Consumer based on the entitlement information registered with the Service Provider;
 4. The Service Provider executes the requested service;
 5. The Service Provider provides the service result to the Machine Service Consumer;
- The Human Service Consumer accesses the result through app.

2. H2M service provision with identity info at the SP

In use case 2, a service is provided by the Service Provider to the Human Service Consumer. Identity info is held at the Service Provider.

Roles

		Delegation info PIP			
		<i>No delegation</i>	S e r v i c e P r o v i d e r	E n t i t l e d P a r t y	A u t h o r i z a t i o n R e g
Auth info PIP	Service Provider	2. H2M service provision with identity info at the SP	2 a	2 b	2 c

As no delegation takes place, the legal entity fulfilling the Entitled Party-role also fulfils the Service Consumer-role.

Depiction

Legal relations



Legend

- Role
- ◌ Adhering role
- ◌ Certified role
- Machine
- Human
- Delegation/authz responsible
- Mandatory relation
- - - Conditional relation
- ⋯ Relation through usage
- M2M interaction
- H2M interaction



Prerequisite registration



Legend

- Role
- ◡ Adhering role
- ◡ Certified role
- Machine
- Human
- Delegation/authz responsible
- Mandatory relation
- - - Conditional relation
- ... Relation through usage
- M2M interaction
- H2M interaction



Use case interaction



Description

It is prerequisite of this use case that:

- The Service Provider has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services*;
 - The Service Consumer has and manages its own authorization information indicating which Human Service Consumers are authorized to act on its behalf**;
 - The delegation/authorization responsible at the the Service Consumer registers the authorization information at the Service Provider;
 - The Human Service Consumer is able to authenticate the Service Provider;
 - The Service Provider is able to authenticate the Human Service Consumer;
 - The Human Service Consumer has been issued identity credentials by the Service Provider.
- In this use case the Entitled Party is also the Service Consumer.

*The Service Provider can outsource this function to a third party

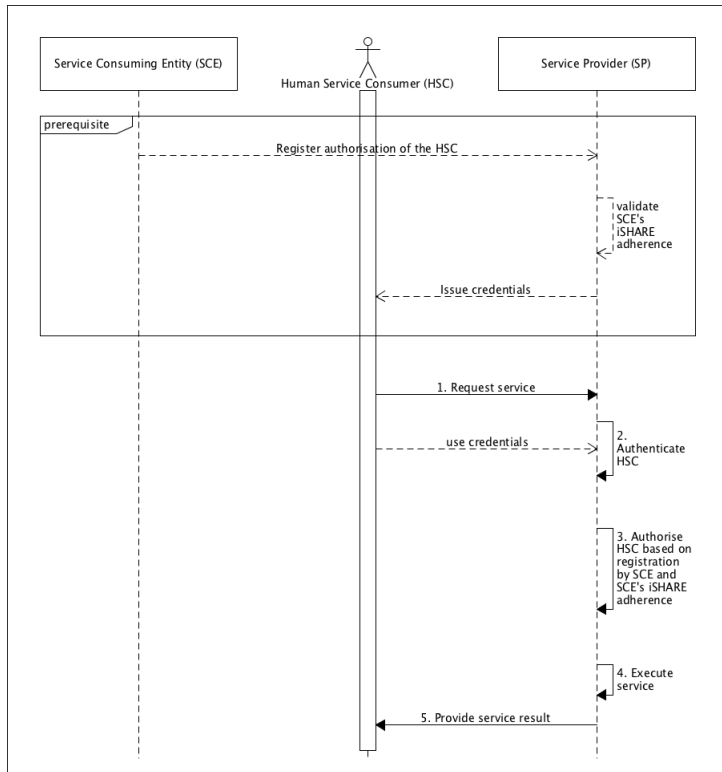
**The Service Consumer can outsource this function to a third party

The use case consists of the following steps:

1. The Human Service Consumer requests a service from the Service Provider;
2. The Service Provider authenticates the Human Service Consumer, and validates the iSHARE adherence of the Service Consumer;
3. The Service Provider authorizes the Human Service Consumer of the Service Consumer based on the entitlement- and authorization information registered with the Service Provider;;

4. The Service Provider executes the requested service;
5. The Service Provider provides the service result to the Human Service Consumer.

Sequence diagram



3. H2M service provision with identity info at the IP

In use case 3, a service is provided by the Service Provider to the Human Service Consumer. Identity info is held at the Identity Provider.

Roles

		Delegation info PIP			
		<i>N</i>	<i>S</i>	<i>E</i>	<i>A</i>
		<i>o</i>	<i>r</i>	<i>n</i>	<i>u</i>
		<i>d</i>	<i>v</i>	<i>t</i>	<i>t</i>
		<i>e</i>	<i>i</i>	<i>e</i>	<i>h</i>
		<i>l</i>	<i>c</i>	<i>d</i>	<i>o</i>
		<i>e</i>	<i>e</i>	<i>P</i>	<i>z</i>
		<i>a</i>	<i>P</i>	<i>a</i>	<i>a</i>
		<i>t</i>	<i>r</i>	<i>r</i>	<i>t</i>
		<i>o</i>	<i>o</i>	<i>t</i>	<i>o</i>
		<i>n</i>	<i>v</i>	<i>y</i>	<i>n</i>
			<i>i</i>		<i>R</i>
			<i>d</i>		<i>e</i>
			<i>e</i>		<i>g</i>
			<i>r</i>		
Auth info PIP	Identity Provider	3	3	3	3
		.	a	b	c

As no delegation takes place, the legal entity fulfilling the Entitled Party-role also fulfils the Service Consumer-role.

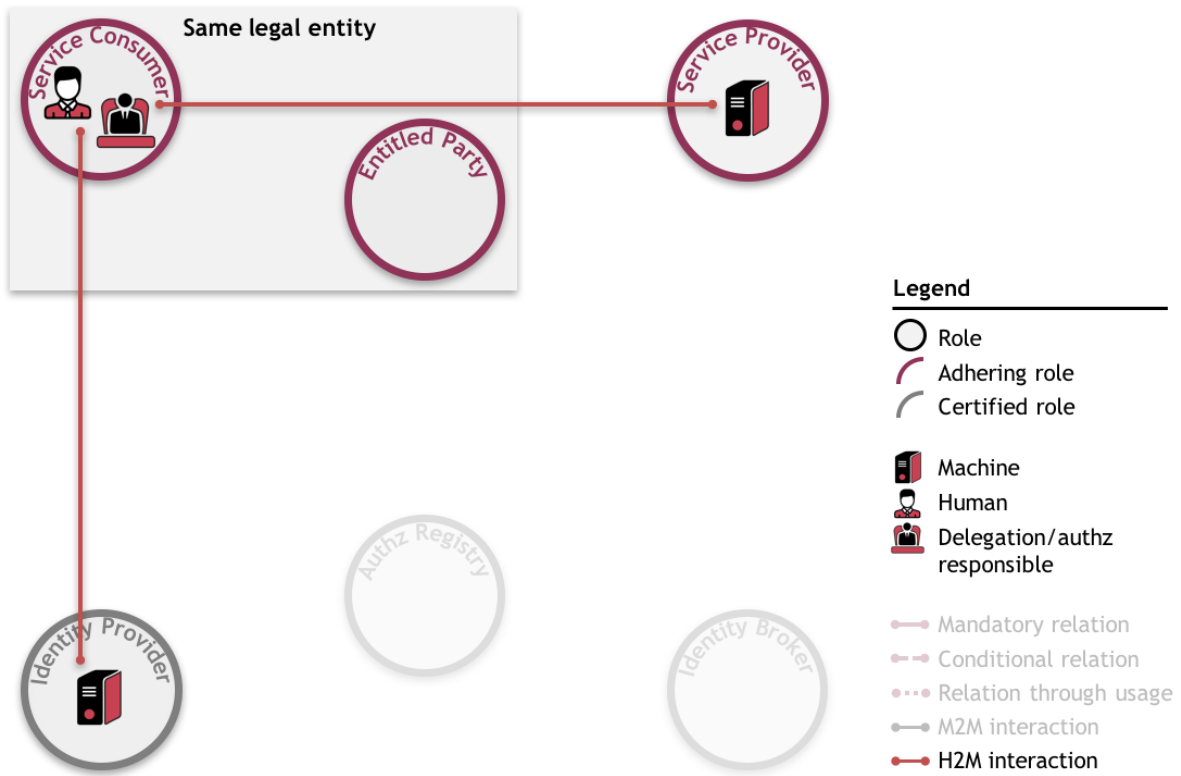
Note that an Identity Broker can be introduced to broker the relation between the Service Provider and the Identity Provider(s). This is optional and useful in situations with several Identity Providers.

Depiction without Identity Broker

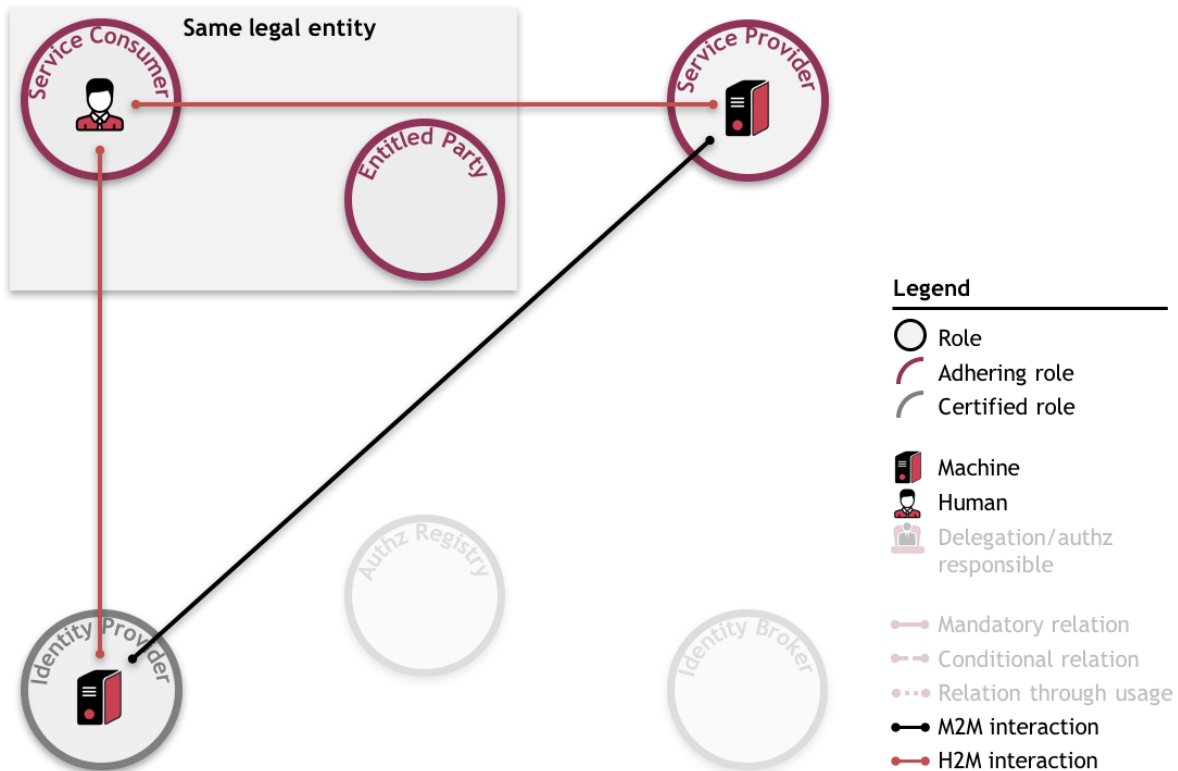
Legal view



Prerequisite registration



Interaction



Description without Identity Broker

It is prerequisite of this use case that:

- The Service Provider has and manages its own entitlement information indicating what Entitled Parties are entitled to what (parts of) services*;
 - The Service Consumer has and manages its own authorization information indicating which Human Service Consumers are authorized to act on its behalf**;
 - The Service Consumer registers the authorization information at the Identity Provider;
 - The Human Service Consumer is able to authenticate the Service Provider;
 - The Service Provider is able to authenticate the Human Service Consumer;
 - The Identity Provider is able to authenticate the Service Provider;
 - The Service Provider is able to authenticate the Identity Provider;
 - The Human Service Consumer has been issued identity credentials by the Identity Provider.
- In this use case the Entitled Party is also the Service Consumer.

*The Service Provider can outsource this function to a third party

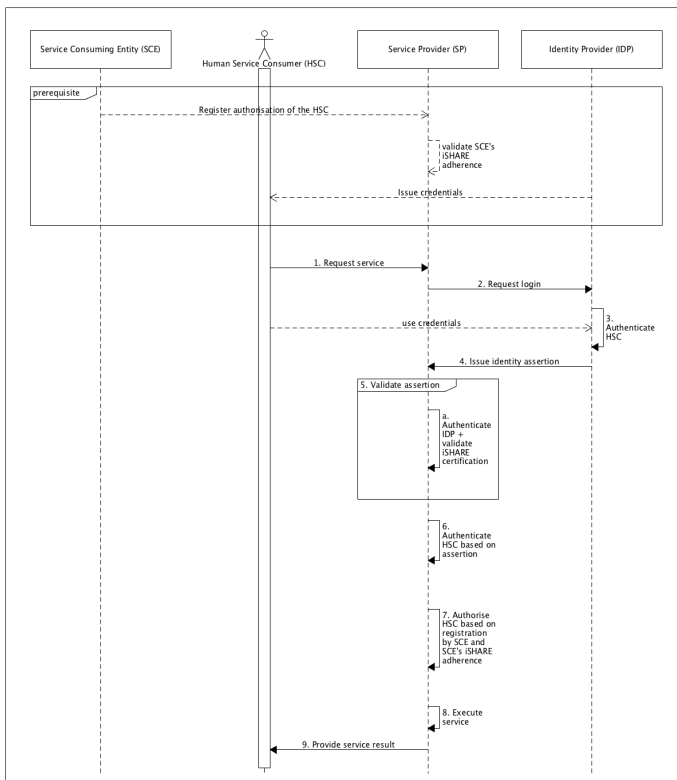
**The Service Consumer can outsource this function to a third party

The use case consists of the following steps:

1. The Human Service Consumer requests a service from the Service Provider;
2. The Service Provider asks the Human Service Consumer to select his Identity Provider;
3. The Service Provider requests a login from the Identity Provider;
4. The Identity Provider authenticates the Human Service Consumer;

5. The Identity Provider issues an identity assertion and authorization assertion to the Service Provider;
6. The Service Provider validates the identity assertion and authorization assertion through the following steps:
 - a. The Service Provider authenticates the Identity Provider and validates its iSHARE certification.
7. The Service Provider authenticates the Human Service Consumer based on the validity of the identity assertion, and validates the iSHARE adherence of the Service Consumer;
8. The Service Provider authorizes the Human Service Consumer of the Service Consumer based on the authorization assertion and the entitlement information registered with the Service Provider;
9. The Service Provider executes the requested service;
10. The Service Provider provides the service result to the Human Service Consumer.

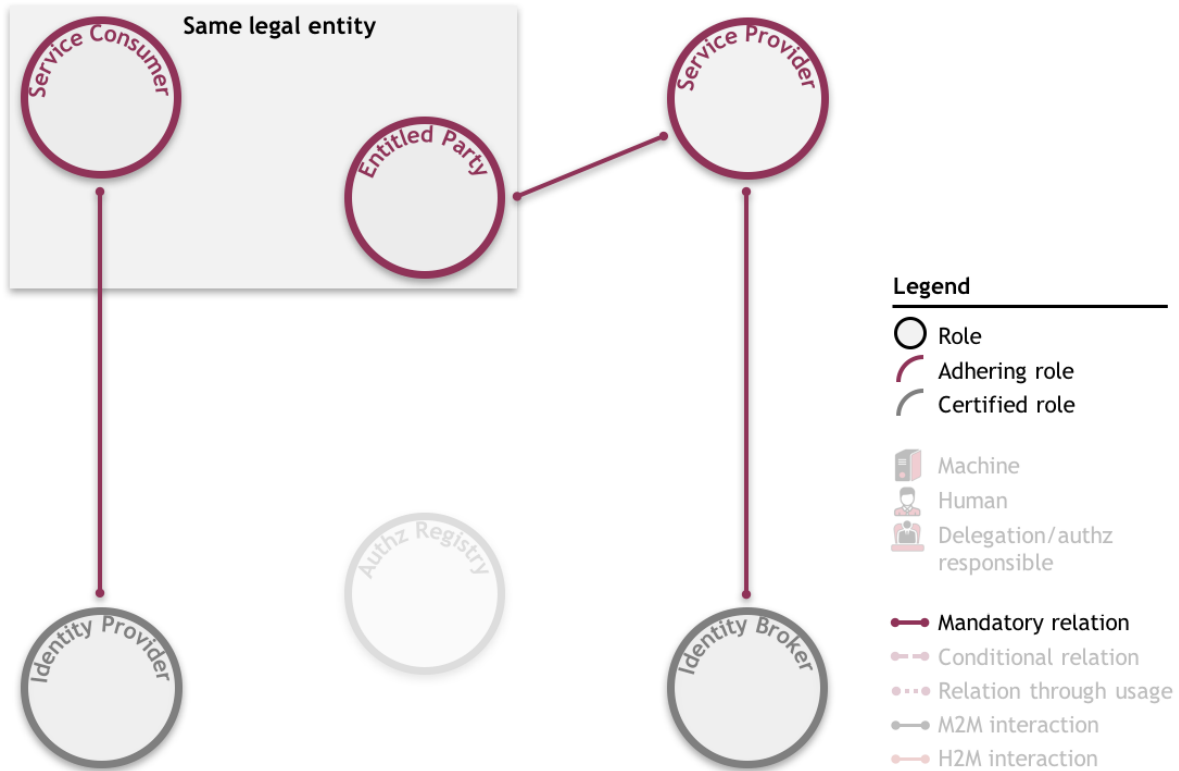
Sequence diagram without Identity Broker



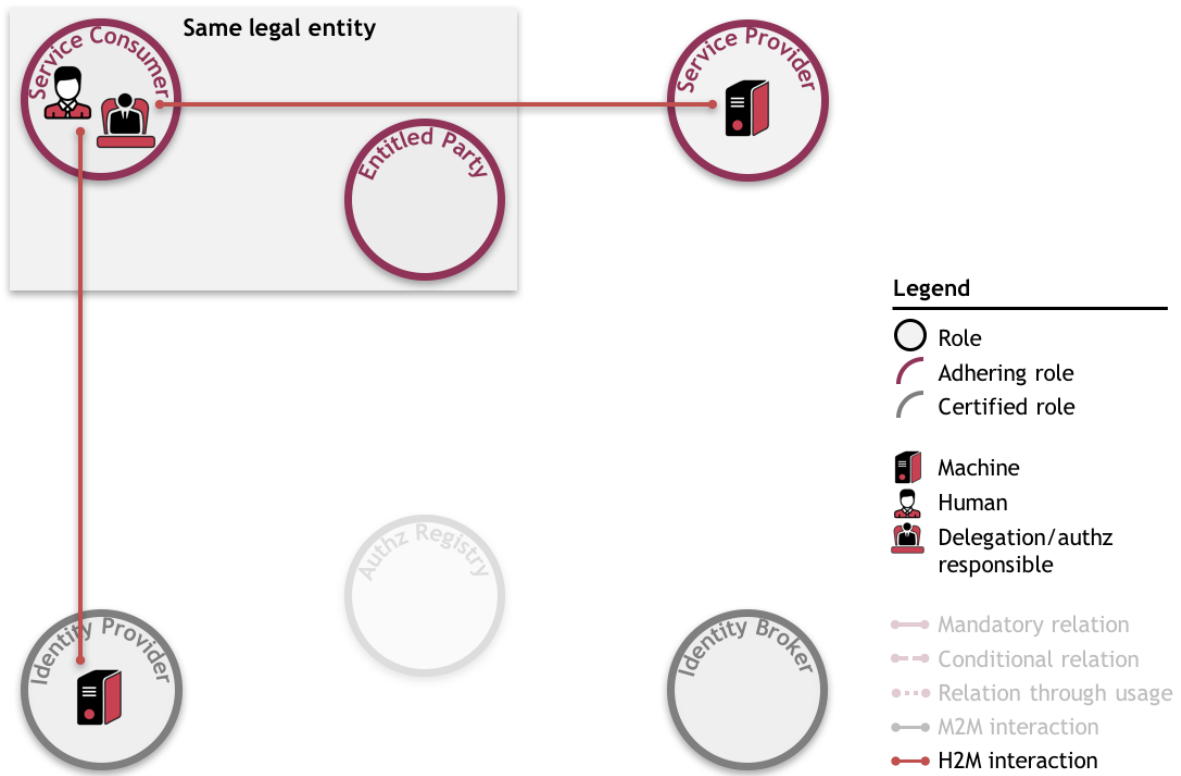
This use case would look as follows without an Identity Broker:

Depiction with Identity Broker

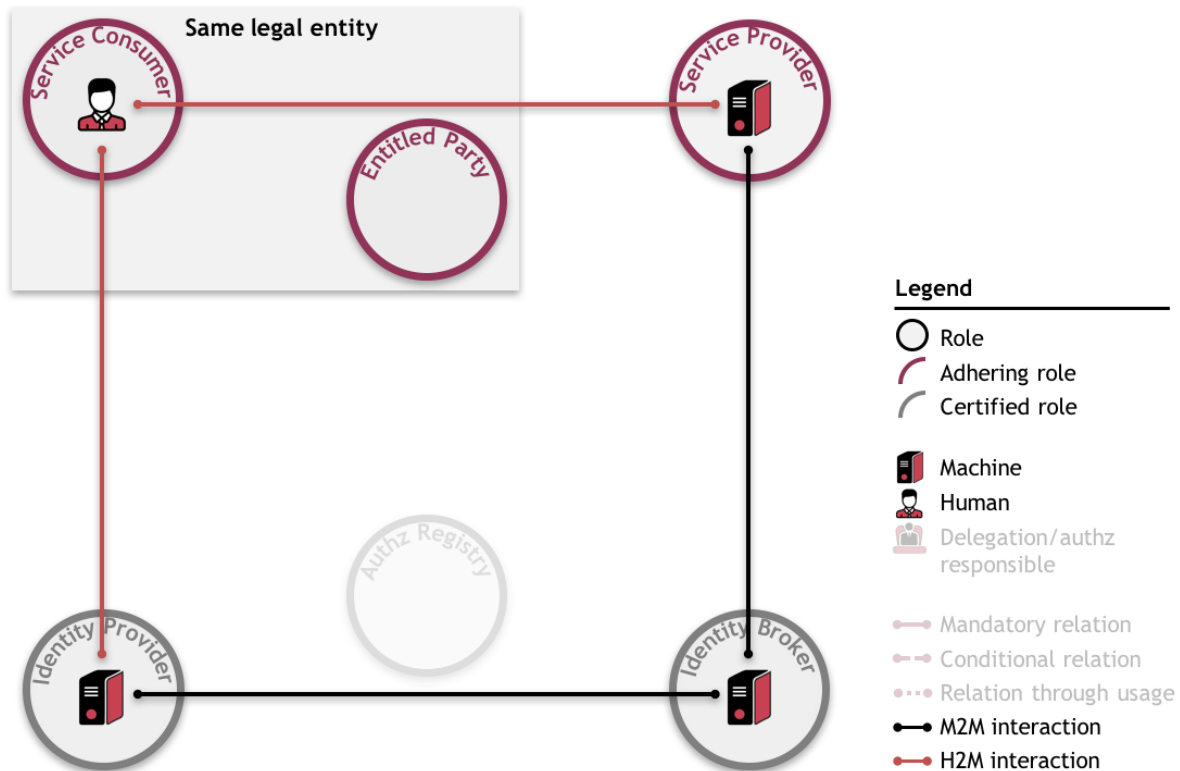
Legal relations



Prerequisite registration



Use case interaction



Description with Identity Broker

It is prerequisite of this use case that:

- The Service Provider has and manages its own authorization information indicating what Entitled Parties are entitled to what (parts of) services*;
- The Service Consumer has and manages its own authorization information indicating which Human Service Consumers are authorized to act on its behalf**;
- The delegation/authorization responsible at the the Service Consumer registers the authorization information at the Identity Provider;
- The Human Service Consumer is able to authenticate the Service Provider;
- The Service Provider is able to authenticate the Human Service Consumer;
- The Identity Provider is able to authenticate the Service Provider;
- The Service Provider is able to authenticate the Identity Provider;
- The Identity Broker is able to authenticate the Service Provider;
- The Service Provider is able to authenticate the Identity Broker;
- The Human Service Consumer has been issued identity credentials by the Identity Provider.

