Preparatory work in view of the procurement of an open source cloud-to-edge middleware platform

Architecture Vision *Document*

-

V4.00

30 March 2022

Table of contents

Table of figures

Table of tables

1 Introduction

With the ongoing exponential growth of data, there is a pressing need within the European Union to provide access to resilient and competitive data storage and processing capacities for both the private and the public sector. In particular, the European Commission aims to address the need for more data sharing and decentralised data processing closer to the user (at the edge). It is also critical to deploy EU data services in the public and the private sector to grant Europe a leading status as a data-driven society and improve the data usage within the European Union. The data services of various organisations within the same industry sector should be abstracted into sector-specific data spaces. This could bring several benefits, such as greater productivity, improvements in health and well-being, environment and climate change adaptation, transparent governance and convenient public services.

To support the above-mentioned objectives, the European Commission will create an open source, multi-vendor, large-scale, modular and interoperable smart middleware platform. The smart middleware will be the basis for a European Cloud Federation enabling the operation and interconnection within and in between various European data spaces and the safe migration of the users to the cloud. The current project supports DG CONNECT in the preparatory phase for the procurement of such middleware. As such, the objectives are the following:

- Objective 01: Develop a high-level implementation roadmap to realise the vision of DG CONNECT regarding the smart middleware;
- Objective 02: Develop the requirements, including a high-level architecture vision that will serve as a basis for the elaboration of a Minimum Viable Product of the open-source smart middleware.

The Smart Middleware Platform will federate data, software, and infrastructure across the European Union with secure, resilient, energy efficient, and accessible cloud-to-edge capabilities. It will allow EU stakeholders to pool together resources to create more business value, increase resource usage efficiency, and reduce costs and duplication of efforts. The Smart Middleware Platform considers both the public sector as well as EU business as core stakeholders. Using the features provided