- In this use case the Entitled Party is also the Service Consumer.

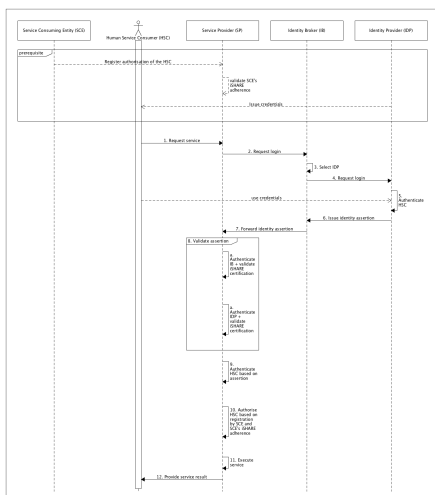
*The Service Provider can outsource this function to a third party

**The Entitled Party can outsource this function to a third party

The use case consists of the following steps:

1. The Human Service Consumer requests a service from the Service Provider;
2. The Service Provider requests a login from the Identity Broker;
3. The Identity Broker asks the Human Service Consumer to select his Identity Provider;
4. The Identity Broker requests a login from the Identity Provider;
5. The Identity Provider authenticates the Human Service Consumer;
6. The Identity Provider issues an identity assertion and authorization assertion for the Service Provider to the Identity Broker;
7. The Identity Broker forwards the identity assertion and authorization assertion to the Service Provider;
8. The Service Provider validates the identity assertion and authorization assertion through the following steps:
 - a. The Service Provider authenticates the Identity Broker and validates its iSHARE certification;
 - b. The Service Provider authenticates the Identity Provider and validates its iSHARE certification.
9. The Service Provider authenticates the Human Service Consumer based on the validity of the identity assertion, and validates the iSHARE adherence of the Service Consumer;
10. The Service Provider authorizes the Human Service Consumer of the Service Consumer based on the authorization assertion and the entitlement information registered with the Service Provider;
11. The Service Provider executes the requested service;
12. The Service Provider provides the service result to the Human Service Consumer.

Sequence diagram with Identity Broker



Secondary use cases

iSHARE's [three primary use cases](#) are supported by seven secondary use cases. These include:

- Processes related to registration;
- Processes that recur in primary use cases.

Processes related to registration

These four secondary use cases need to be completed before any, or specific, primary use cases can be initiated.

Any party needs to:

- 1a. Register adherence/certification at Scheme Owner
and later needs to be able to:
- 1b. Modify adherence/certification at Scheme Owner

Before initiating Human to Machine use cases, the **Service Consumer** needs to:

- 2a. Create Service Consumer and/or Human Service Consumer identity at Identity Provider
Prerequisites:
 - An agreement needs to be in place between Service Consumer and Identity Provider;
 - An agreement needs to be in place between Service Provider and Identity Provider.
 later, a Service Consumer needs to be able to:
- 2b. Modify Service Consumer and/or Human Service Consumer identity at Identity Provider

When delegating rights, the **Entitled Party** needs to:

- 3a. Register delegation at Service Provider, Entitled Party, or Authorization Registry
Prerequisite:
 - For registration at Service Provider or Authorization Registry, an agreement needs to be in place between Entitled Party and Service Provider or Authorization Registry.
 later, an Entitled Party needs to be able to:
- 3b. Modify delegation at Service Provider, Entitled Party, or Authorization Registry

When authorizing something or -one, the **Service Consumer** needs to:

4a. Register authorization at Service Provider, Entitled Party, or Authorization Registry

Prerequisite:

- For registration at Service Provider or Authorization Registry, an agreement needs to be in place between Service Consumer and Service Provider or Authorization Registry.

later, a Service Consumer needs to be able to:

4b. Modify authorization at Service Provider, Entitled Party, or Authorization Registry

Processes that recur in primary use cases

These three secondary use cases form the wiring of all primary use cases. Without them, primary use cases cannot be completed successfully.

In any primary use case, **any party** needs to:

5a. Check whether its counterparty is iSHARE adherent/certified (with the Scheme Owner)

5b. Check whether its counterparty's certificate is valid

In any primary use case, the **Service Provider** *also* needs to:

6. Determine an authorization decision based on entitlement-, delegation-, and/or authorization info in its own contract administration and/or from external PIPs

When delegation- or authorization info is requested by a Service Provider, an **Authorization Registry** or **Entitled Party** also needs to:

7. Determine authorization decision based on Service Consumer assertion included in Service Provider's request

Please note that the secondary use cases will not be detailed more than the above. No depictions or sequence diagrams are to be developed (contrary to for the primary use cases). This (deliberately) leaves freedom in implementation.

Licenses

Within iSHARE it is possible to explicitly provide instructions on how a service may be consumed or under which conditions data is exchanged. These instructions or conditions are called 'licenses'. Licenses are a crucial part of iSHARE, because they provide its participants the possibility to clearly state what is and what is not allowed. Since all iSHARE participants are bound to the same contract and underlying scheme rules, participants can appeal to each other to follow the provided licenses.

License code list

Purpose code	Description
0000	No limitations
0001	Re-sharing with Adhering Parties only
0002	Internal use only
0003	Non-commercial use only: licensee may not use the data to generate revenue
0004	Licensee may enrich received data with own data before re-sharing
0005	Licensee may enrich received data with data of others before re-sharing
0006	Licensee may enrich received data with own data before re-sharing on a non-commercial basis
0007	Licensee may enrich received data with data of others before re-sharing on a non-commercial basis
9999	As determined between Parties

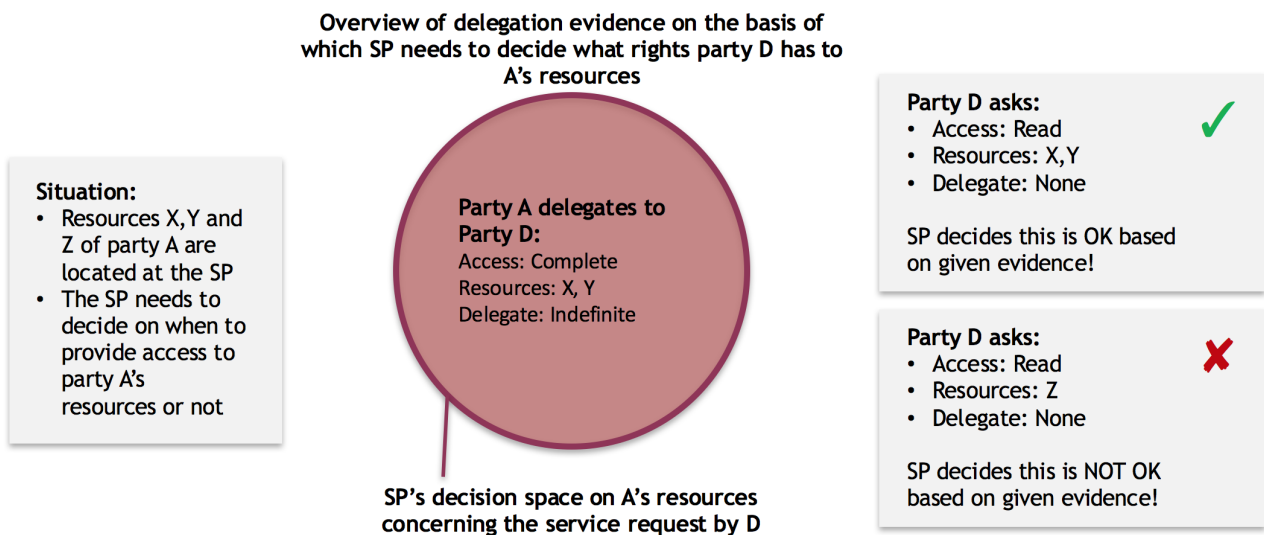
Delegation paths

A key functionality of iSHARE is delegating rights to another party, authorising them to act on your behalf. A single delegation was described in the [delegation use case](#).

In essence, Service Providers need to decide whether a Service Consumer is allowed access to a certain resource. To take the right access decisions, Service Providers need to interpret all relevant evidence to come to a decision: in other words: a 'logical sum' of evidence. This page further elaborates on situations where more than one delegation are issued that have overlapping properties.

Example 1: Single delegation

In the situation of a single delegation, a Service Provider could encounter the following situation:



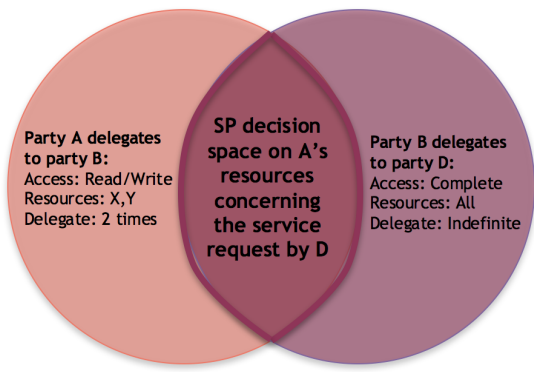
Example 2: Simple path of delegation

In practice, it can occur that various organisation delegate rights to various other organisation. Combining these delegations, a 'path of delegation' can be established, as is illustrated in the following example:

Overview of delegation evidence on the basis of which SP needs to decide what rights party D has to A's resources

Situation:

- Resources X,Y and Z of party A are located at the SP
- The SP needs to decide on when to provide access to party A's resources or not



Party D asks:

- Access: Read
- Resources: X,Y
- Delegate: None

SP decides this is OK based on given evidence!

Party D asks:

- Access: Read
- Resources: Z
- Delegate: None

SP decides this is NOT OK based on given evidence!

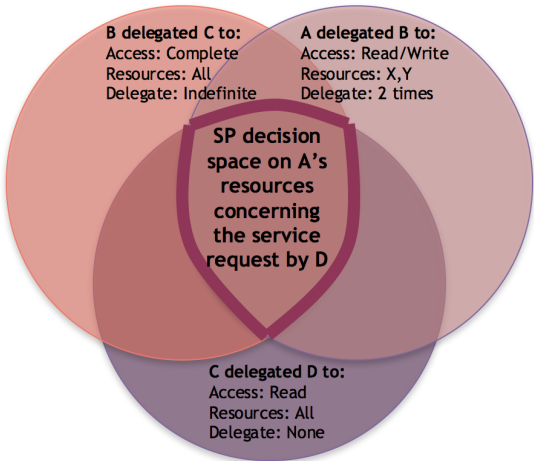
Example 3: Complex path of delegation

The following example illustrates a more complex delegation situation, where specific rights are delegated in terms of actions, resources and the right to further delegate these rights:

Overview of delegation evidence on the basis of which SP needs to decide what rights party D has to A's resources

Situation:

- Resources X,Y and Z of party A are located at the SP
- The SP needs to decide on when to provide access to party A's resources or not



Party D asks:

- Access: Read
- Resources: X,Y
- Delegate: None

SP decides this is OK based on given evidence!

Party D asks:

- Access: Read
- Resources: Z
- Delegate: None

SP decides this is NOT OK based on given evidence!

Party Q resides over party A's resources. When evaluating the available delegation evidence, organisation Q can conclude that organisation D has 'read' rights to resources X and Y but is not allowed to delegate these reading rights any further.

What is important to note for this path of delegation, is that the delegation rights **do not have to be given in a chronological order**. If party C just now delegated rights to D while party D would have requested access earlier than party C would have delegated rights, the delegation path would not exist.

Within iSHARE, it is possible to define more detailed rights to resources - as described in the [key functionality section in the introduction](#). For a detailed technical explanation of delegations, please refer to the '[structure of delegation evidence](#)' chapter.

Functional requirements per role

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

This chapter summarises the responsibilities and functional requirements per role:

- Adhering roles:
 - [Functional requirements per role#Service Consumer](#);
 - [Functional requirements per role#Service Provider](#);
 - [Functional requirements per role#Entitled Party](#).
- Certified roles:
 - [Functional requirements per role#Identity Provider](#);
 - [Functional requirements per role#Identity Broker](#);
 - [Authorization Registry](#).

One requirement to any legal entity fulfilling a role is that they **MUST** [provide a unique identifier](#).

Adhering roles

Please refer to the [detailed Operation descriptions](#) for what criteria need to be met to be admitted to the iSHARE network.

Service Consumer

The Service Consumer-role is fulfilled by a legal entity that consumes a service, such as data, as provided by a Service Provider.

A Service Consumer can be represented by a machine (its system) or a human, fittingly called the Machine Service Consumer and the Human Service Consumer.

The **functional requirements** applicable to Service Consumers are as follows:

- iSHARE adherence is REQUIRED.

Service Provider

The Service Provider-role is fulfilled by a legal entity that provides a service, such as data, for consumption by a Service Consumer.

The **functional requirements** applicable to Service Providers are as follows:

- iSHARE adherence is REQUIRED;
- All user interfaces available in an iSHARE context **MUST** comply with the iSHARE's [user interface requirements](#).

Entitled Party

The Entitled Party-role is fulfilled by a legal entity that has one or more rights to a service provided by a Service Provider, for example to data. These rights, or entitlements, are established in a legal relation between the Entitled Party and the Service Provider.

The Entitled Party- and Service Consumer-roles can be fulfilled by the same entity - i.e. a legal entity that consumes a service based on its own entitlements to this service - but this is not necessary. Legal entities that are entitled to a

service can delegate other entities to consume this service on its behalf: the legal entity consuming the service, then, does so on the basis of *another entity's* entitlements. In such use cases, as always, the Service Consumer consumes a Service Provider's service on the basis of the Entitled Party's entitlements, but the Service Consumer-role is fulfilled by another entity than the Entitled Party-role.

The **functional requirements** applicable to Entitled Parties are as follows:

- iSHARE adherence is REQUIRED.
 - Note: as it is the responsibility of the Service Provider to determine the Entitled Party, the Service Provider can choose to provide services where the Entitled Party is not admitted to iSHARE. In this event, the responsibilities of the Entitled Party are shifted to the Service Provider in question. This is particularly useful for Service Providers who have existing (smaller) customers, who do not have own systems, or are only an Entitled Party for services at a single Service Provider.

Certified roles

Please refer to the [detailed Operation descriptions](#) for what criteria need to be met to be admitted to the iSHARE network.

Identity Provider

The Identity Provider-role is fulfilled by a legal entity whose tooling identifies and authenticates humans (and specifically, Human Service Consumers representing Service Consumers). An Identity Provider:

- Provides identifiers for humans;
- Issues and manages [credentials](#) (i.e. a password or electronic keycard) for humans;
- Receive authentication requests from Service Providers;
- Provides an online interface to authenticate humans based on their credentials;
- Can hold information on authorisations of humans representing a Service Consumer; i.e. information indicating which humans are authorised to act on a Service Consumer's behalf;
- Can, after successful identification and authentication, on the basis of this information, determine whether the human representing a legal entity is authorised to take delivery of a service;
- Can confirm the identity, authentication and authorisation information to the Service Provider.

As a result, Service Providers can outsource identification and authentication to an Identity Provider instead of implementing their own tooling.

The **functional requirements** applicable to Identity Providers are as follows:

- The Identity Provider **MUST** have a clear agreement with the authorising entity concerning the process of allowing the registering, updating or removing of an authorisation;
- The Identity Provider **MUST** prevent that a revoked authorisation is processed as a valid authorisation;
- The Identity Provider **MUST** ensure that the identification and authentication process conforms to the Level of Assurance requested by the Service Provider;
- The Identity Provider **MUST** conform to the service levels for Certified Parties as described [here](#);
- The Identity Provider **MUST NOT** claim accordance with a Level of Assurance for which it has not been certified by the Scheme Owner;
- The processes of the Identity Provider **MUST** be in accordance with the Level of Assurance for which the Identity Provider has been certified;
- iSHARE certification is REQUIRED;
- All user interfaces available in an iSHARE context **MUST** comply with the iSHARE's [user interface requirements](#).

Identity Broker

Different humans might hold identifiers at different Identity Providers. Also, Service Providers might need to connect to several Identity Providers. To make sure Service Providers do not need a relation with each Identity Provider individually, an Identity Broker is introduced. The **Identity Broker**-role is fulfilled by a legal entity that provides Service Providers access to different Identity Providers, and that offers humans the option to choose with which Identity Provider to identify and authenticate themselves throughout the iSHARE Scheme.

As a result, if Service Providers choose to outsource identification and authentication to more than one Identity Provider, they can connect to an Identity Broker instead of to several Identity Providers.

The **functional requirements** applicable to Identity Brokers are as follows:

- The Identity Broker MUST provide users an interface to select their Identity Provider;
- The Identity Broker MUST conform to the service levels for Certified Parties as described [here](#);
- The Identity Broker MUST NOT claim accordance with a Level of Assurance for which it has not been certified by the Scheme Owner;
- The processes of the Identity Broker MUST be in accordance with the Level of Assurance for which the Identity Broker has been certified;
- iSHARE certification is REQUIRED;
- All user interfaces available in an iSHARE context MUST comply with the iSHARE's [user interface requirements](#).

Authorization Registry

The Authorization Registry-role is fulfilled by a legal entity who provides solutions for Adhering Parties for the storage of delegation information. An Authorization Registry:

- Can hold information on delegations to Service Consumers; i.e. information indicating what parts of the rights of an Entitled Party are delegated to a Service Consumer;
- Has a process in place allowing for the registration, update and revocation of delegations;
- Can check, on the basis of this information, whether a legal entity is authorised to take delivery of a service;
- Can confirm whether this is the case to the Service Provider.

As a result, Adhering Parties can outsource tasks concerning the management of delegation information to an Authorization Registry instead of implementing their own tooling.

The **functional requirements** applicable to Authorization Registries are as follows:

- The Authorization Registry MUST have a clear agreement with the delegating entity concerning the process of allowing the registering, updating or removing of a delegation;
- The Authorization Registry MUST prevent that a revoked delegation is processed as a valid delegation;
- The Authorization Registry MUST conform to the service levels for Certified Parties as described [here](#);
- The Authorization Registry MUST NOT claim accordance with a Level of Assurance for which it has not been certified by the Scheme Owner;
- The processes of the Authorization Registry MUST be in accordance with the Level of Assurance for which the Authorization Registry has been certified;
- iSHARE certification is REQUIRED.

Identification by EORI

This part of the iSHARE scheme is considered normative and is therefore compliant with RFC 2119.

In order for parties to identify other parties, any party fulfilling a role in the iSHARE framework MUST provide a unique identifier.

For this purpose, the Economic Operators Registration and Identification ([EORI](#)) number is reused. An EORI is unique and valid throughout the European Community and is assigned by a customs authority or designated authority in a Member State. Even non-European Community parties doing business in/with Europe have an EORI, but if no EORI is available, a party is also allowed to supply its Chamber of Commerce number as alternative identifier.

Currently the following identifiers are supported in iSHARE:

Identifier	Example	Description
EORI number	EU.EORI.NL123456789	Economic Operators Registration and Identification
KvK number	NL.KVK.12345678	Dutch Chamber of Commerce number
Scheme Owner Identifier	EU.EORI.NL000000000	Unique identifier for the Scheme Owner

Role identifiers

In certain cases, when identifying a Certified Party it is also important to identify their iSHARE 'role'. For this purpose iSHARE specifies the following identifiers:

Role identifier
IDENTITY_PROVIDER
IDENTITY_BROKER
AUTHORIZATION_REGISTRY

User interface requirements

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

For all Human to Machine interactions, as in [primary use case 2](#) and [3](#), an interface is required. This interface MUST comply with the following guidelines:

- The name of the legal entity that provides a broker service or identity provisioning service MUST be clearly visible;
- During the process of authentication, information not directly relating to the identity provision process or supporting the identity provision process MAY NOT be present. Links to websites irrelevant to the identity provisioning process or advertisements MAY NOT be present;
- Parties facilitating the identity provision process MAY use their own corporate styling and logos;
- The iSHARE brand MUST be shown during the identity provision process. Showing the iSHARE brand MUST be in line with iSHARE [communication](#) guidelines;
- Human Service Consumer that are being identified through the use of a browser MUST be able to verify the URL and used SSL certificate during all steps of identity provisioning process.

Please note that extra guidance will need to be added for the context of apps: how can Human Service Consumers verify that they are not being tricked?

Technical

This section covers the Technical details of the iSHARE Scheme.

The section starts out with a chapter containing an [overview of relevant technical standards](#) that apply to the iSHARE Scheme in general. Next, this section provides a dedicated chapter on the '[delegation evidence structure](#)', a JSON data structure which specifies how Authorisation Registries and Entitled Parties need to be able to present delegation evidence upon request.

iSHARE role-specific API requirements and the APIs which are exposed by the Scheme Owner can be found in the dedicated [iSHARE Developer Portal](#).

- [Generic technical standards](#)
- [Structure of delegation evidence](#)

Generic technical standards

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

This chapter contains information on the generic technical standards that are applied in the iSHARE Scheme, relevant to all parties involved.

iSHARE can be described as an API architecture, which enables all parties involved to engage in direct communication. For interoperability reasons, iSHARE makes use of widely used open standards. Modified implementations of OAuth 2.0 and OpenID Connect 1.0 are used to facilitate an ecosystem in which parties can interact with previously unknown parties. Pre-registration, therefore, is not a prerequisite and this requires alterations to the official standards. Also, for the authentication of parties within an iSHARE context, iSHARE uses PKI and digital certificates relating to all participating parties.

Technical standards used in iSHARE and configuration aspects

The iSHARE Scheme also prescribes various general interface specifications regarding **Caching, Dates & Times, Party Identifiers, Response Codes** and **Web Server configuration**. These are described in the following table and corresponding topic pages as referred to in the table.

***BOLD:** Contains specific iSHARE specifications

Technical standard	Character	Description
API	Architectural principle	<p>Application Programming Interface</p> <p>API's are used in iSHARE to facilitate direct and realtime communication between different parties, eliminating the need for a central platform.</p> <p><i>An API (Application Programming Interface) is a technical interface, consisting of a set of protocols and data structuring standards ('API specifications') which enables computer systems to directly communicate with each other. Data or services can be directly requested from a server by adhering to the protocols. APIs are used to hide the full complexity of software and make it easy for third parties to use parts of software or data services. APIs are mainly meant for developers to make the creation of new applications depending on other applications easier.</i></p> <p>iSHARE prescribes caching requirements relating to the use of APIs in various situations.</p>
PKI	Architectural principle	<p>Public Key Infrastructure</p> <p>System for issuing and managing digital certificates. For authentication purposes, iSHARE requires adhering and Certified Parties to acquire an X.509 certificate which is distributed by a trusted root under certain PKI's (Public Key Infrastructure). For interoperability on a European scale, all trusted roots under the eIDAS regulation will be trusted within iSHARE. However, initially, this will be limited to certificates issued under PKIoverheid.</p>

Technical standard	Character	Description
OAuth 2.0	Open standard for authentication	<p>Authentication standard, used in iSHARE to gain access to services through access tokens. iSHARE has modified the OAuth 2.0 standard to work without pre-registration.</p> <p>Pre-registration of clients MUST NOT be used. Certificate and status validation with the iSHARE Scheme Owner is sufficient for authentication purposes. If needed, clients can be registered after authenticating. To ensure security in unknown clients, iSHARE prescribes whitelisted Certificate Authorities that MUST be used.</p> <p>The OAuth 2.0 subpage also describes the iSHARE generic Authentication flow.</p>
OpenID Connect 1.0	Open standard for authentication of humans	<p>Authentication standard for the authentication of humans in an online context. Functions as an additional layer on top of the OAuth 2.0 protocol.</p>
HTTP(S)	Communication protocol	<p>HyperText Transfer Protocol (Secure)</p> <p>iSHARE Scheme communication MUST be carried out over the HTTP protocol, and secured through TLS 1.2 resulting in HTTPS.</p> <p>iSHARE authentication/authorisation data is generally transferred in HTTP Headers. These headers can become very large when containing multiple encrypted certificates or JWT's. iSHARE parties SHOULD configure their web servers to accept HTTP headers of 100K length to minimise implementation impact on current services</p> <p>The most recent version of the HTTP specification can be found here.</p> <p>An overview of relevant iSHARE HTTP response codes can be found here.</p>
TLS 1.2	Cryptographic protocol	<p>Transport Layer Security</p> <p>Transport Layer Security (TLS) is a cryptographic protocol that describes communication security for computer networks. It is used to secure the HTTP protocol, resulting in HTTPS. Within iSHARE, TLS 1.2 MUST be used for securing all HTTP communications.</p> <p>For the most recent version of the specification click on this link.</p>

Technical standard	Character	Description
RESTful	Architectural style for API design	<p>Representational State Transfer</p> <p>REST is an architectural style for building systems and services, systems adhering to this architectural style are commonly referred to as 'RESTful systems'. REST itself is not a formal standard, but it is an architecture that applies various common technical standards such as HTTP, JSON and URI.</p> <p>Within iSHARE RESTful architectural principles MUST be applied to the APIs that are specified.</p> <p>A RESTful API indicates that the API architecture follows REST 'constraints'. Constraints restrict the way that servers respond and process client requests, in order to preserve the design goals which are intended by applying REST. Goals of REST are, among others, performance and scalability. Both are of utmost importance in iSHARE.</p> <p>RESTful systems are able to process common HTTP operations, such as GET, POST and DELETE.</p>
JSON	Open standard for file formatting	<p>JavaScript Object Notation</p> <p>JSON is an open standard data format that does not depend on a specific programming language. This compact data format makes use of human-readable (easy to read) text to exchange data objects (structured data) between applications and for data storage.</p> <p>Within iSHARE, JSON is used as data structuring standard for scheme related communication. For the most recent version of the JSON specification click on this link.</p>
JSON Web Token (JWT)	Open standard for definition of access tokens	<p>JSON Web Token</p> <p>A JSON Web Token (JWT) is used in iSHARE when non-repudiation between parties is required. A statement, of which the data is encoded in JSON, is digitally signed to protect the authenticity and integrity of the statement.</p> <p>All iSHARE JWTs MUST be signed using the JWS specifications.</p>
XACML 3.0	Access control policy language	<p>eXtensible Access Control Markup Language</p> <p>Standard for defining authorisation policies. Within iSHARE, a JSON port of XACML 3.0 is used to enable parties to communicate delegation evidence.</p> <p>For the most recent version of the specification click on this link.</p>

Technical standard	Character	Description
X.509	Standard for the format of public key certificates	<p>X.509 is a cryptographic standard for public key infrastructures (PKI's) that specifies the management of digital certificates and public-key encryption and keys of the Transport Layer Security (TLS) protocol that is used to secure web and email communication.</p> <p>For the most recent version of the specification click on this link.</p>
UTC	Time standard	<p>In iSHARE all dates and times MUST be communicated in UTC time.</p> <p>All dates and times MUST be formatted in the Unix timestamp format.</p>

Caching

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

Often data is temporarily stored on a different medium, to enable faster access to the data.

For every API exposed under iSHARE caching MUST Be made explicit to the API consumer.

If a response is not cacheable it MUST contain the following headers:

Adherence information
Cache-Control: no-store Pragma: no-cache

If a response is cacheable it MUST contain the following headers:

Adherence information
Cache-Control: max-age=31536000

Note: max-age MAY vary

HTTP response codes

After sending a HTTP request to a server, the server responds with (among others) a Status Code which indicates the outcome of the request made to the server.

Within the iSHARE Scheme, the HTTP standard concerning response codes is followed as established by the IETF. Please refer to the [IETF website](#) for further specification. Within iSHARE the HTTP response codes 401, 403, 406, 409 and 412 are most relevant.

HTTP Verb	CRUD	Entire Collection (e.g. /customers)	Specific Item (e.g. /customers/{id})
POST	Create	201 (Created), 'Location' header with link to /customers/{id} containing new ID.	404 (Not Found), 409 (Conflict) if resource already exists..
GET	Read	200 (OK), list of customers. Use pagination, sorting and filtering to navigate big lists.	200 (OK), single customer. 404 (Not Found), if ID not found or invalid.
PUT	Update/Replace	404 (Not Found), unless you want to update/replace every resource in the entire collection.	200 (OK) or 204 (No Content). 404 (Not Found), if ID not found or invalid.
PATCH	Update/Modify	404 (Not Found), unless you want to modify the collection itself.	200 (OK) or 204 (No Content). 404 (Not Found), if ID not found or invalid.
DELETE	Delete	404 (Not Found), unless you want to delete the whole collection—not often desirable.	200 (OK). 404 (Not Found), if ID not found or invalid.

ggEPADCCAQoCggEBALp1Yk0cu6U7M13mcQupWj+TADy/
 hwEHWmmGGFFRaLMURuuJdSkChEu89Lvq71QQWkxxJWAHP6oRe3UCzCdMjXh6PoEQnnyC3dUgX3pGjQG1oT0a7
 tyMdMwzAW92MJW2BF8P7ZRMldskf7BJh1QPrF/p53+S0uexZNSB3/
 ZZjLpantIL3iF9mxPKBJX2c5dY666rt3+fadLhGf0AtG950M6BV3GmMNbx8sNQV+hCQ00eIirhki+08ovcHT
 E7DP7+3BB2edDoCgSXStLHrodB8wZBBGLBgm7tqcDz5pkaDetIrsYKNXQqERFjU2TYurBlnl64pce/
 SOUiy+KDxhjcCAwEAAaOCAVwvvggFYMAKGA1UdEwQCMAAwEQYJYIZIAyb4QgEBBAQDAgZAMDMGCGSAGG+EIB
 DQmFIRPcGVuU1NMIEdlbnVYXRlZCBTZjJ2ZXIgc2VydGlnaWNhdGUwHQYDVR00BBYEFBEKZ61hpM+ZDLNW
 LKlxbBliVoMIG+BGNVHSMegbYwgb0AFI5GTdMUiapXImVUu9T21ZsCN9zroYGWpIGTMIGMQswCQYDVQGEW
 J0TDELMAG1UECAwCTkgxEjAQBGNVBACMCUFTc3RlcmRhbTEPMA0GA1UECgwaVNIQVJFMREwDwYDVQQLDAh
 TZWN1cmloTEUMBIGA1UEAwWLaVNIQVJFIFJvb3QxJjAkBgkqhkiG9w0BCQEF2luZm9AaXNoYXJLLXByb2pL
 Y3Qub3JnggIQADA0BgNVHQ8BAf8EBAMCBaAwEwYDVR0lBAwwCgYIKwYBBQUHAwEwDQYJKoZIhvcNAQELBQADg
 gIBAANm0QHtsbKfVqxr6jJgtDvVIGQmuryMpue1Log6HZZ2QowZWrG8o/
 4SAGlpMPTuVU0UfABk5dVf0XnmBa5LRMI7h19dSM1HNle6C9WA7RQtNV/
 v4qBe00lgfaD4cUAJDkHsIwWSMLcel0oxVZNcd0wadXAQHgYduzBSdR8/
 Ps3plvIDIE9lRgt7GkUzS3WU1XVss6nKZFWLZktqQH5Y+WEG//60+1Wf4aI6VHIuoRj10/
 NlEvjct0z0yiZU2RrqpQsrtyBYbCPIu0+Cl9QM73pHY3zYqkwy4CLaewvypZaY5KDBh7nZ0p9NJ1Z2XWfUVI
 DTZReH2ARXfPkWdaHmhAcMZ9BiqM+hX4IXeC68Vvua+guypPJZfRyE33sox/lu8ecL2L7/
 ehDgji8IESymUPI32CpKfMN1KKNL/
 KEtftGPpuV+6iQNTE4htCBcBaSf3dxsGhcl0SC6Ke9tL4YRLiX3+YsHqYD98vLRR0bIQZGWXiqvSfSLCwK0M1
 RIwsfb6B9S+XMRaiwr1iezBdHxXaH81LT+WxJfnDun6uxXUz4xTuzaVXsV0gvcY3quYp64LR6Rrhnc2DKndzZ
 U6JHyF7LX70rn8Lj180jjG1ge6ll9DLPQtkVyShTqCUV+9wrU2aE16rqe2YAq/
 LLJ0dW0HS2pJcQjh1iDpAqG0bwT",
 "MIIGBDCCA+ygAwIBAgICEAAwDQYJKoZIhvcNAQELBQAwZAxzAJBgNVBAYTAk5MMQswCQYDVQQIDAJOsDES
 MBAGA1UEBwwJQW1zdGvYzGFtMQ8wDQYDVQQKDAZpU0hBUkUxETAPBgNVBAsMCFNlY3VyaXR5MRQwEgYDVQQDD
 AtpU0hBUkUgUm9vdDEmMCQGCsQsIb3DQEJARYXaW5mb0Bpc2hhcmUtcHJvamVjdC5vcmcwHhcnMTcwNjI3MD
 YxNDM0WhcnMjcwNjI1MDYxNDM0WjCBkDELMAKGA1UEBhMCTkwzCzAJBgNVBAGMAk5IMQ8wDQYDVQQKDAZpU0h
 BUkUxETAPBgNVBAsMCFNlY3VyaXR5MSGwJgYDVQQDDB9pU0hBUkUgTkkgQ2VydGlnaWNhdGUgXV0aG9yaXR5
 MSYwJAYJKoZIhvcNAQkBFhdhpbmZvQGZlzaGFyZS1wcm9qZWN0Lm9yZzCCAiIwDQYJKoZIhvcNAQEEBQADggIPA
 DCCAgocggIBAKiUQA3R14zhAZ59xu4J6j8h5LaMyOmzFIRzeV+665E8bnoK2eQSC5aYMvsb80o8DxHHjiqPJ
 iJW1n1oT86UAkgaEEnJ8/GhF+2ky0lybdpM21Q6bNwkJ5vL4SD/
 GYBPMwsVt7X6pLakFswXu6wRzvrYXWcVm/z/3SHX9AZFg6HDZRsGGDnnRw14nHYpb/
 oh7p5h6atI+X0a0D4HhQJ1ahQLVozIbbCFdfpKu9mrDXzKYDaNstrYn2DmbtV40BC5FW10T0HLO/
 texnnhtNrq6IVgUkZ54NHT8Y2EQNmajy8RilrLU1FHS/
 ka03w8sV0sXDsLeBwbx13BYHpisPMWpMJDFU8b92tlyjk2lUDnHqDUi5BHRdRYqVZeWBSL+QevpeZr3mCL9yn
 hGPy+tNyDjyVjb/y7h1ftAzZodRwItQpZpKHCWQLsWX+JwM9lwu5uq6lcqxHGPSAB0o/
 ME80KE3a4KnjCGYvLzIPjtrad0HZDRR/hEAnen3HE3fTZE42u2wBLTfZB1D8Kxjc44vJv8L4J0aLlt/
 ooMT913sGCSJDPQUwv+/
 VvYDV8ngkgzRCpFng2NrPaWao6RKJ99bBMkKFQ4vsdtvwi42GK20BK7ozXnNS7B5qcljdVnrmmH8fCPJ/
 s7L9BDYw/IKrd5wHaSH1kZ7/
 DRa0fa2mMjJfLAgMBAAGjZjBkMB0GA1UdDgQWBBSORk3TFImqVyJlVIvU9tWbAjfc6zAfBgNVHSMEGDAWgBRL
 YazzoNpuBb//4Sr1Ior0iP04RjASBgNVHRMBAf8ECDAGAQH/AgEAMA4GA1UdDwEB/
 wQEAWIBhjANBgkqhkiG9w0BAQsFAAOCAGEAQ7GRIQ3Q+TRUumX0xc+dGJnx98EK1qD8E1uEptLS+VR0WAZ22Q
 eLCPk/
 nNDVs2AmCADZF0JoX2+RjNjq7pX2+AxcL1owWQoi3GRtL6GABHByuLtfJfvodYjUipI1zmIvcWkzNnNCC9A0r
 dtUJyRjZvGZt32K77aiRNGiSctVyCBuFBPqtqCkXIz86e/wzQ1fwBeCRB0WDMoSdXbSkt/
 tapyGoU7oAj2DVWbtKaCnkKiysE19r1RCiZI2WAHLcuU9iLvNM1mfow9avI+rVq2YlKU0uCIrD7s/
 ILRLXg55VwqMJT33/50NFnu3H8ebmqEhkGYStk7p3FGRxgptd20JnqAt5nG1pspHQUD+vYBoKkkMw40qvy+ee
 YkhyzKcJTKe0sfI29fyQx/
 eFSmWrheT188+jZQyCqEQpZrgtku+2KPQbCBxVeCfHacyS16+9Z0Vs1zXWGFiqXKXwZF1v6CDyL7bedJZEuT
 1wUpkEUJH/IPmdp2ZMMcss/BIdyf/
 +pYwyMnB6DJocDwMjlm3cUEsbtT393wIie0mohWiu8myTxkCFN1VJs4W9chSx5/

```

DxVwDtFoT4nsqVK9F6DKYJAJ95UR+C+RiCS+x7t4r8cup0AiJpg7JEGr5I9kdLBXKak78Pr/
oqZogFymWPzoWUy48Ye6WxaAA1s+h6hVQ22MCg=",
"MIIGCDCCA/
CgAwIBAgIJAN7kMSjuGT9KMA0GCSqGSIB3DQEBCwUAMIGQMqswCQYDVQQGEwJOTDELMAkGA1UECAwCTkgxEjA
QBgNVBACMCUFtc3RlcmRhbTEPMA0GA1UECgwGaVNIQVJFMREwDwYDVQLDAhTZWNN1cm0eTEUMBIGAIUEAwWL
aVNIQVJFIFJvb3QxJjAkBgkqhkiG9w0BCQEFW2luZm9AaXNoYXJlLXByb2p1Y3Qub3JnMB4XDTE3MDYyNzA2M
DY1NFoXDTM3MDYyMjA2MDY1NFowZAxzAJBgNVBAYTAk5MMQswCQYDVQQIDAJOSDESMBAGA1UEBwwJQW1zdG
VyZGFtMQ8wDQYDVQKDAZpU0hBUkUxETAPBgNVBAsMCFNlY3VyaXR5MRQwEgYDVQQDDAtU0hBUKUgUm9vdDE
mMCQGCSqGSIB3DQEJARYXaW5mb0Bpc2hhcmUtcHJvamVjdC5vcmcwggIiMA0GCSqGSIB3DQEBAQUAA4ICDwAw
ggIKAoICAQCuvV8gSQDx7g1PYvxcuK6uDIwWZnZKdF0wJoGjcI1qPJWF3pP2xCJq5ZRnUQnpW0TcWw9A4uZf
CIWqJSoISh4nE7UmMfw45ms+7SwJxfc7E+Y4ffMDMyLcmvHYYLqGDoZu4qnpys+Xs1e8ESyNtILXooZ60+
C1he+XmXpSCh0jh5Pnz7RHZ4cS7Zw38B9aNA3WktF+yR7ijkTUN887aZAp1mrNZoioGnM/
E1RFwnLqrk5E84k0TzexsNILBpBzoZkLpZ8yYI00J42/
RT5m9YPOIRWmWcgQmBHIgMCAU0jAslex2Bx1uQtBJcti+DI+f+ZIGPm/TcwsaCC+RbE51SmMqwXyE/
BoAlAZKYSgG8outN+Fd3Ew3h3YJcA86wDT9bKMS9oSV1EjXc8v+40pp//AdWMPf2q15i/
QotTj+SGP2cJQvcfRBJ94IfG41IBoleK7jEUN2JUcowaDnsYfM7EdoKGhIn+bkCXnZR3bnfBKBAXH64M0p7Ii
1NukjPpEqemyM3//XKn/1lXzU43rbazwbxyGBYq3Ahv5io7MS1TwegD/hNmTOEGHffZ/
CzJghm6WvgmcmUALu2IFcDpmYZI3UI3SKzwardi7/vcGer5VVDgLRtVtvHk3eJRT0L9SFqHIWsLunI/
dZSJE8pKpStG5Lzj7Y/ps0lBpahnwIDAQABo2MwYTAdbgNVHQ4EFgQUS2Gs86DabgW//
+Eq9SKK9Ij90EYwHwYDVR0jBBgwFoAUS2Gs86DabgW//+Eq9SKK9Ij90EYwDwYDVR0TAQH/BAUwAwEB/
zA0BgNVHQ8BAf8EBAMCAYYwDQYJKoZIhvcNAQELBQADggIBABt+I4ZnS45c2IFsKV+BA8JP50WCPoaJyL3PCE
LjedVfB4rXyF946+ILARcBZg35F07WzHtmi/
afXh6entvVMEAlEmH6ln8sR9hoM3rHnYRtSIJmJs2gSF8wgCOZd1pyfrBVu3f3SXbUgNjoe5tEW6xs0QsJJqN
nQiJMjx6ea56L2kbW7E6k0s+Mck9Kz8ZQQB58iPGyx3ahML0rxbPoItbqKuyDwco7AzL7f7ea9zye0up8ubNDY
FaKJfiIDsnVbUPA6FbTn4fSQpee0K1fjrzSaUpeM0d0omj0ZjwT6K7W7ckxof0ecW4dcPENpXImkBPQXsGxv
CLdf/6e/jPtI3CPgxJF3747Hq6tBSH0GzCrYv4bForQu43MzqZKHbX8FF1PpHSaiXJK/Hu8WJDXXBq5PedeL/
lJSajdTUDsiFXkj/pY0uJ6M0x/
EonozaCPcHwdPsvr6nrrHWQXo4ayP7ntegJA3GrGXcaP+peEQmF9nWgVwPq33C1TrPNaiMxHz5toxl0YhnbA+
5eH2CTBjDFJnr2uaeh8BzbdkE29Wk0sqZAIUEmMwxcxBW20JGAlhF6MPFADjPYG0Ljddjbwq2H+LQtRf+tE6
Z7nmW/cAT4dEB7Me1uJrYucVjLLSNXCRKQaDKKZMoHhAcileKLMrSgUDaAi0AUKXduCymUw3n"
]
}

```

Payload

For the payload of an iSHARE signed JWT the following requirements apply:

- The JWT payload MUST conform to the `private_key_jwt` method as specified in OpenID Connect 1.0 [Chapter 9](#)¹²
- The JWT MUST always contain the `iat` claim
- The `iss` and `sub` claims MUST contain the valid [iSHARE identifier](#) of the client¹
- The `aud` claim MUST contain only the valid [iSHARE identifier](#) of the server. (Including multiple audiences creates a risk of impersonation and is therefore not allowed)
- The JWT MUST be set to expire in 30 seconds. The combination of `iat` and `exp` claims MUST reflect that. See [Dates and times](#) for requirements
- Depending on the use of the JWT other JWT payload data MAY be defined

Additional rationale

¹ In OAuth 2.0 clients are generally pre-registered. Since in iSHARE servers interact with clients that have been previously unknown this is not a workable requirement. Therefore iSHARE implements a generic client identification and authentication scheme, based on iSHARE whitelisted [PKIs](#).

² Since OAuth 2.0 doesn't specify a PKI based authentication scheme, but OpenID Connect 1.0 does, iSHARE chooses to use the scheme specified by OpenID Connect in all use cases. This is preferred above defining a new proprietary scheme.

Example JWT payload

```
{
  "iss": "EU.EORI.NL123456789",
  "sub": "EU.EORI.NL123456789",
  "aud": "NL.KVK.12345678",
  "jti": "378a47c4-2822-4ca5-a49a-7e5a1cc7ea59", // Note this is not necessary a GUID
  "exp": 1504683475, // Equals iat + 30 seconds
  "iat": 1504683445
}
```

Processing a JWT

- A server SHALL NOT accept a JWT more than once for authentication of the Client. However within it's time to live a Service Provider MAY forward a JWT from a Service Consumer to one or more other servers (Entitled Party or Authorization Registry) to obtain additional evidence on behalf of the Service Consumer. These other servers SHALL accept the JWT for indirect authentication of the Service Consumer during the JWT's complete time to live
- A server SHALL only accept a forwarded JWT if the aud claim of the forwarded JWT matches the iss claim of the JWT from the client that forwards the JWT
- JWT contents that are not specified within the iSHARE scope SHOULD be ignored

OAuth 2.0

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

iSHARE uses the OAuth 2.0 protocol for authenticating parties and providing access tokens when requesting access to a service within iSHARE.

On this page a brief description of OAuth is provided. For the most recent version of the OAuth 2.0 specification click on [this link](#).

Furthermore this page describes the generic iSHARE Authentication flow.

iSHARE facilitates an ecosystem within which parties can interact with previously unknown parties, pre-registration is therefore not a prerequisite and thus requires alterations to the official standard.

Generic OAuth 2.0 requirements

In addition to the specifications below, for all uses of OAuth 2.0 the following requirements apply:

- Clients MUST NOT be pre registered. A look-up in the iSHARE adherence registry is sufficient. It is up to the server create a new entry for Clients that perform requests for the first time ¹
- The `client_id` MUST contain the valid iSHARE identifier of the client
- For interoperability reasons clients SHALL only make HTTP GET calls to the `/oauth2.0/token` endpoint.
- Servers SHALL NOT issue refresh tokens

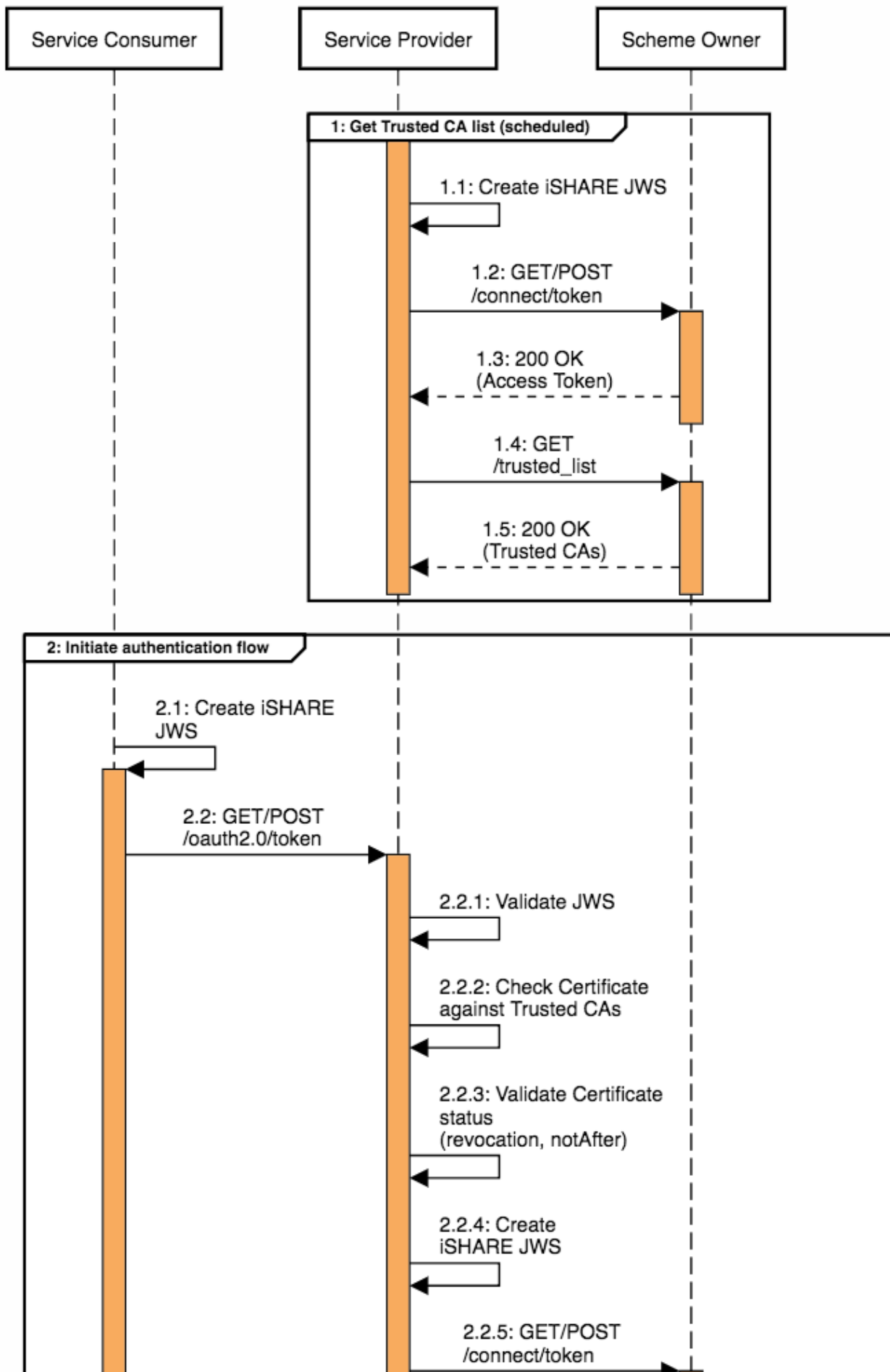
Additional rationale

¹ In OAuth 2.0 clients are generally pre-registered. Since in iSHARE servers interact with clients that have been previously unknown this is not a workable requirement. Therefore iSHARE implements a generic client identification and authentication scheme, based on iSHARE whitelisted PKIs.

iSHARE authentication flow

Based on the described standards and specifications in this scheme, the generic iSHARE Authentication flow is described in the following sequence diagram.

iSHARE Generic Authentication flow



Access token specifications in OAuth 2.0

Used to obtain an OAuth access token from a party that exposes an iSHARE API.

Based on the requirements in <https://tools.ietf.org/html/rfc6749>

OAuth access token API specifications example

Parameter	Contained in	Type	Required	Description
<div style="border: 1px solid black; padding: 5px; display: flex; align-items: center;"> <div style="background-color: #800040; color: white; padding: 5px 10px; border-radius: 5px; margin-right: 10px;">GET</div> <div style="border: 1px solid black; padding: 5px 20px; border-radius: 5px;">/authorisation_registry/oauth2.0/token</div> </div> <p>Used to obtain an OAuth access token from the Authorisation Registry</p>				
grant_type	query	string	Yes	OAuth 2.0 grant type. MUST contain "authorization_code"
scope	query	string	No	OAuth 2.0 scope. Defaults to "iSHARE", indicating all rights are requested. Other values MAY be specified by the API owner and allow to get tokens that do not include all rights
client_id	query	string	Yes	OpenID Connect 1.0 client ID. Used in iSHARE for all client identification for OAuth/OpenID Connect. MUST contain a valid iSHARE identifier
client_assertion_type	query	string	Yes	OpenID Connect 1.0 client assertion type. Used in iSHARE for all client identification for OAuth/OpenID Connect. MUST contain "urn:ietf:params:oauth:client-assertion-type:jwt-bearer"
client_assertion	query	string	Yes	OpenID Connect 1.0 client assertion. Used in iSHARE for all client identification for OAuth/OpenID Connect. MUST contain JWT conform iSHARE specifications

Example OAuth token request

```

https://randomserver.com/authorisation_registry/oauth2.0/token?
grant_type=client_credentials&
scope=iSHARE&client_id=EU.EORI.NL000000001&client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-bearer&client_assertion=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCIsIng1YyI6IjYwJmVudC51LHlQVRDQ0ErbWdBd0lCQWdJQ0VBB3dEUUVlKS29aSwH2Y05BUUVMQlFBd2daQXhDekFKQmd0VkJBWBVRBazVNTVFzd0NRWURWUVFJREFKT1NERVBNQTBHQTfVRUNnd0dhVk5JUUVZKRk1SRXdEd1lEVlFRTERBaFRaV04xY21sMGVURW9NQ1LHQTFVRUF3d2ZHVk5JUUVZKRk1FNu1JRUSyY25ScFptbGpZWJJSUVGMWRHaHZjbWwwZVRFBu1DUUdDU3FHU0liM0RRRUUpBUllyYVc1bWlWQnBjMmhoY21VdGNISnZhbVZqZEM1dmNtY3dIaGNOTVRjeE1ERXlNRGd6TXpVNVdoY05NVGd4TURJeU1EZ3pNeU1V2pDQmxURUxNQWtHQTFVRUJoTUNUa3d4Q3pBSkNtLlZCQWdNQws1SU1SSXdFQVlEVlFRSERBBEJiWE4wWlhKa1lXMhEekFOQmd0VkJBb01CbWxUU0VGU1JURVdNQlFHQTfVRUN3d05SSFZ0YlhrZ1ptOXlJRkJKUXpFVU1CSUdBMVVFQXd3TFRrd3dNREF3TURBd01ERXhKakFrQmdrcWhraUc5dzBCQ1FFV0YybhVabTlBYVh0b1lYSmxMWEJ5YjJwbFkzUXViM0puTUlJQklqQU5CZ2txaGtpRzl3MEJBUUVGQUFPQ0FROEFNSUlcQ2dLQ0FRRUF2LzVmdEldwTE0bn12MG9DUmRydEpsVk9icEV0TmptKFNNSSs3Szl05GWHl0UGFQM3c0ajB2a29Ma0lVUmJEaTJ3S290UXNqanJEM21yR1RWN1ZqdWlaSzM1WlZPMGtlczlyZUhoeTNNInxMTJ4S2x1N3QWwMjxU2I2U3hrMW94c0FxQWppOUesyemtWn055UW00NVd40FJEQzBsTytYalBpbExUZEp2VlVnWktlVkrMcl16QjV5NXc2WG90aE1VNGgyNzBoVUNibkYwZ1FyQjFzL0RPeC9yd2xqMHc5ZmRyYjlsZk9SdzE2SUM5T3B3VzdsQU5kdTFkeTY4RnpGdHdxK1BHNXJBWThXOGU3MGNiT0hSSVRERjNKSjRleEzY0NsaFlBQ3JpZUEvdKv2bnU0MzN0bm13UGVGa2VWShGhQlIzOG53RzljQWJGU2d2Sm9rM1FJREFRQUJvNElCWERDQ0FWZ3dDUVlEVlIwVEJBSXdBREFSQmdsZ2hrZ0JodmhDQVFRUJBTUNCa0F3TXdZSllJWklBWWI0UWdFTkJKDwVdKRTl3Wlc1VFUwd2dSMLZ1WlhKaGRHVmtJRk5sY25abGNpQkRaWEowYVdacFkyRjBaVEFkQmd0VkhRNEVGZ1FVZnpHMElFMEV20FBxZUNTbHJvTXVOTmlkNFA0d2diNEdBMVVKsXdTQnRqQ0JzNEFVamtaTjB4U0pxbGNpWlZTTDFQYlZtd0kzM091aGdaYWtnWk13Z1pBeEN6QUpCZ05WQkFZVEFrNU1NUNXN3Q1FZRFZRUU1EQUpPU0RFU01CQUdBdBMVVFQnd3S1FXMXpkR1Z5WkdGdE1R0HdEUVlEVlFRS0RBWnBVMGhCVWtVeEVUQVBCZ05WQkFzTUNGtmxZM1Z5YVhSNU1SUXdFZ1lEVlFRFRERBdHBVMGhCVWtVZ1VtOXZkREVtTUNRRR0NTcUdTSWIZRFFFSkFSWVhhVzVtYjBCcGMyaGhjbVv0Y0hKdmFtVmpkQzV2Y21lQ0FoQUFNQTRHQTfVZER3RUIvd1FFQXdJRm9EQVRRCZ05WSFNVRUREQUtCZ2dyQmdFRkRY0RBVEF0QmdrcWhraUc5dzBCQVZfZkF0b0N0Z0VBRGpvUVlxdjVI53pKckY5bUttUy9PQjMvM

```



```

0FodFJQVm1WRHFkTmNPdytHVTE0SXNLczQwN3ArbFhuMmh0N2VaUEpwSVpyVDdqZl2ZnVOT
G1PbzEybXZ0YLBGeTdDcT
krU1lCNkEvZ2NXK0NZemIrKzBWM3o0ZlR5Y1ZjRXVRQwc40DM5TWM2c1NhSXRha0Q3YnN1Y2
EvTldyZfVqSUU1eDBNq2Z
YQ21WTS9NeXdlnHdGdEE5ZU5yN0Z1SUN4L0JhdKvNsu8rVmRFdDBKVUp6WGNyK0wrBGM1T0F
CQlnidHjZt2pWblRHRDVu
T3ZodFdrQy80eWpaVmx3Y2EyK1E40VBGcVI20TRYRjhvc0szWFZTWGZVK2pmUk50K0EyeXpW
TpaemVucDF6a1RiZkdua
XBDek9QcLlaMGxDUzZPUKf3cDdoZ1BrS3hJb1BqTjJkTE11TzAyWVZkV2t2SFVvTXNoN0F0M
jh0RGM2TEVsTjc1Zlgxbn
VPdjgxMUcyYURxU0d6Rnc0SkdRU25lR3hsaDlMekdzaEkxaWorTm5pQm5u0TYrMGpHSm50Sj
JGY3F1Y2dpZ21tSWVvY0h
6QjLLV3Q3eFRzZmdnVFduZjd3dzA30GgvZStrNFZ3V2JITjUrcXdURHFra2poTHh6Vm9jTzR
HVWw2RzZXUmg2VHp6ekFq
SUNiS1BZclI5VwCxaGYxSFN0cG93RHU1MWlxVStmV1VZV0FRcVVSakV2Z0k0aFBmcVhya0x5
NVR60VdLQ1diR0hyRnR5Z
VlXdGpxenhXdzlpN3Fwc3d3S002M1RLQ1dVvWJoNtLJTitFSTZqczhUMTc2b2tkSkVUUFra0
XpGcjNvNHBIVERBS0VORE
hmcjJwT293emZsazlhM3F3RTkvNdxMlhalI1R289Il19.eyJpc3MiOiJFVS5FT1JJLk5MM
DAwMDAwMDAxIiwic3ViIj
oiRlUuRU9SSS50TDAwMDAwMDAwMSIsImp0aSI6InpGMXFHS0J2NzgiLCJpYXQiOiIxNTEzMD
cwODY3IiwiaXhwIjoxNTE
zMDcwODk3LkZuYmYiOjE1MTMwNzA4NjcsImF1ZCI6Ik5MLkVPUkkuTkx4MTI0NTg4MzciFQ.
bPOTFBBg8pedWkfoRKMt1
uaa0albFwkgS0bePt0SKxV0V0PUb1uencLihV086HWJeq07DBZ2jx_rn96FfpojJnn2z2aQn
BSX06IYPTYycze543-wb-
8vCor7hM6idGBbDCmeKQvFrIYaYmt34GeU0UjWnNMPGdh90vzbhqqPULzixtUWfQYn0NxYJf
7RGmEhmRybXm2zFl0oom
1d-
zoZzwuTAzfZqa2rM986VG8WikewxN2IUafhKoQ_w42MB6WpPki8a0EJ07xUZozSybSQvFRWy
KxN-TCtixp3B5nGo9T
uZvk0f1f0RpL8-zTU2DQ0Fnhz8p7gwF10srNYYv3Sw

```

Responses

Code	Description
------	-------------

200	OK
-----	----

Example value

<pre> { "access_token": "string", "token_type": "string", "expires_in": 3600 } </pre>

OAuth 2.0 general description

OAuth is an open standard for authorisation which is used by i.e. Google, Facebook, Microsoft, Twitter etc. to let their users exchange information about their accounts with other applications or websites. OAuth is designed to work with HTTP.

Through OAuth users can authorise third party applications or websites to access their account information on other "master" systems without the need of exchanging with them their credentials to login onto the platform. OAuth provides a "secure delegated access" to resources (email accounts, pictures accounts, etc.) on behalf of the resource owner.

It specifies a method for resource owners to authorise third parties access to their resources without exchanging their credentials (username, password). Authorisation servers (of the platform) issue access tokens to third party clients (applications or websites) with the approval of the resource owner (= end user). The third party client needs the access token to get access to the resources that are stored on the resource server (of the master system).

OpenID Connect 1.0

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

Besides Machine2Machine interaction, it can occur that it is relevant if a specific person requests data or a service. In order to provide a Service Provider with identity information on a human subject, iSHARE uses the OpenID Connect 1.0 protocol.

The iSHARE use of OpenID Connect 1.0 is based on the requirements from the [official standard](#).

iSHARE facilitates an ecosystem within which parties can interact with previously unknown parties, pre-registration is therefore not a prerequisite and thus requires alterations to the official standard.

Generic OpenID Connect 1.0 requirements

In addition to the endpoint specifications described in the [iSHARE Developer Portal](#), for all uses of OpenID Connect 1.0 the following requirements apply:

- Clients MUST NOT be pre registered. See [Generic/oauth2.0/token](#) for more details.
- The `client_id` MUST contain the valid [iSHARE identifier](#) of the client
- For interoperability reasons clients SHALL only make HTTP GET calls to the `/oauth2.0/token` endpoint.
- Servers SHALL NOT issue refresh tokens

Description

OpenID Connect (OIDC) is the authentication layer that is built on top of OAuth 2.0 protocol which is an authorization framework. The OIDC authentication layer allows clients to verify the ID and obtain basic profile information of their end-users

The authentication is performed by the authorization server (managing the access rights and conditions) in an interoperable and REST-like manner.

OpenID Connect's building blocks

OIDC specifies a RESTful HTTP API using JSON as data format.

REST (Representational state transfer) or RESTful web services provide a method to achieve interoperability between computer systems and the internet.

APIs (Application Programming interfaces) enable Machine to Machine (M2M) communication where one machine calls upon the software functionality of another machine. They facilitate connectivity between applications. It is a software architectural approach that revolves around the view on digital interfaces that APIs provide self-service, one-to-many, reusable interfaces.

With OIDC a broad range of clients (web-based, mobile, JavaScript) can request and receive data about authentication sessions end-user profiles.

The specification is extensible (meaning it takes future growth into consideration) and supports optional features for encryption, ID data, discovery of OpenID providers and session management

OpenID Connect 1.0

OpenID Connect 1.0 is an adapted version of OpenID, combined with OAuth 2.0.

OpenID Connect performs many of the same tasks as OpenID 2.0, but in an API-friendly way and usable by native and mobile applications.

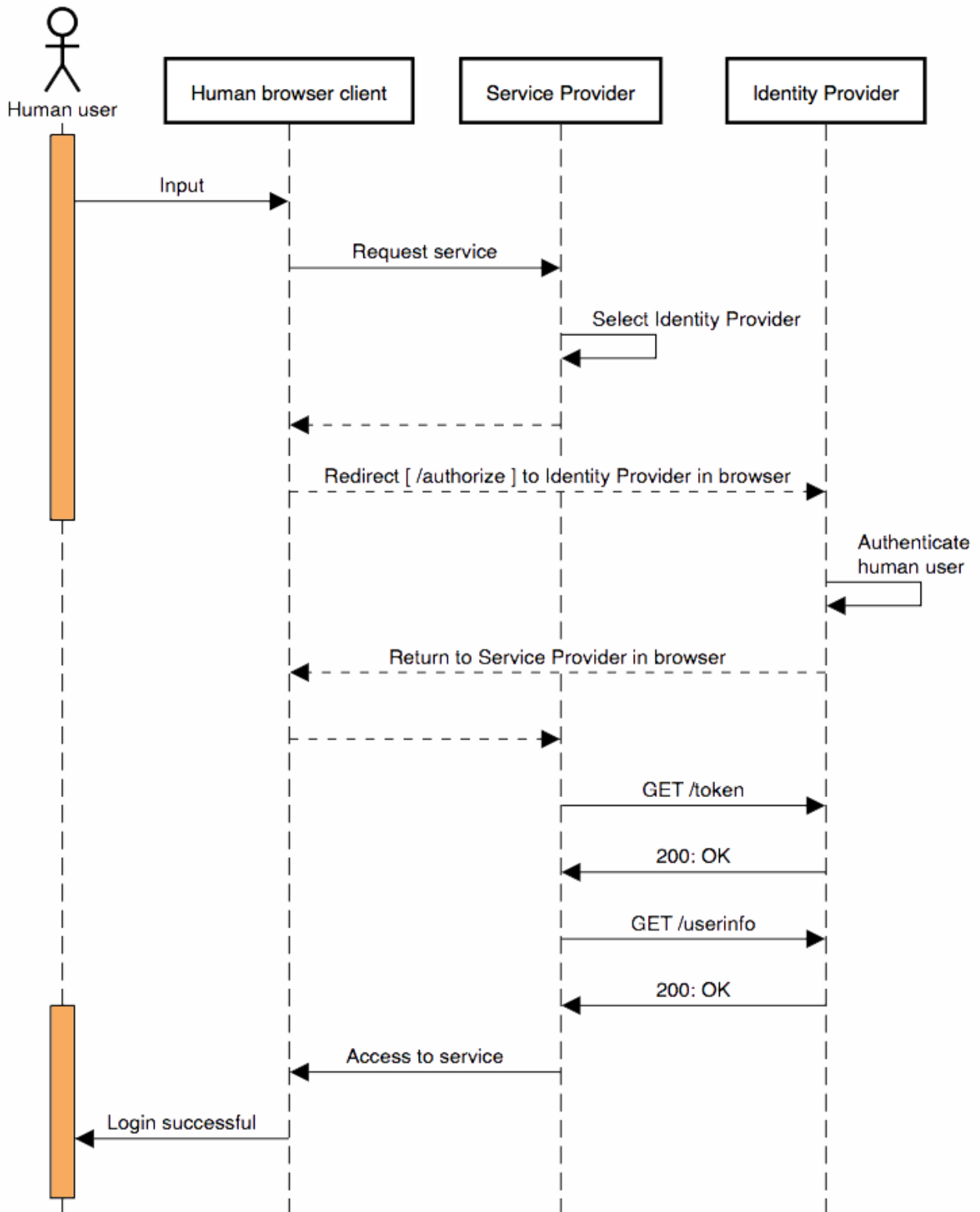
OpenID Connect defines optional mechanisms for robust signing and encryption.

Whereas the integration of OAuth 1.0a with OpenID 2.0 required an extension, in OpenID Connect, OAuth 2.0 capabilities are integrated with the protocol itself.

iSHARE H2M authentication flow

Based on the described standards and specifications in this scheme, the generic iSHARE Human2Machine Authentication flow is described in the following sequence diagram.

Generic H2M authentication flow



iSHARE Identity JWTs

The OpenID Connect 1.0 flow contains 2 important iSHARE-specific JWTs, which are described in more detail in [this section](#) of the developer portal.



Authorisation in OpenID Connect flow

The generic OpenID Connect 1.0 flow does not take into account Authorisations of a human. However, in iSHARE it is essential that authorisations of a user are combined with their identity details before a service can be offered. This authorisation flow is heavily dependent on the pseudonym used to refer to humans without exposing their identity. This section of the scheme is under construction and parties wishing to implement authorisations of a user are advised to contact support@ishareworks.org.

PKI

For authentication purposes, iSHARE requires adhering and Certified Parties to acquire an X.509 certificate which is distributed by a trusted root under certain PKIs (Public Key Infrastructure). For interoperability on a European scale, all trusted roots under the eIDAS regulation will be trusted within iSHARE. However, initially, this will be limited to certificates issued under PKIoverheid.

Brief description

A PKI is a system for distribution and management of digital keys and certificates, which enables secure authentication of parties interacting with each other.

Generally, three different methods exist for creating trust within PKI's. These are through 'Certificate Authorities', 'Web of Trust' and 'Simple PKI'. Within iSHARE the 'Certificate Authority' approach is used, and as such the other methods will not be discussed.

A PKI can be considered as a chain of certificates. At the beginning of the chain is the root 'Certificate Authority' (CA), a public trusted party which is allowed to digitally sign their own certificates (SSC, self-signed certificate). This 'Root CA' distributes certificates and encryption keys to organisations. The certificate is signed by the 'root CA' as proof that the owner of the certificate is trusted. These organisations can start distributing certificates as well, if allowed by their root. They become CA's, and as such sign the certificates that they distribute. Repeating these steps, a chain of certificates is created, with each certificate signed by the CA who distributed the certificate.

Parties need to trust a certificate for authentication purposes. Instead of trusting individual certificates of organisations, root certificates can be trusted. By trusting a root, all certificates that have the root within their PKI chains are automatically trusted. Most large root CA's are automatically trusted within web browsers, enabling computers to safely interact with most web servers.

Trusted roots and eIDAS

In the future, iSHARE aims to support digital certificates that are recognized under eIDAS as Qualified Certificates. The eIDAS regulation aims to provide secure and seamless electronic interactions between businesses, citizens and public authorities throughout the entire European Union. A main part of this regulation is that each EU country is required to establish and maintain 'trusted lists', among which trusted root information is found. Each EU country is required to implement these trusted lists in their own countries. Therefore, iSHARE aims to make use of these trusted lists as trust roots within iSHARE to ensure secure and seamless interaction throughout the entire EU.

However, during the initial phase of iSHARE, the use of digital certificates will be limited to PKIoverheid certificates only.

TLS 1.2

HTTP communication within iSHARE is encrypted using the TLS 1.2 protocol.

On this page a brief description of TLS is provided. For the most recent version of the specification click on [this link](#).

Description

Transport Layer Security (TLS) is a cryptographic protocol that describes communication security for computer networks. The first version of TLS 1.0 is built upon and is an upgrade of SSL 3.0 (Secure Sockets Layer).

Differences and similarities between TLS and SSL

Both TLS and SSL provide means for data encryption and authentication between applications, machines and servers when data is sent through insecure network.

The differences between TLS and its forerunner 'Secure Sockets Layer' (SSL) are the addressed vulnerabilities. TLS for instance works with

- a wider variety of hash functions.
- more secure and stronger cipher suites, such as the Advanced Encryption Standard (AES) cipher suites which are integrated into TLS version 1.1.
- browser security warnings. TLS has more alert descriptions than SSL.

XACML 3.0

Within iSHARE, it is essential to provide fine-grained authorization. Besides rules on the authorization, it is important to have varying options to describe the resources and its attributes to which the rules apply.

XACML 3.0 is a specification for describing such authorization rules, but it is XML-based. For iSHARE, a JSON port was created for expressing the XACML specifications regarding authorization. This 'delegation evidence structure' is discussed in more detail in the chapter on [delegation evidence structure](#).

On this page a brief description of XACML is provided. For the most recent version of the specification click on [this link](#).

Description

XACML (eXtensible Access Control Markup Language) is an XML-based specification that is designed to control access to applications. One of the main advantages of this specification is that applications and systems with their own and different authorization structure can be integrated into one authorization scheme. Authorization and the rules surrounding it can be managed centrally regardless of authorization mechanism of the applications themselves. This phenomenon is called externalisation. XACML is derived from SAML and provides the underlying specification for ABAC (Attribute-Based Access Control). XACML is also suitable to be used in combination with RBAC (Role-Based Access Control).

Moreover, with the help of XACML authorization can be arranged and managed in detail. This is called fine-grained authorization. XACML supports the use of security labels, rules with arbitrary attributes, rules with a certain duration and dynamic rules.

In XACML two main functions can be distinguished. One function defines the criteria with which authorization are assigned, such as 'only an experienced user from department X is allowed to modify documents'. The other function compares the criteria with the rules or policies to determine whether a person is allowed to perform the operation on the object or not.

The architecture of XACML is fairly complex. This is partly due to the fact that it is difficult to fit the various components of XACML in the application landscape. These components should be positioned in such a way that the owner of the data can somehow control the authorization to his or her data, but at the same time the components should be positioned in such a way that the performance is not negatively influenced. This is extra important when independent parties need to cooperate with each other and want to jointly organise the access to their applications. Finally, applications need to be compatible with XACML.

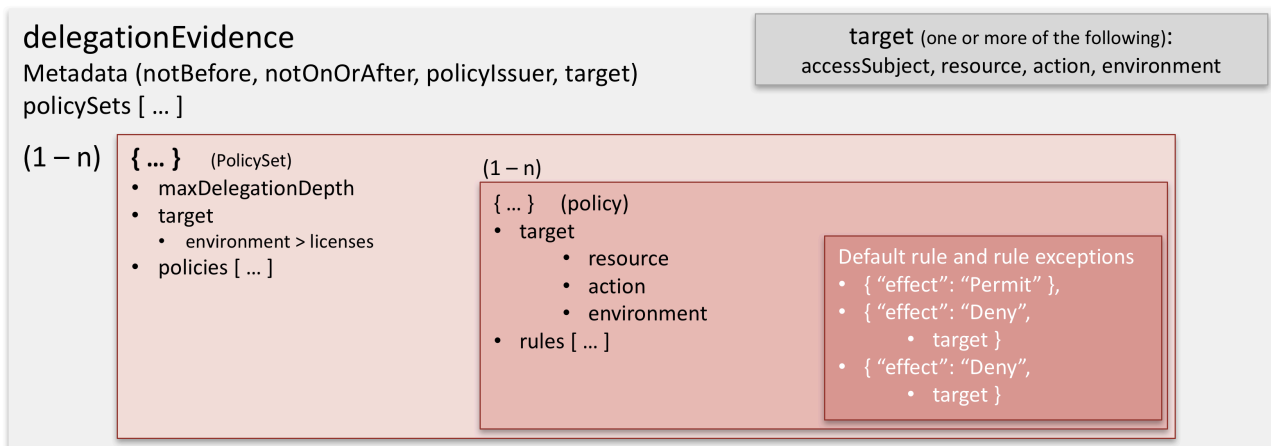
Structure of delegation evidence

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

This page describes (and prescribes) how, in iSHARE, delegation is communicated between different parties.

In iSHARE, delegation evidence expresses the delegation of rights from a delegator (the party that delegates rights; the policyIssuer) to the delegate (the party that receives the delegated rights; i.e. the accessSubject). Rights are expressed in rules in terms of allowed actions to be performed on resources, under the license(s) as defined in policySets.

Delegation evidence is modeled as a JSON object inspired by the XACML 3.0 specifications and structured as follows:



The JSON object consists of a root delegationEvidence element (modeled after an XACML PolicySet element) containing one or more policySet objects in the policySets array. The root element is only meant as a container element and extends the XACML specifications to cater for some iSHARE required metadata such as timestamps. Each of the second level policySet elements only acts as a container for actual policy elements with an indication of the rights in this policySet can be further delegated (with maxDelegationDepth) and what license(s) do apply. No other delegation logic is conveyed a second level policySet. Each policy element is used to express the actual rights being delegated.

The root delegationEvidence element contains the following parameters.

Parameter	Contained in	Type	Required	Description
delegationEvidence		{ }	Yes	The root of any delegation evidence

Parameter	Contained in	Type	Required	Description
notBefore	delegationEvidence	int	Yes	Unix timestamp in UTC indicating the start of validity period of this delegation evidence. SHOULD equal the time of issuing of the evidence unless historic evidence is requested.
notOnOrAfter	delegationEvidence	int	Yes	Unix timestamp in UTC indicating the end of validity period of this delegation evidence. It is up to the issuer off the evidence to set this time. Note that a reasonable amount of time SHOULD be allowed for processing of longer delegation paths. Also note that evidence cannot be revoked, so setting very long validity periods SHOULD be avoided.
policyIssuer	delegationEvidence	string	Yes	iSHARE identifier of the delegator (the delegating entity)
target	delegationEvidence	{}	Yes	MUST for the root level contain an accessSubject. No other elements are allowed. It makes the entire delegation evidence applicable only to this accessSubject.
accessSubject	target	string	Yes	iSHARE identifier of the delegate (the entity that receives the delegated rights)
policySets	delegationEvidence	[]	Yes (1..n)	Container for one or more objects containing policy elements with an indication for further delegation. Note that policySet elements within one delegationEvidence MUST not restrict each other, but rather offer a mechanism to express additional rights. They MUST be evaluated in a "permit-override" manner, allowing a "Permit" if only one of the policySet elements evaluates to "Permit".

The second level objects in policySets each contain the following parameters. Other parameters are not allowed. Note that XACML spec is heavily restricted, a.o. for the reason to prevent redundancy (and resulting possible conflicts) with the root policySet element.

Parameter	Contained in	Type	Required	Description
maxDelegationDepth	policySets	int	No	Optional element that, if present, indicates that further delegation of the rights, conveyed in the policy elements that are part of this PolicySet, is allowed. The value indicates the delegation steps that are allowed after this step in order to evaluate the entire delegation path to "Permit"
target	policySet	{}	Yes	

Parameter	Contained in	Type	Required	Description
environment	target	{}	Yes	
licenses	environment	[]	Yes	Array which describes which iSHARE licenses apply to this policySet.
policies	policySets	[]	Yes (1..n)	Used to express the actual rights being delegated. Note that policies within one policySets object MUST not restrict each other, but rather offer a mechanism to express additional rights. They MUST be evaluated in a "permit-override" manner, allowing a "Permit" if only one of the policy elements evaluates to "Permit".

A Policy element contains the following parameters.

Parameter	Contained in	Type	Required	Description
target	policies	string	Yes	Describes the target, in terms of resource and action, this policy applies to. It is also the scope that is permitted through the default Rule. Additional Rule elements can be described to exclude Resources and Actions from the default policy rights
resource	target	{}	Yes	
type	resource	string	Yes	String which describes the type of resource to which the rules apply.
identifiers	resource	[]	Yes	Array of strings containing one or more resource identifiers. Depending on the Type an identifier SHOULD be a urn.
attributes	resource	[]	No	Optional array of attributes of the resources the delegated rights apply to. If omitted defaults to all attributes. Depending on the Type an attribute SHOULD be a urn.
actions	target	[]	Yes	
environment	target	{}	No	
serviceProviders	environment	[]	Yes	Array which lists the iSHARE client ID's of serviceProviders which are allowed to provide services to the accessSubject as described within this policy.

Parameter	Contained in	Type	Required	Description
rules	policies	[]	Yes (1..n)	The first rule element is the default rule that applies to the target at policies level. Note that additional rule elements within one policies object are intended to restrict each the default rule. All rule elements in a Policy MUST be evaluated in a "deny-override" manner, allowing a "Permit" only if all of the rule elements evaluate to "Permit".

The default Rule element contains the following parameters.

Parameter	Contained in	Type	Required	Description
effect	rules	string	Yes	MUST contain 'Permit'

Additional Rule elements contains the following parameters.

Parameter	Contained in	Type	Required	Description
effect	rules	string	Yes	MUST contain 'Deny'
target	rules	{}	Yes	Describe the target, in terms of resource and action, this additional rule applies to. Additional rule elements are limitations of the default rule and resource scope.
resource	target	{}	Yes	
type	resource	string	No*	Optional string which describes the type of resource to which the rule applies. Defaults to none if not specified.
identifiers	resource	[]	No*	Optional array of strings containing one or more resource identifiers. Depending on the type an identifier SHOULD be a urn.
attributes	resource	[]	No*	Optional array of attributes of the resources the delegated rights apply to. If omitted defaults to all attributes. Depending on the type an attribute SHOULD be a urn.
actions	target	[]	No	Optional array of actions, the additional rule applies to the actions listed. If no actions are listed then the default is to all iSHARE actions defined within the policy.

* Note: Although not individually required, at least one of the parameters within the resource object needs to be specified to which the additional rules apply.

Example delegation JSON

//Organisation A delegates rights to organisation B. A allows B READ and CREATE access to all ETA and WEIGHT of A's containers of which the data is located at service provider C and can only be accessed with service provider C. However, A does not allow B to CREATE to ETA information and completely denies access to data regarding container ID.00000000000001. Furthermore, all rights of B are allowed under iSHARE licenses 1 and 3, and B has the right to delegate it's right two more times.

```
{
  "delegationEvidence": {
    "notBefore": 1509633681,
    "notOnOrAfter": 1509633741,
    "policyIssuer": "EU.EORI.NL123456789",
    "target": {
      "accessSubject": "EU.EORI.NL012345678"
    }
  },
  "policySets": [
    {
      "maxDelegationDepth": 2,
      "target": {
        "environment": {
          "licenses": ["ISHARE.0001", "ISHARE.0003"]
        }
      },
      "policies": [
        {
          "target": {
            "resource": {
              "type": "GS1.CONTAINER",
              "identifiers": ["*"],
              "attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA",
"GS1.CONTAINER.ATTRIBUTE.WEIGHT"]
            },
            "actions": ["ISHARE.READ", "ISHARE.CREATE"],
            "environment": {
              "serviceProviders": ["EU.EORI.NL123412345"]
            }
          },
          "rules": [
            {
              "effect": "Permit"
            },
            {
              "effect": "Deny",
              "target": {
                "resource": {
                  "attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA"]
                },
                "actions": ["ISHARE.CREATE"]
              }
            }
          ]
        }
      ]
    }
  ]
}
```

```

    },
    {
      "effect": "Deny",
      "target": {
        "resource": {
          "identifiers": ["GS1.CONTAINER.ID.
00000000001"]
        }
      }
    }
  ]
}

```

example code - for copying purposes

```

{"delegationEvidence":{"notBefore":1509633681,"notOnOrAfter":
1509633741,"policyIssuer":"EU.EORI.NL123456789","target":
{"accessSubject":"EU.EORI.NL012345678"},"policySets":[{"maxDelegationDepth":
2,"target":{"environment":{"licenses":["ISHARE.0001","ISHARE.0003"]}}},"policies":
[{"target":{"resource":{"type":"GS1.CONTAINER","identifiers":["*"],"attributes":
["GS1.CONTAINER.ATTRIBUTE.ETA","GS1.CONTAINER.ATTRIBUTE.WEIGHT"]},"actions":
["ISHARE.READ","ISHARE.CREATE"],"environment":{"serviceProviders":
["EU.EORI.NL123412345"]}}},"rules":[{"effect":"Permit"},{"effect":"Deny","target":
{"resource":{"attributes":["GS1.CONTAINER.ATTRIBUTE.ETA"]},"actions":
["ISHARE.CREATE"]}}, {"effect":"Deny","target":{"resource":{"identifiers":
["GS1.CONTAINER.ID.00000000001"]}}}}]]}}]}

```

Please note that although in XACML the attributes PolicySetId, Version and PolicyCombiningAlgId are mandatory in XACML they are not ported to the iSHARE JSON structure. iSHARE follows the **"deny-override"** Policy Combining Algorithm. This implies that if at least one policy is evaluated as “deny”, the integrated output must also be “deny”.

Example cases

The main variations in the JSON code for `delegationEvidence` are the (1-n) `policySets`, `policies` and `rules` arrays. These variations are based on the most efficient way of expressing the rights that an `accessSubject` has.

Various examples are described in the table below.

Description	Code
<p>Organisation A delegates rights to organisation B. A allows B READ and CREATE access to all ETA and WEIGHT of A's containers of which the data is located at service provider C and can only be accessed with service provider C. However, A does not allow B to CREATE to ETA information and completely denies access to data regarding container ID. 00000000000001. Furthermore, all rights of B are allowed under iSHARE licenses 1 and 3, and B has the right to delegate it's right two more times.</p> <p>The code shows default access to a set of resources, with a few exceptions in terms of actions or specific resources. This results in additional "Deny" rules within the policy.</p>	<div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p>Code - for visual/reading purposes</p> </div> <pre> { "delegationEvidence": { "notBefore": 1509633681, "notOnOrAfter": 1509633741, "policyIssuer": "EU.EORI.NL123456789", "target": { "accessSubject": "EU.EORI.NL012345678" } }, "policySets": [{ "maxDelegationDepth": 2, "target": { "environment": { "licenses": ["ISHARE.0001", "ISHARE.0003"] } }, "policies": [{ "target": { "resource": { "type": "GS1.CONTAINER", "identifiers": ["*"], "attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA", "GS1.CONTAINER.ATTRIBUTE.WEIGHT"] }, "actions": ["ISHARE.READ", "ISHARE.CREATE"], "environment": { "serviceProviders": ["EU.EORI.NL123412345"] } }, "rules": [{ "effect": "Permit" }, { "effect": "Deny", "target": { "resource": { "attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA"] }, "actions": ["ISHARE.CREATE"] } }] }] }] } </pre>

Description	Code
	<pre> }, { "effect": "Deny", "target": { "resource": { "identifiers": ["GS1.CONTAINER.ID.00000000001"] } } }] } </pre>
	<p data-bbox="427 958 762 992">Code - for copying purposes</p> <pre> {"delegationEvidence":{"notBefore":1509633681,"notOnOrAfter": 1509633741,"policyIssuer":"EU.EORI.NL123456789","target": {"accessSubject":"EU.EORI.NL012345678"},"policySets": [{"maxDelegationDepth":2,"target":{"environment":{"licenses": ["ISHARE.0001","ISHARE.0003"]},"policies":[{"target":{"resource": {"type":"GS1.CONTAINER","identifiers":["*"],"attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA","GS1.CONTAINER.ATTRIBUTE.WEIGHT"]}," actions":["ISHARE.READ","ISHARE.CREATE"],"environment": {"serviceProviders":["EU.EORI.NL123412345"]}}],"rules": [{"effect":"Permit"}, {"effect":"Deny","target":{"resource": {"attributes":["GS1.CONTAINER.ATTRIBUTE.ETA"]},"actions": ["ISHARE.CREATE"]}], {"effect":"Deny","target":{"resource": {"identifiers":["GS1.CONTAINER.ID.00000000001"]}}]}]}]}]} </pre>

Description	Code
<p>Organisation A delegates rights to organisation B. A allows B READ access to all ETA of A's containers of which the data is located at service provider C and can only be accessed with service provider C. A also allows B CREATE access to all WEIGHT of A's containers, at any service provider possible. Furthermore, all rights of B are allowed under iSHARE licenses 1 and 3, and B has the right to delegate it's right two more times.</p> <p>The code shows that the same delegation rights and licenses apply to a resource set, but different actions are allowed to different subsets of these resources. This results in variations in policies within the policySets.</p>	<div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p>Code - for visual/reading purposes</p> </div> <pre> { "delegationEvidence": { "notBefore": 1509633681, "notOnOrAfter": 1509633741, "policyIssuer": "EU.EORI.NL123456789", "target": { "accessSubject": "EU.EORI.NL012345678" } }, "policySets": [{ "maxDelegationDepth": 2, "target": { "environment": { "licenses": ["ISHARE.0001", "ISHARE.0003"] } }, "policies": [{ "target": { "resource": { "type": "GS1.CONTAINER", "identifiers": ["*"], "attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA"] }, "actions": ["ISHARE.READ"], "environment": { "serviceProviders": ["EU.EORI.NL123412345"] } }, "rules": [{ "effect": "Permit" }] }, { "target": { "resource": { "type": "GS1.CONTAINER", "identifiers": ["*"], "attributes": ["GS1.CONTAINER.ATTRIBUTE.WEIGHT"] }, </pre>

Description	Code
	<pre data-bbox="405 327 1422 779"> "actions": ["ISHARE.CREATE"] }, "rules": [{ "effect": "Permit" }] }] } </pre> <div data-bbox="405 801 1422 891" style="border: 1px solid #ccc; background-color: #f9f9f9; padding: 5px; margin-top: 10px;"> <p>Code - for copying purposes</p> <pre data-bbox="405 913 1422 1323"> {"delegationEvidence":{"notBefore":1509633681,"notOnOrAfter":1509633741,"policyIssuer":"EU.EORI.NL123456789","target":{"accessSubject":"EU.EORI.NL012345678"},"policySets":[{"maxDelegationDepth":2,"target":{"environment":{"licenses":["ISHARE.0001","ISHARE.0003"]}}},"policies":[{"target":{"resource":{"type":"GS1.CONTAINER","identifiers":["*"],"attributes":["GS1.CONTAINER.ATTRIBUTE.ETA"]},"actions":["ISHARE.READ"],"environment":{"serviceProviders":["EU.EORI.NL123412345"]}}},"rules":[{"effect":"Permit"}]},{"target":{"resource":{"type":"GS1.CONTAINER","identifiers":["*"],"attributes":["GS1.CONTAINER.ATTRIBUTE.WEIGHT"]},"actions":["ISHARE.CREATE"]},"rules":[{"effect":"Permit"}]}]}]} </pre> </div>

Description	Code
<p>Organisation A delegates rights to organisation B. A allows B READ and CREATE access to all ETA and WEIGHT of A's containers of which the data is located at service provider C, and rights can only be used with service provider C. Furthermore, all rights of B are allowed under iSHARE licenses 1 and 3, and B has the right to delegate it's right two more times. A also provides B READ access to the Container origins, but does not allow delegation for this information and it is only accessible under iSHARE license 2.</p> <p>The code shows two groups of resources with specific policies, executed under different licenses and delegation rights. This results in variations on the policySets level within the delegationEvidence.</p>	<div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p style="text-align: center; margin: 0;">Code - for visual/reading purposes</p> </div> <pre> { "delegationEvidence": { "notBefore": 1509633681, "notOnOrAfter": 1509633741, "policyIssuer": "EU.EORI.NL123456789", "target": { "accessSubject": "EU.EORI.NL012345678" }, "policySets": [{ "maxDelegationDepth": 2, "target": { "environment": { "licenses": ["ISHARE.0001", "ISHARE.0003"] } }, "policies": [{ "target": { "resource": { "type": "GS1.CONTAINER", "identifiers": ["*"], "attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA", "GS1.CONTAINER.ATTRIBUTE.WEIGHT"] }, "actions": ["ISHARE.READ", "ISHARE.CREATE"], "environment": { "serviceProviders": ["EU.EORI.NL123412345"] } }, "rules": [{ "effect": "Permit" }] }] }], "target": { "environment": { "licenses": ["ISHARE.0002"] } } } } </pre>

Description	Code
	<pre> }, "policies": [{ "target": { "resource": { "type": "GS1.CONTAINER", "identifiers": ["*"], "attributes": ["GS1.CONTAINER.ATTRIBUTE.ORIGIN"] }, "actions": ["ISHARE.READ"] }, "rules": [{ "effect": "Permit" }] }] } </pre>
	<p data-bbox="427 1182 762 1216">Code - for copying purposes</p> <pre> {"delegationEvidence":{"notBefore":1509633681,"notOnOrAfter": 1509633741,"policyIssuer":"EU.EORI.NL123456789","target": {"accessSubject":"EU.EORI.NL012345678"},"policySets": [{"maxDelegationDepth":2,"target":{"environment":{"licenses": ["ISHARE.0001","ISHARE.0003"]}},"policies":[{"target":{"resource": {"type":"GS1.CONTAINER","identifiers":["*"],"attributes": ["GS1.CONTAINER.ATTRIBUTE.ETA","GS1.CONTAINER.ATTRIBUTE.WEIGHT"]}," actions":["ISHARE.READ","ISHARE.CREATE"],"environment": {"serviceProviders":["EU.EORI.NL123412345"]}},"rules": [{"effect":"Permit"}]}]},{"target":{"environment":{"licenses": ["ISHARE.0002"]}},"policies":[{"target":{"resource": {"type":"GS1.CONTAINER","identifiers":["*"],"attributes": ["GS1.CONTAINER.ATTRIBUTE.ORIGIN"]},"actions": ["ISHARE.READ"]},"rules":[{"effect":"Permit"}]}]}]}]} </pre>

Operational

This section covers the Operational details of the iSHARE Scheme.

The Scheme Owner facilitates the correct operation of the iSHARE Scheme and the iSHARE network through administering several aspects:

- [Operational processes](#)
- [Service levels](#)
- [Communication](#)

The Scheme Owner is part of a wider governance framework, which can be found in the [introduction of the scheme](#).

The assumptions underlying the processes and service levels can be found [here](#).

Operational processes

This section describes the operational processes necessary to administer the iSHARE Scheme (specifications), network and brand.

Per process described in this section, the goal and responsibilities (per party) are described, before a process sequence is included.

The following processes are described:

- [Admission](#)
- [Withdrawal](#)
- [Warnings, Suspension and Exclusion](#)
- [Incident Management](#)
- [Release Management](#)
- [Management reporting](#)

Admission

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

This admission process describes the steps that all parties **MUST** take to be admitted to the iSHARE network. For Certified Parties, additional steps are required and are described below. The process is the responsibility of, and facilitated by, the Scheme Owner.

Admission of prospective iSHARE participant includes:

- A potential Adhering Party wants to start fulfilling one or more adhering role(s) in the network.
- A potential Certified Party wants to start fulfilling one or more certified roles(s) in the network.
- An already Adhering and/or Certified Party wants to expand its current role(s) by one or more role(s) in the network.

Note: prospective iSHARE participants can start testing at any moment in time before initiating the admission process.

Goal

The goal of the admission process is to let prospective iSHARE participants join iSHARE in a simple and controlled way. A controlled admission process is important to warrant trust in the iSHARE Scheme. It provides assurance that all parties signing an accession agreement fulfil the scheme's accession criteria.

Admission criteria

To be admitted to the iSHARE network, prospective iSHARE participants need to comply with several criteria*:

- Provide a signed iSHARE accession agreement;
- Provide a valid EORI number (or other nationally recognised chamber of commerce number which can be verified);
- Provide a [PKIOverheid](#) or Qualified Electronic Seal (under [Admission#eIDAS](#)) that will be used for iSHARE;
- Provide a successful test report of iSHARE certification tool (or otherwise acceptable proof that party's technical implementation adheres to iSHARE specifications, e.g. in case of using 'iSHARE compatible' software).

* For operational reasons, additional admission criteria **MAY** apply.

Responsibilities

Several parties have responsibilities and tasks in the admission process:

- The **Scheme Owner** is responsible for facilitation of the process while safeguarding the integrity of the iSHARE Scheme;
- The prospective iSHARE participant is responsible for implementing the guidelines set out in iSHARE, and for showing compliance with the relevant admission criteria.

Sequence

1. A representative of the prospective iSHARE participant registers with the Scheme Owner and provides the Scheme Owner with:
 - a. Primary contact details: name, role, e-mail.
 - b. Description of how they will be using iSHARE
 - c. A valid EORI number (or other nationally recognised chamber of commerce number which can be verified);

2. The Scheme Owner checks whether there are potential impediments that could block the completion of the admission process for the prospective iSHARE participant:
 - a. E.g. previous exclusions from the iSHARE Scheme in the recent past;
3. The Scheme Owner will facilitate testing and certification of the prospective iSHARE participant.
 - a. The Scheme Owner may provide testing material and documentation on the Scheme test environment: certificates, keys, SDKs, etc.;
 - b. For prospective Certified Parties this will include role-specific non-technical requirements.
4. The prospective iSHARE participant formally requests admission to the iSHARE network to the Scheme Owner by providing:
 - a. An iSHARE accession agreement signed by an authorized representative of the prospective iSHARE participant;
 - b. The certificate that you will use for iSHARE;
 - c. Acceptable proof that the prospective participant's technical implementation adheres to iSHARE specifications;
 - d. For a prospective Certified Party: The level of assurance for which the prospective Certified Party wants to be certified, accompanied by a filled in [Assessment Framework](#) (and related evidence) to prove that the operational processes of the prospective Certified Party comply with the indicated level of assurance.
 - i. If desired, a [standard NDA](#) can be signed before providing the Assessment Framework and related evidence. Please request a signed NDA at info@ishareworks.org.
 - e. For prospective Certified Parties additional verification may be required.
5. The Scheme Owner verifies the acceptance of the prospective participant's admission request and its conformance with the admission criteria;
 - a. For prospective Adhering Parties the Scheme Owner has 5 working days to verify the acceptance of the prospective participant's admission request and its conformance with the admission criteria;
 - b. For prospective Certified Parties the Scheme Owner has 30 days to verify the acceptance of the prospective participant's admission request and its conformance with the admission criteria, but aims to respond and soon as possible;
6. A legal representative of the Scheme Owner signs the participant's iSHARE accession agreement, and communicates the verified acceptance to the new participant;
7. The Scheme Owner records the participant's status within the scheme in the iSHARE participant registry.

Withdrawal

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

The withdrawal process describes the steps that parties **MUST** take to withdraw from the iSHARE network.

Withdrawal includes:

- A Certified and/or Adhering Party wants to withdraw from the iSHARE network;
- A Certified Party wants to downgrade to an Adhering Party;
- Any other situation in which an Adhering or Certified Party (un)expectedly withdraws from the iSHARE network (e.g. bankruptcy).

The **term of notice** for withdrawal is 1 month for Adhering Parties, and 6 months for Certified Parties.

Goal

The goal of the withdrawal process is to let parties withdraw from iSHARE in a simple and controlled way, minimising impact to the trust in the iSHARE Scheme and disruption to the functioning of the iSHARE network.

Responsibilities

Several parties have responsibilities and tasks in the withdrawal process:

- The **Scheme Owner** is responsible for facilitation of the process, so that continued operation of the iSHARE network is not disrupted in any way;
- The **withdrawing/downgrading party** is responsible for delivering a withdrawal plan and to minimise the disruption to the functioning of the iSHARE network. The withdrawing party also benefits from a controlled process itself, as it should help to minimise disruption to internal operations.

Sequence

Withdrawal of an Adhering Party

1. The withdrawing party formally indicates its intention to withdraw from the iSHARE network to the Scheme Owner.
2. The Scheme Owner has 5 working days to acknowledge the intention to withdraw of the withdrawing party; the Scheme Owner makes the acknowledgement known to the withdrawing party, and provides a date on which the withdrawing party will be considered withdrawn from the iSHARE scheme by the Scheme Owner;
3. The withdrawing party communicates its withdrawal to the parties it interacts (interacted) with under iSHARE;
4. The withdrawing party, in cooperation with the Scheme Owner, withdraws from the iSHARE network.

Withdrawal of a Certified Party, downgrade of a Certified Party

1. The withdrawing/downgrading party formally indicates its intention to withdraw/downgrade from the iSHARE network to the Scheme Owner. It includes a withdrawal plan based on the (to be set up) template withdrawal procedure;
2. The Scheme Owner has 5 working days to acknowledge the intention to withdraw/downgrade of the withdrawing/downgrading party; the Scheme Owner makes the acknowledgement known to the withdrawing/downgrading party, and provides up to date guidelines;
3. If necessary, the withdrawing/downgrading party sends an updated withdrawal plan to the Scheme Owner, keeping in mind the guidelines provided by the Scheme Owner;
4. The Scheme Owner accepts the withdrawal/downgrading plan or indicates where it requires changes;

5. The Scheme Owner and the withdrawing/downgrading party communicate the intended withdrawal with the iSHARE network per date dd-mm-yyyy;
6. The withdrawing/downgrading party, in cooperation with the Scheme Owner, withdraws/downgrades from the iSHARE network in accordance with the withdrawal plan;
7. The Scheme Owner communicates the completed withdrawal to the iSHARE network.

Warnings, Suspension and Exclusion

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

The warnings, suspension and exclusion process describes the steps that the Scheme Owner MUST take to temporarily suspend or permanently exclude participating parties from the iSHARE network in case of non-compliance with scheme rules and guidelines, or actions with significant negative impact on the normal operation of the iSHARE network.

Three classifications of non-compliance are recognised within iSHARE. Note that the impact or risk described is non-exhaustive.

Classification	Impact or risk
Minor non-compliance	<ul style="list-style-type: none"> • Non-compliance with the iSHARE admission criteria, and/or; • Non-compliance with the iSHARE service levels, and/or; • Expired information security certification (e.g. ISO27001, ISAE 3402), and/or; • Minor data* security breach, for example through the loss of a USB stick, laptop, hard disk, or because of hacking attempts or found malware, and/or; • Fraud or presumption of fraud by, for example an employee or a hacker.
Major non-compliance	<ul style="list-style-type: none"> • Recurring minor non-compliance, and/or; • Combinations of minor non-compliance, and/or • Serious impediment(s) to other Adhering/Certified Party(s), and/or; • Major data security breach and/or breach that needs to be reported in line with meldplicht datalekken, and/or; • (Other) impact on confidentiality and integrity of (data* within) the iSHARE Scheme.
Critical non-compliance	<ul style="list-style-type: none"> • Recurring major non-compliance, and/or; • Network-wide impediment(s) to other parties, and/or; • (Other) impact on confidentiality and integrity of entire iSHARE Scheme.

*Data includes the data used for identification, authentication and authorisation purposes in the context of data exchange, but NOT the contents of the actual data exchange.

- **Warnings** are cautionary advices about non-compliance, about what is needed to rectify non-compliance, and by when;
- **Suspension** involves temporary deactivation of adhering/certified credentials within the iSHARE network;
- **Exclusion** involves permanent deactivation of adhering/certified credentials within the iSHARE network of the excluded party, and involves an iSHARE network wide notification of exclusion for information purposes.

Before the Scheme Owner issues warnings, suspends or even excludes parties, it MUST take into consideration/ weigh the interests of the iSHARE Scheme and -network (i.e. all Adhering/Certified Parties).

Goal

The goal of the warnings, suspension and exclusion process is to warrant trust in the iSHARE's brand, as well as protecting the confidentiality and/or integrity of (data within) the iSHARE network.

Responsibilities

Several parties have responsibilities and tasks in the warnings, suspension and exclusion process:

- The **Scheme Owner** is responsible for facilitation of the process, to protect the confidentiality and/or integrity of (data within) the iSHARE Scheme. More than in other processes he can also take an active role;
- The **Adhering/Certified Party** is responsible for acting, at all times but especially after receiving a warning or suspension, in line with scheme rules and guidelines.

Sequence

1. The reporting party (i.e. any Adhering/Certified Party or the Scheme Owner itself) reports non-compliance to the Scheme Owner, including an estimation of the non-compliance classification;
2. The Scheme Owner assesses the non-compliance and the estimated non-compliance classification by the reporting party, and:
 - a. Accepts the non-compliance classification and moves to step 3;
or
 - b. Changes the non-compliance classification and moves to step 3;
or
 - c. Rejects the reported behaviour as non-compliance, and communicates why to the reporting party.
3. If non-compliance leads to a minor incident, calamity or crisis, the [incident management process](#) is initiated. Otherwise, step 2 is followed by step 4;
4. The Scheme Owner registers the non-compliance and:
 - a. If classified as minor non-compliance, notifies the non-complying party of its non-compliance, the reason(s), and the rectifications/adjustments needed within what timespan;
 - b. If classified as major non-compliance, issues the non-complying party an official warning, and communicates its reason(s) and the rectifications/adjustments needed within what timespan;
 - c. If classified as critical non-compliance, suspends the non-complying party, by updating the party's status in the scheme registry to 'suspended', until necessary rectifications/adjustments are in place. The Scheme Owner communicates this suspension to the iSHARE network.
5. The non-complying party either:
 - a. Rectifies or adjusts within the indicated time span, and informs the Scheme Owner of the rectifications/adjustment;
or
 - b. Communicates its disagreement with the notification/warning to the Scheme Owner within 5 working days, to which the Scheme Owner MUST reply within 5 working days. The non-complying party is given another 5 working days to respond to the Scheme Owner's latest reply (which can include adjustments to its earlier notification/warning);
or
 - c. Does not take any action.
6. If sufficient rectifications/adjustments follow in time, step 8 follows. Otherwise, the Scheme Owner:
 - a. If classified as minor non-compliance:
 - i. Issues the non-complying party a warning, and communicates its reason(s) and the rectifications/adjustments needed within what timespan.
 - b. If classified as major non-compliance:

- i. Issues the non-complying party a last warning before suspension, and communicates its reason(s) and the rectifications/adjustments needed before within what timespan in order not to be suspended.
 - c. If classified as critical non-compliance:
 - i. Issues the non-complying party a last warning before exclusion, and communicates its reason(s) and the rectifications/adjustments needed before within what timespan in order not to be excluded.
7. If the non-complying party continues to dishonour the (final) warning after a reasonable time, the Scheme Owner:
 - a. If classified as minor non-compliance:
 - i. Upscales the non-compliance level to major and goes back to step 6b.
 - b. If classified as major non-compliance:
 - i. Upscales the non-compliance level to critical and goes back to step 4c.
 - c. If classified as critical non-compliance:
 - i. The Scheme Owner terminates the participation of the non-compliant party by cancellation of the Accession Agreement;
 - ii. Excludes the non-complying party from iSHARE, by updating the party's status in the scheme registry to 'ended', and initiates its withdrawal in line (as much as is reasonable) with the [withdrawal process](#);
 - iii. The Scheme Owner communicates this exclusion to the iSHARE network. The excluded party will not be allowed to take part in the [admission process](#) for the next 12 months. Step 7c is followed by step 9.
8. The Scheme Owner considers (new) actions taken by the party adequate, considers the notification or warning honoured and closes the process;
9. The Scheme Owner evaluates the incident with the reporting and/or (an) other party(s), and registers the evaluation for future learning.

Incident Management

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

The incident management process describes the steps that the Scheme Owner and Adhering- and Certified Parties MUST take to solve incidents in the iSHARE network.

An **incident** is an event, not part of the standard service operation, that results in a potential impact or risk with regards to the quality, availability, confidentiality and/or integrity of (data within) the iSHARE Scheme. This includes the data used for identification, authentication and authorisation purposes in the context of data exchange, but not the contents of the actual data exchange.

Note incident resolution is NOT part of regular maintenance, and therefore is NOT subject to maintenance windows as described under [service levels](#).

Three classifications of incidents are recognised within iSHARE. Note that the impact or risk described is non-exhaustive.

Classification	Impact or risk
Minor incident	<ul style="list-style-type: none"> • Expected unavailability of < 8 hours of an Adhering Party or < 4 hours of a Certified Party or < 2 hours of the Scheme Owner, and/or; • (Potential) data security breach, for example through the loss of a USB stick, laptop, hard disk, or because of hacking attempts or found malware, and/or; • Fraud or presumption of fraud by, for example an employee or a hacker.
Calamity	<ul style="list-style-type: none"> • Direct involvement of three or more Adhering/Certified Parties, and/or; • Serious impediment(s) to other Adhering/Certified Party(s), and/or; • Expected unavailability of > 8 hours of an Adhering Party or > 4 hours of a Certified Party or > 2 hours of the Scheme Owner, and/or; • Data security breach that needs to be reported in line with meldplicht datalekken, and/or; • (Other) impact on confidentiality and integrity.
Crisis	<ul style="list-style-type: none"> • Involvement of 10 or more Adhering/Certified Parties, and/or; • Serious impact on image and trustworthiness of iSHARE, and/or; • Expected unavailability of > 48 hours of a Certified Party or > 12 hours of the Scheme Owner, and/or; • Political implications, and/or; • Fundamental legal or technical vulnerability.

Goal

The goal of the incident management process is to handle and solve different levels of incidents in a structured way and with minimal disruption to the functioning of the iSHARE network.

Responsibilities

Several parties have responsibilities and tasks in the incident management process:

- The **Scheme Owner** proactively coordinates the handling and solving of incidents, and assists if necessary;
- **Adhering/Certified Parties** are responsible for reporting all incidents in the iSHARE network, and taking the steps necessary to handle and solve incidents.

Sequence

Before initiating the process as below, the reporting party, in conjunction with the causing party (if not the same) MUST assess together whether the event deemed an incident is indeed an incident.

1. The reporting party (i.e. any Adhering/Certified Party or the Scheme Owner itself) reports an incident to the Scheme Owner, including an estimation of the incident classification;
2. The Scheme Owner assesses the incident and the estimated incident classification by the reporting party, and:
 - a. Accepts the incident classification and moves to step 3;
or
 - b. Changes the incident classification and moves to step 3;
or
 - c. Rejects the reported event as an incident, and communicates why to the reporting party.
3. The Scheme Owner registers the incident and initiates incident handling, as follows:
 - a. If classified as a **minor incident**:
 - i. If the minor incident is assessed the result of non-compliance with scheme rules and guidelines, and/or if it has had significant negative impact on the normal operation of the iSHARE network, the [warnings, suspension and exclusion process](#) will also be initiated;
 - ii. The Scheme Owner gives the reporting party, the causing party and/or (an)other party(s) - whichever it deems most capable/suitable - the responsibility of handling the minor incident, under supervision of the Scheme Owner (see step 4);
 - iii. The party(s) responsible for handling the minor incident communicates the minor incident, the incident manager, and that the minor incident is being solved, to the parties impacted by it.
 - b. If classified as a **calamity**:
 - i. If the calamity is assessed the result of non-compliance with scheme rules and guidelines, and/or if it has had significant negative impact on the normal operation of the iSHARE network, the [warnings, suspension and exclusion process](#) will also be initiated;
 - ii. The Scheme Owner gives the reporting party, the causing party and/or (an)other party(s) - whichever it deems most capable/suitable - the responsibility of handling the calamity, under supervision of the Scheme Owner (see step 4);
 - IF there is a data security breach that needs to be reported in line with [meldplicht datalekken](#), the party(s) responsible for handling the calamity report the data security breach to the *Autoriteit Persoonsgegevens* (personal data authority) and follow the authority's guidelines on the rest of the incident management process;
 - iii. The Scheme Owner informs the iSHARE network of the calamity (and that it is being solved) and who the incident manager is, as well as any parties outside the network that it deems necessary to inform (e.g. branch organisations, the NCSC or even law enforcement);
 - iv. The Scheme Owner sets up an action plan to minimise risks and damage.
 - c. If classified as a **crisis**:
 - i. If the crisis is assessed the result of non-compliance with scheme rules and guidelines, and/or if it has had significant negative impact on the normal operation of the iSHARE network, the [warnings, suspension and exclusion process](#) will also be initiated;

- ii. The Scheme Owner gives the reporting party, the causing party and/or (an)other party(s) - whichever it deems most capable/suitable - the responsibility of handling the crisis, under supervision of and assisted by the Scheme Owner (see step 4). Different to the process for minor incidents and calamities, the Scheme Owner can also choose to take the responsibility of handling the crisis itself even if it is not the causing party;
 - IF there is a data security breach that needs to be reported in line with [meldplicht datalekken](#), the party(s) responsible for handling the calamity report the data security breach to the *Autoriteit Persoonsgegevens* (personal data authority) and follow the authority's guidelines on the rest of the incident management process;
 - iii. The Scheme Owner informs the iSHARE network of the crisis (and that it is being solved) and who the incident manager is, as well as any parties outside the network that it deems necessary to inform (e.g. branch organisations, the NCSC or even law enforcement);
 - iv. The Scheme Owner sets up an action plan to minimise risks and damage.
4. The Scheme Owner coordinates the contact with the involved parties, monitors progress and assists in handling the incident if necessary. The Scheme Owner also communicates progress to the iSHARE network in case of a calamity or crisis. If progress is non-compliant to the incident service level, the Scheme Owner MAY choose to upscale (from incident to calamity or from calamity to crisis);
5. When the incident is handled and therefore solved, the Scheme Owner closes the incident;
 - a. In case of a minor incident, the responsible party communicates the incident closure to the parties impacted by it;
 - b. In case of a calamity or crisis, the Scheme Owner communicates the incident closure to the iSHARE network.
6. The Scheme Owner evaluates the incident with the reporting and/or (an)other party(s), and registers the evaluation for future learning. It can choose to share the gained insights with (selected) parties in the iSHARE network.

Release Management

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

The iSHARE Scheme is dynamic. The release management process describes the steps that the Scheme Owner MUST take to make changes that impact the legal or technical iSHARE Scheme agreements.

These **changes** include alterations to:

- iSHARE Scheme documentation and -specifications;
- The Scheme Owner API;
- Scheme Owner tools (e.g. test- and certification tools).

Goal

The goal of the release management process is to:

- Decide in a standardised, transparent way on what changes are (not) made;
- Release changes in a standardised way, with minimal disruption to the functioning of the iSHARE network.

Responsibilities

Several parties have responsibilities and tasks in the release management process:

- The **Scheme Owner** is responsible for facilitation of a swift course of the process, and for minimising the impact of changes and releases for all participants;
- The **Change Advisory Board** has the responsibility to advise the Scheme Owner on proposed changes;
- **Adhering/ Certified Parties** can (cooperatively) prepare and submit a Request for Change (RFC) to the Scheme Owner.

Sequence

The following sequence is based on [ITIL v3](#).

1. One or several submitting parties (this can also include the Scheme Owner) submit an RFC which describes at a minimum:
 - a. A description of the desired change;
 - b. A description of the context/immediate cause;
 - c. An indication of what priority the change should have;
 - d. The potential solution (direction);
 - e. The impact for Certified and/or Adhering Parties and the Scheme Owner;
 - f. The justification of the change in a business case.
2. The RFC is logged by the Scheme Owner;
3. The Scheme Owner assesses the feasibility and impact of the submitted RFC and:
 - a. Schedules the proposed RFC for review by the Change Advisory Board. He communicates this to the submitting party(s);
 - b. Does NOT schedule the proposed RFC for review by the Change Advisory Board. He issues a written statement to the submitting party(s) explaining why;
4. The Change Advisory Board assesses the RFC and provides the Scheme Owner with advice on how to proceed;
5. On the basis of the CAB advice, a draft solution and the estimated impact, the Scheme Owner either:
 - a. Accepts the RFC and prioritises the change;
 or

- b. Rejects the RFC;
- 6. The Scheme Owner issues a written statement to the CAB and the submitting party(s), explaining the reasoning behind the acceptance/rejection of an RFC and the change's priority;
- 7. If relevant, the Scheme Owner updates the release calendar and the priority of upcoming changes;
- 8. The Scheme Owner alters the iSHARE Scheme based on the release calendar, and publishes a new version of the scheme accordingly.

Emergency changes are changes that MUST be implemented as soon as possible, for example to resolve a major or critical incident. In case of an emergency change the Scheme Owner accelerates the execution of the process. He can choose to consult (members of) the CAB on an ad-hoc basis if timing permits and deemed necessary.

If a change does NOT impact the legal or technical scheme agreements, the change MAY be made without taking the steps described here. Such changes include (but are not limited to) the restructuring of content, correcting grammatical mistakes, and maintenance to hyperlinks and labels.

Versioning guidelines

Each version of the iSHARE Scheme has a unique identifier that conveys the significance of changes between releases, whereby the first sequence is changed for the most significant changes, and changes to sequences after the first digit represent changes of decreasing significance. iSHARE uses a sequence of three digits for its versioning (x.y.z):

1. MAJOR version for significant and/ or backwards-incompatible changes (v1.0 to v2.0)
2. MINOR version for regular changes and/ or new functionality (v1.5 to v1.6)
3. PATCH version for small fixes (v1.7 to v1.7.1)

The Scheme Owner may choose to jump multiple minor versions at a time (v1.2 to v1.5) to indicate significant changes have been made, but are not enough to warrant incrementing a major version number.

Management reporting

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

The management reporting process describes the steps that parties **MUST** take to deliver management information about the use and working of the iSHARE network.

Goal

The goal of the management reporting process is to monitor compliance to service level agreements, and to distribute info about the use of the iSHARE network.

Responsibilities

Several parties have responsibilities and tasks in the management reporting process:

- The **Scheme Owner** is responsible for delivering its own management information on a monthly basis, and to process received management information into a report that does not include commercially sensitive information;
- The **Certified Party** is responsible for delivering management information timely on a monthly basis.

Sequence

1. On a monthly basis, Certified Parties and the Scheme Owner collect management information about:
 - a. the use of the iSHARE network;
 - b. compliance with the service level agreements.
2. Certified Parties and the Scheme Owner deliver the collected management information to the Scheme Owner in compliance with the standard format and service level;
3. The Scheme Owner processes the received management information on compliance, and, if non-compliance is detected, follows the [warnings, suspension and exclusion process](#) to assess whether this is an incident or structural non-compliance;
4. The Scheme Owner verifies whether each Certified Party's management information on the use of the iSHARE Scheme is correct:
 - a. If correct, step 5 follows directly.
 - b. If incorrect, a maximum of 5 working days are available for the Certified Party(s) to rectify. If 5 working days are not enough, step 5 follows without the incorrect information;
5. Quarterly, the Scheme Owner processes and anonymises (if necessary) the management information on the use of the iSHARE Scheme into a report containing:
 - a. Number of Certified Parties (also compared to last month and this month previous years);
 - b. Number of Adhering Parties;
 - c. Other information deemed necessary (to be decided);
 If incorrect information was found and could not be rectified within 5 days in step 4, a description of the missing management information.
6. The Scheme Owner distributes the management report.

Service levels

This section describes the service levels that apply to iSHARE Adhering Parties, Certified Parties and the Scheme Owner.

A **service level** measures the performance of a service. Per service level described in this section, an explanation of the service level is given before both the norm and the minimum level are defined.

The following service levels are described per party. Please click on the 'X' in each column to be redirected to the specific service level description.

	Adhering Parties	Certified Parties	Scheme Owner
Service level			
Availability	X	X	X
Performance	X	X	X
Incidents	X	X	X
Support	X	X	X
Reporting		X	X

The service levels are monitored by the Scheme Owner through:

- Analysis of certified party reports;
- Random sampling.

No norm is set for monitoring frequency or detail.

Service levels for Adhering Parties

For Adhering Parties, the following service levels apply

- [Availability](#)
- [Performance](#)
- [Incidents](#)
- [Support](#)

Availability

Availability is a measure of the time a service is in a functioning condition. It includes the availability window and the maintenance window.

Availability window

The **availability window** includes the times at which Adhering Parties guarantee the availability of their service.

No norm is set for Adhering Parties' availability window, to leave Service Providers free to run their service whenever they deem appropriate (e.g. a trucking company does not need to leave its trucks' board computers on 24 hours * all days of the year).

Minimum level required at times deemed appropriate to run service: guideline of 95% availability* per calendar month, from 00:00-23:59h

*Planned maintenance does NOT count as unavailability

Maintenance window

The **maintenance window** includes the times at which Adhering Parties can perform planned maintenance, that is likely to result in downtime, to their service. If no downtime is expected, maintenance can take place outside of the maintenance window. Planned maintenance does NOT include incident resolution, as this can take place outside the maintenance window as described under [incidents](#).

Norm:

- The maintenance window includes all times outside office hours;
- **No norm** is set for communication about (different forms of) maintenance, as this is a matter between Adhering Parties.

Performance

Performance includes the time it takes for a service to respond when requested or called upon; i.e. the time an Adhering Party's service takes to respond to a received message.

Before an Adhering Party knows whether it may respond to a request, however, it often needs to request (more) information from one or more certified parties; e.g. delegation info or authorisation info. It therefore needs to send out a new message itself, and wait for this message to be responded to by a certified party. While [Certified Parties' response times are short](#), the process of sending out and receiving (sometimes several) new messages before the original request can be answered takes time. Consequently, **no norm** is set for Adhering Parties' total performance. The following **guidelines** are set:

- 95% of Adhering Parties' messages SHOULD be responded within 2 seconds of receiving all information needed from certified parties;

- 99% of Adhering Parties' messages SHOULD be responded within 5 seconds of receiving all information needed from certified parties;
- Each Adhering Party SHOULD be able to process at least 100 simultaneous messages while meeting above requirements.

Incidents

An **incident** is an event, not part of the standard service operation, that results in a potential impact or risk with regards to the quality, availability, confidentiality and/or integrity of (data within) the iSHARE scheme. This **ONLY** includes the data used for identification, authentication and authorisation purposes in the context of data exchange, but not the contents of the actual data exchange.

Three classifications of incidents are recognised within iSHARE, as explained in the [incident management process](#):

- Minor incident;
- Calamity;
- Crisis.

Norm:

- All incidents **MUST** be communicated by the Adhering Party(s) to the Scheme Owner directly after they are discovered;
- Communication **MUST** include date, time, incident level as estimated by the Adhering Party(s), argumentation including impacted service(s), and a potential incident manager;
- In case of a calamity or crisis, the Adhering Party **MUST** have an incident manager available during working days, and **SHOULD** have an incident manager available 24 * 7;
- An update on the incident **MUST** be communicated to the Scheme Owner*:
 - For minor incidents, at the end of each working day;
 - For calamities, within 2 hours of every significant update and at the end of each working day;
 - For crises, within 2 hours of every significant update and every 4 hours.
- All incidents **SHOULD** be handled by the Adhering Party (in cooperation with the Scheme Owner as per the [incident management process](#)) within 3 working days after being appointed as the responsible party - unless agreed otherwise.

*In line with the [incident management process](#), the Scheme Owner presents an overview of current calamities and crises on its website

Support

Support by Adhering Parties could include answering questions, requests, and complaints from other Adhering Parties.

No norm is set for Adhering Parties as it is a matter between them (and other Adhering Parties). The following **guidelines** are set, however:

- Adhering Parties are available for support via e-mail;
- They **SHOULD** confirm receiving a question/request within 1 working day. They **SHOULD** send an underpinned reaction (with an answer/solution or at the very least a direction) within 5 working days.

Service levels for Certified Parties

This part of the iSHARE scheme is considered normative and is therefore compliant with RFC 2119.

For Certified Parties, the following service levels apply:

- [Availability](#)
- [Performance](#)
- [Incidents](#)
- [Support](#)
- [Reporting](#)

Availability

Availability is a measure of the time a service is in a functioning condition. It includes the availability window and the maintenance window.

Availability window

The **availability window** includes the times at which Certified Parties guarantee the availability of their service.

Norm: 24 hours * all days of the year

Minimum level required: 99% availability* per calendar month, from 00:00-23:59h

*Planned maintenance does NOT count as unavailability

Maintenance window

The **maintenance window** includes the times at which Certified Parties can perform planned maintenance, that is likely to result in downtime, to their service(s). If no downtime is expected, maintenance can take place outside of the maintenance window. Planned maintenance does NOT include incident resolution, as this can take place outside the maintenance window as described under [incidents](#).

Norm:

- The maintenance window includes the nights from Friday to Saturday and from Saturday to Sunday, from 00:00-5.59h;
- Maintenance MUST be announced to the impacted parties directly as well as to the Scheme Owner**;
- Announcements MUST be made at least 10 working days before the maintenance and MUST include date, time, and impacted service(s).

**The Scheme Owner presents an overview of its own and Certified Parties' current and planned maintenance on its website

Performance

Performance includes the time it takes for a service to respond when requested or called upon; i.e. the time a Certified Party's service takes to respond to a received message.

Norm:

- 95% of messages MUST be responded within 2 seconds;
- 99% of the messages MUST be responded within 5 seconds;
- Each Certified Party MUST be able to process at least 100 simultaneous messages while meeting above requirements.

Incidents

An **incident** is an event, not part of the standard service operation, that results in a potential impact or risk with regards to the quality, availability, confidentiality and/or integrity of (data within) the iSHARE Scheme. This **ONLY** includes the data used for identification, authentication and authorisation purposes in the context of data exchange, but not the contents of the actual data exchange.

Three classifications of incidents are recognised within iSHARE, as explained in the [incident management process](#):

- Minor incident;
- Calamity;
- Crisis.

Norm:

- All incidents **MUST** be communicated by the Certified Party(s) to the Scheme Owner directly after they are discovered;
- Communication **MUST** include date, time, incident level as estimated by the Certified Party(s), argumentation including impacted service(s), and a potential incident manager;
- In case of a calamity or crisis, the Certified Party **MUST** have an incident manager available during working days, and **SHOULD** have an incident manager available 24 * 7;
- An update on the incident **MUST** be communicated to the Scheme Owner*:
 - For minor incidents, at the end of each working day;
 - For calamities, within 2 hours of every significant update and at the end of each working day;
 - For crises, within 2 hours of every significant update and every 4 hours.
- All incidents **SHOULD** be handled by the Certified Party (in cooperation with the Scheme Owner as per the [incident management process](#)) within 3 working days after being appointed as the responsible party - unless agreed otherwise.

*In line with the [incident management process](#), the Scheme Owner presents an overview of current calamities and crises on its website

Support

Support by Certified Parties includes answering questions and requests from Adhering Parties.

Norm: Certified Parties are available for support via e-mail; they **MUST** confirm receiving a question/request within 1 working day. They **SHOULD** send an underpinned reaction (with an answer/solution or at the very least a direction) within 5 working days.

Reporting

Reports are meant to monitor both compliance to the service level agreements and the (growing) use of the iSHARE network, as described in the [management reporting process](#). The following will be reported on (non-exhaustive):

- Availability;
- Number of relations with Adhering Parties;
- Number of transactions;
- Number of transactions per Adhering Party;
- Number of incidents.

Certified Parties are expected to collect management information about each month: 0:00h on the first day to 23:59h on the last.

Norm: each Certified Party MUST deliver the management information about the last month, conform the iSHARE template, before 23:59h on the 5th working day of the current month

Service levels for Scheme Owner

This part of the iSHARE scheme is considered normative and is therefore compliant with RFC 2119.

For the Scheme Owner, the following service levels apply:

- [Availability](#)
- [Performance](#)
- [Incidents](#)
- [Support](#)
- [Reporting](#)

Availability

Availability is a measure of the time a service is in a functioning condition. It includes the availability window and the maintenance window.

Availability window

The **availability window** includes the times at which the Scheme Owner guarantees the availability of its service.

Norm: 24 hours * all days of the year

Minimum level required: 99% availability* per calendar month, from 00:00-23:59h

*Planned maintenance does NOT count as unavailability

Maintenance window

The **maintenance window** includes the times at which the Scheme Owner can perform planned maintenance, that is likely to result in downtime, to its service(s). If no downtime is expected, maintenance can take place outside of the maintenance window. Planned maintenance does NOT include incident resolution, as this can take place outside the maintenance window as described under [incidents](#).

Norm:

- The maintenance window includes the nights from Friday to Saturday and from Saturday to Sunday, from 00:00-5.59h;
- The maintenance MUST be announced**;
- Announcements MUST be made at least 10 working days before the maintenance and MUST include date, time, and impacted service(s).

**The Scheme Owner presents an overview of its own and Certified Parties' current and planned maintenance on its website

Performance

Performance includes the time it takes for a service to respond when requested or called upon; i.e. the time the Scheme Owner's service takes to respond to a received message.

Norm:

- 95% of messages MUST be responded within 2 seconds;
- 99% of the messages MUST be responded within 5 seconds;
- The Scheme Owner MUST be able to process at least 100 simultaneous messages while meeting above requirements.

Incidents

An **incident** is an event, not part of the standard service operation, that results in a potential impact or risk with regards to the quality, availability, confidentiality and/or integrity of (data within) the iSHARE scheme. This **ONLY** includes the data used for identification, authentication and authorisation purposes in the context of data exchange, but not the contents of the actual data exchange.

Three classifications of incidents are recognised within iSHARE, as explained in the [incident management process](#):

- Minor incident;
- Calamity;
- Crisis.

Norm:

Incident at the Scheme Owner:

- In case of a calamity or crisis, the Scheme Owner **MUST** have an incident manager available 24 * 7;
- An update on the incident **MUST** be communicated*:
 - For calamities, within 2 hours of every significant update and at the end of each working day;
 - For crises, within 2 hours of every significant update and every 4 hours.
- All incidents **SHOULD** be handled by the Scheme Owner within 3 working days - unless unreasonable.

Incident at another party:

- In case of any crisis, the Scheme Owner **SHOULD** be available 24 * 7 for assistance.

*In line with the [incident management process](#), the Scheme Owner presents an overview of current calamities and crises on its website

Support

Support by the Scheme Owner includes answering questions and requests from Certified Parties (other than [incidents](#) and NOT from Adhering Parties).

Norm: the Scheme Owner is available for support via e-mail; it **MUST** confirm receiving a question/request within 1 working day. It **SHOULD** send an underpinned reaction (with an answer/solution or at the very least a direction) within 5 working days.

Reporting

Reports are meant to monitor both compliance to the service level agreements and the (growing) use of the iSHARE network, as described in the [management reporting process](#).

The following will be reported on (non-exhaustive):

- Availability;
- Number of relations with Adhering Parties;
- Number of transactions;
- Number of transactions per Adhering Party;
- Number of incidents.

The Scheme Owner is expected to collect its own management information about each month - 0:00h on the first day to 23:59h on the last - *and* to collect and process Certified Parties' management information into a quarterly management report.

Norm:

- The Scheme Owner MUST have collected its own management information about the last calendar month before 23:59h on the fifth working day of the current month;
- In January, April, July and October, the Scheme Owner MUST process and anonymise its own and the received management information about the last three months into a quarterly report, and distribute this report before 16:59h on the 10th working day of the current month

Communication

This part of the iSHARE Scheme is considered normative and is therefore compliant with RFC 2119.

This section describes the agreements concerning communication about and with the iSHARE brand that is applicable for all Adhering- and Certified Parties.

It includes the guidelines for using iSHARE's name, brand, and iSHARE logo.

Usage of iSHARE name and brand

The following communication rules apply when using the iSHARE name and brand:

- All participating parties **MUST** use the visuals and logos as provided by the iSHARE Style Guide and **MUST** apply the notation and terminology as described in the Glossary. This creates clarity in the communication and brand image of iSHARE;
- The term iSHARE is used as a brand for machine-to-machine and human-to-machine iSHARE-services;
- Participating parties within the iSHARE Scheme **MAY** use the phrase 'powered by iSHARE' to support their own branding;
- If iSHARE is integrated in human-to-machine software, the iSHARE logo **SHOULD** be used in user interfaces;
- Adhering- and Certified Parties **COULD** use standard texts, key messages and other textual and visual elements as provided in the communication toolkit provided by the Scheme Owner.

Usage of iSHARE logo

The following basic principles apply to the use of the iSHARE logo:

- Please use enough white space around the logo;
- Do not alter the colouring of the logo (or use the black and white logo).

iSHARE logo material can be downloaded [here](#).

Legal

The iSHARE Scheme is underpinned by legal agreements to which all participants (both Adhering Parties and Certified Parties) need to adhere. This section contains all of the legal aspects present within iSHARE:

Legal Aspect	Description	Link
Accession Agreement for Adhering Parties	The main contract between the participant and the iSHARE Scheme Owner. This main contract refers to the terms of use, including all iSHARE specifications, to which all participants must abide. After signing this Accession Agreement, an organisation becomes a participant of the iSHARE Scheme as an Adhering Party.	PDF
Accession Agreement for Certified Parties	The main contract between the participant and the iSHARE Scheme Owner. This main contract refers to the terms of use, including all iSHARE specifications, to which all participants must abide. After signing this Accession Agreement, an organisation becomes a participant of the iSHARE Scheme as a Certified Party.	PDF
Terms of Use	<p>The Terms of Use are an appendix to and integral part of the Accession Agreement. The Terms of Use further define the rights and obligations of the various roles within the iSHARE Scheme. The Terms of Use provide a uniform set of rules for both the participants and the Scheme Owner, thereby fostering a level playing field between all parties involved.</p> <p>The Terms of Use are drafted in such a way that data can be exchanged by participants even if they have no other contractual arrangement in place. In that case, the default requirements as set forth in the Terms of Use govern their legal relationship. This includes the (license) conditions that apply to the exchange of data. But the Terms of Use leave room for participants to derogate from or further detail the provisions of the Terms of Use on a bilateral basis. However, there will be certain requirements that participants should comply with at any time, and from which they will not be able to deviate. These are the requirements that deal with the proper functioning of the iSHARE Scheme, such as each party's responsibility to safeguard the security of its IT-systems (articles 3.5 and 4.1).</p> <p>Furthermore, the Terms of Use include a number of annexes, amongst which the pre-defined conditions of exchange, the Legal Framework and the iSHARE Scheme standards and specifications.</p>	PDF
Legal Context	The Legal Framework deals with the legal context of the iSHARE Scheme. It describes the laws and regulations that are of particular importance for participants when exchanging data within the iSHARE Scheme: the eIDAS regulation, the General Data Protection Regulation, competition law and the Dutch civil code. As stipulated in the Accession Agreement and the Terms of Use, all participants are expected to comply with these and all other applicable national and international pieces of legislation.	

Legal context

This section on iSHARE's legal context clarifies which rules and regulations may apply to iSHARE participants, and provides information and formats that participants can use to improve their understanding. This section does not aim to be all-encompassing in the sense that it covers all the rules and regulations applicable to the participants, but it aims to provide useful information to iSHARE's participants. Please note that depending on an organisation's context and specific focus, different rules and regulations might apply, both stemming from national and international law, which might not be mentioned in this section.

Relevant rules, regulations and templates

- [Dutch Civil Code](#)
- [Regulation on Electronic Identification and Trust Services \(eIDAS\)](#)
- [Applicable competition law](#)
- [General Data Protection Regulation \(GDPR\)](#)

Dutch Civil Code

In setting up the iSHARE Scheme, the relevant provisions of the Dutch Civil Code need to be taken into account. This primarily relates to the Accession Agreements and the Terms of Use, which need to be drafted in accordance with Dutch contract law. With the expansion of the iSHARE Scheme, other national laws may become relevant as well. Any specific (national and international) rules for the transport and logistics sector, such as rules for agreements on the carriage of goods, fall outside the scope of this legal framework. These types of sector specific rules are not relevant for operating and using the iSHARE Scheme, although participants may need to adhere to them when contracting services through the scheme.

Regulation on Electronic Identification and Trust Services (eIDAS)

The eIDAS Regulation – formally the Regulation on electronic identification and trust services for electronic transactions in the internal market – was adopted on 23 July 2014. It aims to provide a predictable regulatory environment to enable secure and seamless electronic interactions between businesses, citizens and public authorities throughout the entire EU. It ensures that people and businesses can use their own eIDs to access public services in other EU countries and enhances cross-border interoperability of electronic trust services.

The first section of the eIDAS Regulation relates to the government-recognized eIDs and establishes a legal framework that will allow all EU countries to recognize each other's eIDs. The second section of eIDAS deals with the various electronic signatures (i.e. simple, advanced and qualified). It clarifies existing rules, but also introduces a new legal framework for electronic signatures, seals and timestamps. The new legal framework is not mandatory but introduces certain requirements that can be followed in order to grant greater legal certainty and to improve the reliability of these services.

For the purpose of the iSHARE Scheme, the governing body will determine which eID providers are to be used, which trust service providers are to be engaged and the roles these trust service providers have within the iSHARE Scheme. The selection of eID and trust service providers is also relevant for the international orientation of the iSHARE Scheme and to foster the cross-border interoperability of electronic trust services.

Applicable competition law

Agreements

Depending on whether an agreement or other behaviour has an effect in the entire EU or not, EU competition law or national competition law (and enforcement) applies. Competition law prohibits agreements that restrict competition, unless there is a justification for them.

There are different types of agreements with different rules. The rules for agreements between companies at the same level of the production chain are generally stricter than those for companies at different levels of the production chain. The iSHARE Scheme facilitates both horizontal and vertical exchanges of information.

What is problematic under competition law, is the exchange of information that is sensitive to competition, such as price lists, data on turnover, etc. Restrictive effects may, for instance, be found in cases where exchanges of information enables companies to be better aware of each other's market strategies. Agreements that have as their purpose or effect the restriction of competition (such as price fixing, market sharing) are very likely to be prohibited. On the other hand, a justification for exchanging information can be found if this leads to efficiency gains. To determine whether there are indeed efficiency gains, three conditions must be taken into account:

1. The efficiency must at least be partially passed on to the consumers which are affected by the restriction (e.g. quicker delivery of products or reduction of search costs).
2. The agreement must not restrict competition more than is necessary for the attainment of the efficiency gains (proportionality requirement).
3. The restriction of competition must not result in the total elimination of competition. As a result, competition law leaves room for such agreements.

The iSHARE Scheme could lead to efficiencies (e.g. in terms of costs or by removing barriers).

It is important to carefully draft the agreements and always assess whether they could restrict competition, and whether a restriction could be justified by – for example – efficiencies. Admittedly, it is mainly up to the participants sharing data to comply with competition law, but the iSHARE Scheme itself is not designed in a way to directly or indirectly have an adverse effect on competition. In all cases, an important principle of the iSHARE Scheme is to create a level playing field.

Dominant position

Competition law also deals with the abuse of a dominant position. Companies can also have a dominant position collectively. Whether there is a dominant position, is assessed on the basis of market shares, amongst other factors. When there is a (collective) dominant position, it is important to assess whether, for example, parties not participating in iSHARE are excluded from the market via abuse of dominance. A dominant position is not in itself anti-competitive. Only when that position is exploited to eliminate competition, it is considered an abuse. Examples of practices that can (but do not necessarily have to) lead to abuse of dominance are exclusive dealing agreements, a refusal to supply, and certain pricing practices.

The iSHARE Scheme is intended to be an open framework, accessible to any party – admitted to the iSHARE Scheme or not - seeking to use its functionalities

General Data Protection Regulation (GDPR)

On the 25th of May 2018, the Dutch privacy law (*Wet bescherming persoonsgegevens*) was overhauled by a European privacy regulation, the 'General Data Protection Regulation' (GDPR). This regulation will ensure that the same privacy rules apply throughout the entire EU and will entail substantial changes for businesses and industry.

Two of those changes are the requirements of 'privacy by design' and 'privacy by default'. Broadly speaking, this means that privacy must be taken into account throughout the entire process in which products and services are developed. This can be achieved by using techniques such as pseudonymisation and by processing as few personal data as possible, i.e. by processing only the necessary personal data. This requirement of necessity also applies to the accessibility of data (i.e. who has access to which data) and the period for which data are retained. The default settings of a product or service must also be as privacy-friendly as possible. Products and services will therefore have to be developed and designed in such a way as to ensure that they are 'privacy proof'.

Personal data must be protected adequately, via technical and organisational measures. For example: passwords, encryption, secure (SSL/TLS) network connections and pseudonymisation of data. Technical norms such as the ISO 27001 are not mandatory, but in practice they are the best way to make sure a service provider uses adequate protection. Service providers who are able to provide a statement from an independent auditor offer even more security. The most well-known statements are the ISAE 3402 and the SSAE No. 16. When you exchange data within the iSHARE Scheme and you adhere to the iSHARE technical specifications, this means that you comply with GDPR with respect to the *technical* security measures required for the exchange of personal data.

Although the majority of data shared via the iSHARE Scheme may not be personal data, there could be personal data involved. For example, data relating to employees or clients of participating parties. If personal data is shared via the iSHARE Scheme, the participating parties will need to have a legal basis to do so. A legal basis can be, for example, consent of the data subjects, or an agreement to which the data subject is a party.

When data is exchanged between two data controllers, both need a legal basis for this. A data exchange agreement then also needs to be concluded. When a data processor processes personal data on behalf of the controller, they are obliged to enter into a data processing agreement. The GDPR explains what such an agreement should contain.

Within the iSHARE Scheme, the participating parties are in control with respect to the types and amount of data they like to share and in this respect should also easily facilitate the conclusion of data processing or data sharing agreements. To facilitate participants in their GDPR compliance efforts between themselves, two contract templates can be used: depending on the role of the respective parties, they can either use the [Data Processing Agreement](#) or the [Data Exchange Agreement](#) as a basis for their contractual arrangements. Before using any of these contract templates, it should first and foremost be assessed whether the personal data can actually be lawfully processed or exchanged.

In certain cases, the GDPR requires that the privacy effects of a project are assessed in advance (a Privacy Impact Assessment). This is the case when the processing of personal data constitutes a high risk for the data subjects. For certain companies, for example, companies which monitor individuals or systematically process sensitive data, it will become mandatory to have a Privacy Officer.

For more information on how GDPR affects you, we provide a [GDPR Factsheet](#).

Glossary and legal notices

This chapter includes the iSHARE glossary and legal notices. It is presented as follows:

- [Glossary](#)
- [Legal notices](#)

Glossary

DISCLAIMER: all descriptions are definitions written by iSHARE, unless specified otherwise

- ABAC
- Accountability
- Adherence (iSHARE)
- API
- Authentication
- Authenticity
- Authorization
- Authorization Registry (role)
- Caching
- Certificate authority
- Certification (iSHARE)
- Confidentiality
- Credentials
- CRUD
- Data classification
- Data exchange
- Data Owner
- Delegation
- eIDAS
- Encryption
- Entitled Party (role)
- EORI
- HTTP(S)
- Human Service Consumer (role)
- Identification
- Identity Broker (role)
- Identity Provider (role)
- Integrity
- JSON
- JWT
- Levels of assurance
- Machine Service Consumer (role)
- Non-repudiation
- OAuth
- OIN
- OpenID Connect
- PDP
- PEP
- PIP
- PKI
- PKI Root
- RBAC
- Responsibility
- REST
- Scheme
- Scheme Owner (role)
- Service Consumer (role)
- Service Provider (role)

- [Service provision](#)
 - [Signing](#)
 - [Status Code](#)
 - [TLS](#)
 - [Token](#)
-

ABAC

ABAC (Attribute-Based Access Control) is assigning authorizations based on attributes (contextual pieces of information that are relevant to an access decision, such as device type, [RBAC](#) role, time, location, or [CRUD](#) level). The attributes can be associated with all entities that are involved with certain actions, such as the subject, the object, the action itself and the context (e.g. time, location). The attributes are compared with policies to decide which actions are allowed in which context, granting access based on the policy outcomes.

Accountability

There is a clear distinction between accountability and [Responsibility](#).

Accountability can be described as being liable or answerable for the completion of a certain task. Someone or something who is accountable oversees and manages the stakeholder(s) who are responsible for performing the work effort. In order to be effective, accountability should lie with a sole entity or role.

Responsibility may be delegated, but accountability cannot.

Adherence (iSHARE)

An **iSHARE Adhering Party** adheres to the iSHARE terms of use. An iSHARE Adhering Party MUST sign an Accession Agreement with the [Scheme Owner \(role\)](#).

API

An API (Application Programming Interface) is a technical interface, consisting of a set of protocols and data structuring standards ('API specifications') which enables computer systems to directly communicate with each other. Data or services can be directly requested from a server by adhering to the protocols. APIs are used to hide the full complexity of software and make it easy for third parties to use parts of software or data services. APIs are mainly meant for developers to make the creation of new applications depending on other applications easier.

Authentication

Authentication is the process of determining or validating whether someone or something is, in fact, who or what it is claiming to be. There are several means of authenticating the identity of an entity, which can be used alone or in combination:

- Something the entity knows – examples includes a password, PIN, passphrase, or answer to a secret question;
- Something the entity possesses – examples include electronic keycard, smartcard, token, and smartphone;
- Something the entity is (biometrics) – examples include recognition by fingerprint, retina, iris, and face;
- Something the entity does (behavioral dynamics) – examples include recognition by voice pattern, swipe characteristics, handwriting characteristics, and typing rhythm;

- Something about the context of the entity – examples include IP address, device type, geolocation, and time of day.
-

Authenticity

In the context of information security, **authenticity** refers to the truthfulness of information and if this has been sent or created by an authentic sender.

Authenticity can be achieved by digitally [Signing](#) a message with the private key from the sender. The recipient can verify the digital signature with the matching public key. Certificates containing public and private keys are issued by a [Certificate Authority](#).

Authorization

Authorization is the process of giving someone or something permission to something, for example to access to services, data or other functionalities. Authorization is enabled by [Authentication](#). Policies and attributes determine what types of activities are permitted by an entity.

Authorization Registry (role)

The **Authorization Registry**:

- Manages records of [Delegation](#) and [Authorization](#) of [Entitled Party \(role\)](#) and/or [Service Consumer \(role\)](#);
- Checks on the basis of the registered permission(s) whether a [Machine Service Consumer \(role\)](#) Service Consumer is authorized to take delivery of the requested service, and;
- Confirms the established powers towards the [Service Provider \(role\)](#).

Within the iSHARE Scheme, the term Authorization Registry always refers to an external Authorization Registry (not part of the [Service Provider \(role\)](#) or [Entitled Party \(role\)](#)).

The Authorization Registry is a role for which iSHARE [Certification \(iSHARE\)](#) is REQUIRED.

Caching

Web servers can temporarily store data in order to enable faster access to this data at a later moment, this is called 'caching'.

Certificate Authority

A **Certificate Authority (CA)** is:

- An entity that issues digital certificates;
- A trusted party, and;
- Responsible for the binding to a specific entity of the certificate (registration & issuance).

A digital certificate certifies the ownership of a public key by the named subject of the certificate, so other parties can rely upon signatures or assertions made with the private key that corresponds to the certified public key.

A **Registration Authority** verifies the identity of entities requesting digital certificates to be issued by the CA and validates the correctness of the registration.

A **Validation Authority** verifies the validity of digital certificates on behalf of the CA.

Certification (iSHARE)

Roles for which certification is required facilitate certain functions for the iSHARE Scheme that every party within iSHARE must be able to rely upon. An **iSHARE Certified Party** MUST apply to the [Scheme Owner \(role\)](#) for certification and, after providing sufficient proof, MUST sign a certification agreement with the [Scheme Owner \(role\)](#).

Confidentiality

In the context of information security, **confidentiality** refers to the protection of information from disclosure to unauthorized parties.

Confidentiality can be achieved by the use of cryptography, as well as access control; the message the recipient gets can be proven not to have been read by anyone else but the legitimate sender and recipient.

Credentials

In the context of information security, **credentials** are used to control access of someone or something to something, for example to services, data or other functionalities. The right credentials validate (i.e. [Authentication](#)) the identity claimed during [Identification](#).

The best-known example of credentials is a password, but other forms include electronic keycards, biometrics and, for machines, public key certificates.

CRUD

CRUD (acronym for Create, Read, Update, Delete) are considered to be basic functions regarding stored data. In computer programming, possible actions are often mapped to these standard CRUD functions in order to clarify the actions. For example, standard [HTTP\(S\)](#) actions GET and POST refer to Read and Create functions regarding stored data.

Data classification

The **classification of data** in categories is an important pre-requisite for proper [Authorization](#). Data can be classified in categories defining their type, location, sensitivity and protection level.

Clustering data in categories does not only simplify the authorization process (i.e. giving someone or something permission to data), it also provides a clear overview and lowers the risk of exchanging sensitive data with unauthorized entities. A risk analysis is part of the data classification process.

Data exchange

Data exchange is the process of supplying data and receiving (an)other (set of) data in return.

Data Owner

The **Data owner** is the legal person [Accountability](#) for the [Confidentiality](#), [Integrity](#), [availability](#) and accurate reporting of data.

The Data Owner can be the [Service Provider \(role\)](#). In this case, he is not only accountable for the availability of data, but also [Responsibility](#).

Delegation

Delegation is the act of empowering someone or something to act for another or to represent other(s).

In the iSHARE network, a delegated [Service Consumer \(role\)](#) acts on behalf of an [Entitled Party \(role\)](#).

eIDAS

eIDAS is an EU regulation on electronic identification and trust services for electronic transactions in the European Single Market. The regulation provides important aspects related to electronic transactions, such as qualified electronic certificates.

Encryption

Encryption is the process of converting data from plaintext to ciphertext. Plaintext (also called cleartext) represents data in its original (readable) format, whereas ciphertext (also called cryptogram) represents data in encrypted (unreadable) format.

Decryption is the process of converting data from ciphertext to plaintext.

The algorithm represents the mathematical or non-mathematical function used in the encryption and decryption process.

A cryptographic key represents the input that controls the operation of the cryptographic algorithm. With symmetric encryption the same key is used for encryption and decryption, whereas with asymmetric encryption two different, but mathematically related keys are used for either encryption or decryption, a so-called public key and a private key.

A crypto system represents the entire cryptographic environment, including hardware, software, keys, algorithms and procedures.

Entitled Party (role)

The **Entitled Party** is the legal entity that has one or more rights to something, e.g. to data at a [Service Provider \(role\)](#) that it has a legal agreement with. The Entitled Party is either the same entity as the [Service Consumer \(role\)](#), or delegates its rights to another Service Consumer. In the latter case, this other Service Consumer('s machines and humans) can consume services on the Entitled Party's behalf.

The Entitled Party is a role for which iSHARE [Adherence \(iSHARE\)](#) is REQUIRED.

EORI

An **EORI (Economic Operator Registration and Identification)** is an identification number, unique throughout the European Community, assigned by a customs authority or designated authority in a Member State to economic operators and other persons, and valid throughout the Community.

The format of the EORI number consists of a country code followed by a unique code which is established within an EU member state. For example, in the Netherlands the EORI consists of: NL, followed by an RSIN (*Rechtspersonen en Samenwerkingsverbanden Identificatie* number). If the NL-RSIN combination contains less than 9 digits, the EORI is prefixed with 0's.

In the iSHARE network, the EORI number is used to uniquely identify legal persons. Note that non-European Community legal persons doing business in/with Europe also have an EORI.

**Source**

EORI.eu

HTTP(S)

HTTP stands for 'Hypertext Transfer Protocol', and when secured via [TLS](#) or SSL it is referred to as HTTPS (HTTP Secure). It is a protocol for (secure) communication over a computer network and is widely used on the Internet.

Human Service Consumer (role)

The **Human Service Consumer** is a role that represents a human (person) who requests, receives, and uses certain services, such as data, from a [Service Provider \(role\)](#) on behalf of and authorized by the [Service Consumer \(role\)](#).

The Human Service Consumer is not a separate role, but belongs to the Adhering Party Service Consumer.

Identification

Identification is the process of someone or something claiming an identity by presenting characteristics called identity attributes. Such attributes include a name, user name, e-mail address, etc. The claimed identity can be validated (i.e. [Authentication](#)) with the right [credentials](#).

Identity Broker (role)

If multiple distinct [Service Provider \(role\)](#) exist where each data set is protected under a distinct trust domain, multiple [Identity Provider \(role\)](#) may be needed. Moreover, the iSHARE Scheme may require different [Levels of assurance](#) for specific data and may wish to designate specific Identity Providers for specific services.

In order to support multiple Identity Providers (with possible multiple rules) and Service Providers, an **Identity Broker** is required. An Identity Broker allows [Human Service Consumer \(role\)](#) to select the Identity Provider they prefer to [Authentication](#) themselves at. It prevents the need for a direct relationship between all Service Providers and all Identity Providers.

The Identity Broker is a role for which iSHARE [Certification \(iSHARE\)](#) is REQUIRED.

Identity Provider (role)

The **Identity Provider**:

- Provides identifiers for [Human Service Consumer](#);
- Issues credentials to Human Service Consumers;
- Manages records of [Authorization](#) of the [Service Consumer \(role\)](#);
- Identifies and authenticates [Human Service Consumers](#) based on provided credentials
- Checks on the basis of the provided credentials and the registered permission(s) whether a [Human Service Consumer \(role\)](#) Service Consumer is authorized to take delivery of the requested service, and;
- Confirms the established powers towards the [Service Provider \(role\)](#).
- Possibly provides other information (which are frequently referred to as attributes) about the user that is known to the Identity Provider.

In the iSHARE environment an Identity Provider could support various methods of [Authentication](#), such as:

- Password authentication;
- Hardware-based authentication (e.g. smartcard, token);
- Biometric authentication;
- Attribute-based authentication.

Depending on parameters such as the quality of the registration process, quality of credentials, use of biometrics or multiple authentication factors and information security, an Identity Provider can provide a client with a high or low confidence in the claimed identity of the user which is known to the Identity Provider. This is also known as the [Levels of assurance](#).

The Identity Provider is a role for which iSHARE [Certification \(iSHARE\)](#) is REQUIRED.

Integrity

In the context of information security, **integrity** refers to the protection of information from being modified by unauthorized parties.

Integrity can be achieved by a.o. hash functions (hashing the received data and comparing it with the hash of the original message); the message the recipient receives from the sender can be proven not to have been changed during the transmission.

JSON

JSON is short for 'JavaScript Object Notation' and is an open standard data format that does not depend on a specific programming language. This compact data format makes use of human-readable (easy to read) text to exchange data objects (structured data) between applications and for data storage.

JSON is most commonly used for asynchronous communication between browsers and servers.

JWT

A JSON Web Token (JWT) is used when [Non-repudiation](#) between parties is required. A statement, of which the data is encoded in [JSON](#), is digitally [Signing](#) to protect the [Authenticity](#) and [Integrity](#) of the statement.

Levels of Assurance (LoA)

Within online [Authentication](#), depending on the authentication protocol used, the server is to some extent assured of the client's identity. Depending on parameters such as the quality of the registration process, quality of credentials, use of biometrics or multiple authentication factors and information security, an authentication protocol can provide a server with a high or low confidence in the claimed identity of the client. For low-interest products, a low certainty might be sufficient, while for sensitive data it is essential that a server is confident that the client's claimed identity is valid.

Machine Service Consumer (role)

The **Machine Service Consumer** is a role that represents a machine that requests, receives, and uses certain services, such as data, from a [Service Provider \(role\)](#) on behalf of and authorized by the [Service Consumer \(role\)](#).

The Machine Service Consumer is not a separate role, but it belongs to the Adhering Party [Service Consumer \(role\)](#).

Non-repudiation

In the context of information security, **non-repudiation** (Dutch 'onweerlegbaarheid') refers to the fact that the sending (or broadcast) and receipt of the message cannot be denied by either of the involved parties (sender and recipient).

Non-repudiation is closely related to [Authenticity](#) and can be achieved by digital [Signing](#) in combination with message tracking.

OAuth

OAuth is an open standard for [Authorization](#) which is used by i.e. Google, Facebook, Microsoft, Twitter etc. to let their users exchange information about their accounts with other applications or websites. OAuth is designed to work with [HTTP\(S\)](#). Within iSHARE, a modified version of OAuth 2.0 is used.

Through OAuth users can authorize third party applications or websites to access their account information on other 'master' systems without the need of exchanging with them their [Credentials](#) to login onto the platform. OAuth provides a 'secure delegated access' to resources (email accounts, pictures accounts, etc.) on behalf of the resource owner.

It specifies a method for resource owners to authorize third parties access to their resources without exchanging their credentials (username, password). Authorization servers (of the platform) issue access tokens to third party clients (applications or websites) with the approval of the resource owner (= end user). The third party client needs the access token to get access to the resources that are stored on the resource server (of the master system).

OIN

The OIN format is used to uniquely identify organisations. OIN stands for Organization Identifying Number. An OIN consists of the following concatenated elements:

- An 8-digit prefix that tells the register where the number is defined (e.g. Chamber of Commerce, RSIN etc.)
 - A number whose value depends on the register
-

OpenID Connect

OpenID Connect (OIDC) is the authentication layer that is built on top of [OAuth 2.0](#) protocol which is an authorization framework. The OIDC authentication layer allows clients to verify the ID and obtain basic profile information of their end-users

The authentication is performed by the authorization server (managing the access rights and conditions) in an interoperable and [REST](#)-like manner. Within iSHARE, OpenID Connect 1.0 is used.

PDP

Policy Decision Point. Entity that evaluates access requests that are received from the policy enforcement point ([PEP](#)). Subsequently an answer is sent back to the PEP.

PEP

Policy Enforcement Point. Entity that determines whether an action is permitted or not. It takes any access requests and forwards these to the policy decision point ([PDP](#)).

PIP

Policy Information Point. Entity that holds policy information and is contacted as a source of information regarding [Delegation/Authorization](#) information.

PKI (Public Key Infrastructure)

A PKI is a system for distribution and management of digital keys and certificates, which enables secure authentication of parties interacting with each other.

Generally, three different methods exist for creating trust within PKI's. These are through 'Certificate Authorities', 'Web of Trust' and 'Simple PKI'. Within iSHARE the 'Certificate Authority' approach is used, and as such the other methods will not be discussed.

A PKI can be considered as a chain of certificates. At the beginning of the chain is the root '[Certificate Authority](#)' (CA), a public trusted party which is allowed to digitally [Signing](#) their own certificates (SSC, self-signed certificate). This '[PKI Root CA](#)' distributes certificates and encryption keys to organisations. The certificate is signed by the 'root CA' as proof that the owner of the certificate is trusted. These organisations can start distributing certificates as well, if allowed by their root. They become CA's, and as such sign the certificates that they distribute. Repeating these steps, a chain of certificates is created, with each certificate signed by the CA who distributed the certificate.

Parties need to trust a certificate for [Authentication](#) purposes. Instead of trusting individual certificates of organisations, root certificates can be trusted. By trusting a root, all certificates that have the root within their PKI chains are automatically trusted. Most large root CA's are automatically trusted within web browsers, enabling computers to safely interact with most web servers.

PKI Root

A PKI root is another term for root certificate, and stands for an unsigned or self-signed public key certificate that identifies the Certificate Authority, the party who is trusted by all members in the trust framework. The most common type of PKI certificates are based on the X.509 standard and normally include the digital signature of the Certificate Authority. The certificate authority issues digital certificates to all members in the trust framework.

RBAC

Role-Based Access Control. Assigning authorizations through business roles. An RBAC role represents a set of tasks or activities translated into authorizations, reflecting one or more of the following:

- Organisational structure
- Business processes
- Policies (rules)

RBAC authorizations can either give access to the front door of the information system or can be translated to access rights within the information system (often through application roles or groups).

Responsibility

There is a clear distinction between responsibility and [Accountability](#).

Responsibility can be described as tasked with getting the job done. Someone or something who is responsible performs the actual work effort to meet a stated objective.

Responsibility may be delegated, but accountability cannot.

REST(ful)

REST stands for 'Representational State Transfer' and is an architectural style for building systems and services, systems adhering to this architectural style are commonly referred to as 'RESTful systems'. REST itself is not a formal standard, but it is an architecture that applies various common technical standards such as [HTTP\(S\)](#), [JSON](#) and [URI](#).

A RESTful [API](#) indicates that the API architecture follows REST 'constraints'. Constraints restrict the way that servers respond and process client requests, in order to preserve the design goals which are intended by applying REST. Goals of REST are, among others, performance and scalability. Both are of utmost importance in iSHARE.

Scheme

A **scheme** can be defined as a collaborative effort to establish and maintain a set of agreements, to achieve a common goal.

iSHARE is a scheme with [common goals](#). Other schemes include credit card schemes such as MasterCard and Visa, payment scheme iDEAL and identity scheme eHerkenning.

Scheme Owner (role)

The **Scheme Owner** represents the body that governs the iSHARE Scheme and its participants.

As part of the [secondary use cases](#), parties will need to register themselves as certified or adhering at the Scheme Owner. They will also need to consult the Scheme Owner to check whether their counterparty is adherent or certified.

Service Consumer (role)

The **Service Consumer** is the legal entity that consumes the [Service Provider \(role\)](#)'s service on the basis of the [Entitled Party \(role\)](#)'s rights to that service. It can do so because the Service Consumer is either the same legal entity as the Entitled Party (i.e. it already has these rights), or because the Entitled Party has delegated rights to the Service Consumer

The Service Consumer interacts with the Service Provider; in the form of a [Machine Service Consumer \(role\)](#) or [Human Service Consumer \(role\)](#).

The Service Consumer is a role for which iSHARE [Adherence \(iSHARE\)](#) is REQUIRED.

Service Provider (role)

The **Service Provider** is a role that provides certain services, such as data, to a [Service Consumer \(role\)](#). In case the service pertains to data provisioning, the Service Provider is either the [Data Owner](#), or has explicit consent of the Data Owner to provide the services.

The Service Provider is [Responsibility](#) for the availability of services, and [Accountability](#) for these services if it is also the Data Owner.

The Service Provider is a role for which iSHARE [Adherence \(iSHARE\)](#) is REQUIRED.

Service provision

Service provision is the act of providing or supplying something for consumption or use. One of the most common forms of service provision is the [Data exchange](#).

Signing

Signing is the process of [Encryption](#) data (message, document, transaction) with the private key of the sender. It enables a receiver to confirm the [Authenticity](#) of the data. Signing also provides for [Non-repudiation](#), so that it is ensured that a sender cannot deny having sent a message.

In most cases, a hash of the data is encrypted. Thus, both the [Integrity](#) and the [Authenticity](#) of the data can be verified. Confirmation takes place by the receiver using the public key of the sender. The public key is contained in the digital certificate that is sent by the sender along with the signed data. The association of the key pair with the sender MUST be assured by a [Certificate Authority](#).

Status Code / Response Code

After sending a [HTTP\(S\)](#) request to a server, the server responds with (among others) a Status Code which indicates the outcome of the request made to the server. A well known response is 404 Not found, indicating that the requested location or resource is not (yet) found.

TLS

TLS (Transport Layer Security) is a set of protocols that provides for secure communication in computer networks. TLS makes use of cryptography and is widely used by a variety of applications such as web browsing, email and voice-over-IP. Securing [HTTP\(S\)](#) communication via (among others) TLS results in the [HTTP\(S\)](#) protocol. Securing communication with TLS v1.2 is mandatory for all iSHARE communication.

Token

Something that serves as a verifiable representation of some fact, e.g. an identity or entitlement.

Within iSHARE, Tokens are issued after successfully completing [API](#) requests which are then used to process the next request. For example, to access a certain service, first an access token is required. Upon receiving this access token, it can be used to request the service itself.

Legal notices

No part of these specifications may be reproduced in any form by print, photo print, microfilm or any other means or stored in an electronic retrieval system, without the prior written consent of the iSHARE project organisation, which must never be presumed.

Project history

This chapter describes the iSHARE project history.

- [Background information](#)
- [Assumptions](#)

Background information

The project to establish the iSHARE Scheme was initiated by the Neutral Logistics Information Platform (NLIP), as part of the government programme 'Topsector Logistiek', through a tender project in 2016. NLIP requested market parties to present plans to lower barriers for more efficient data exchange in the Dutch logistics sector. The combination of the parties **INNOPAY** and **Maxcode** won the tender, with its plan to set-up a scheme of multilateral agreements instead of, for instance, a technology-centric approach relying on a software platform.

The establishment of the iSHARE Scheme was set out in 4 phases:

- **Phase 1** (Jun 2016 - Jan 2017): Preparatory phase
Organisations were openly invited to participate in the iSHARE initiative which resulted in the so called 'startdocument v0.1'. Startdocument v0.1 provided the preliminary scope for the iSHARE Scheme based on identified challenges and use cases of involved organisations;
- **Phase 2** (Part I: Jan 2017 - Jun 2017; part II: Jul 2017 - Dec 2017): Co-creation phase
Participating organisations worked collaboratively in Functional, Technical, Operational and Legal working groups towards the first full version of the iSHARE Scheme v1.0 (and later versions 1.2 and 1.5) a first full set of agreements for improved data exchanging conditions. Participating organisations also realised Proofs of Concept to verify the correct functional and technical working of the iSHARE Scheme;
- **Phase 3** (Jan 2018 - Jun 2018): Soft launch phase
The involved organisations organise how the iSHARE Scheme's integrity and sustainability are kept in check. This involves establishing/designating an organisation entrusted with the responsibility to safeguard the integrity of the iSHARE Scheme;
- **Phase 4** (Jul 2018 and onwards): iSHARE live
iSHARE opens up to any party interested and willing to abide by the agreements as set out by involved organisations.

iSHARE's co-creation partners come from a variety of backgrounds: private and public organisations, organisations of different sizes, (serving) different modalities, both providers and receivers of data, etc. This variety of organisations ensures that the iSHARE Scheme is widely applicable.

Assumptions

The iSHARE Scheme was developed with the following assumptions in mind:

1. **Conditions for the exchange of data are assumed to be established;**
The iSHARE Scheme needs to rely upon the responsibility of participants to know what rights they have to what data. iSHARE is meant as an instrument to exchange data in a uniform, controlled and straightforward way; it is not meant as a means to resolve questions of data ownership. In practice this means that a party sharing data bears responsibility to sufficiently establish whether the party receiving the data is authorized to receive it.
2. **Data formats and semantics are assumed to be in place;**
In order to be able to exchange data, a mutual understanding of the meaning of data and the way data is structured is required. Within iSHARE, it is assumed that this mutual understanding exists and thus the exchange of data between involved parties is possible (in line with [guiding principle 4](#)). Please note that this assumption emphasises the need for industry initiatives on data standards and formats.
3. **Data classification has taken place;**
It is assumed that within the iSHARE Scheme, participants have sufficiently identified and classified their data. iSHARE participants are responsible for the classification of their data; the iSHARE Scheme does not prescribe its participants how to classify their resources. Please refer to [data classification in the glossary](#) for further detail.

Operational assumptions:

The Operational details of the iSHARE Scheme were developed with the following assumptions in mind:

1. **There will be a Scheme Owner of a yet to be defined form;**
This can be an existing body or a new body, and/or responsibilities can be split between different bodies.
2. **The Scheme Owner is financed through some type of financing constellation;**
This can be through participants paying some type of fee or in any other feasible way. The Operational working group did not decide upon the financing constellation of the Scheme Owner.
3. **The complexity of the operational processes is expected to be as follows:**
 - It is considered reasonable to expect between 1000 and 10000 Adhering Parties in the first 5 years after iSHARE goes live;
 - It is considered reasonable to expect between 20 and 50 Certified Parties in the first 5 years after iSHARE goes live;
 - It is considered reasonable to expect parties to participate from countries all over the world in the first 5 years;
 - The Scheme Owner aims to keep effort needed for admission as low as possible for both Adhering- and Certified Parties without compromising the integrity of the iSHARE Scheme and -network;
 - The Scheme Owner regularly tests the robustness of the scheme and plans for mitigation of risks/ threats (e.g. identifying Single Points of Failure);
 - The Scheme Owner is assumed to have at least some responsibility in realising sustainable growth of the iSHARE network;
 - The management of disputes regarding the contents of the data shared through iSHARE is not a core role of the Scheme Owner; disputes should be handled by involved parties.

These assumptions are, in NO WAY, ambitions. They were simply defined to base processes and service levels upon.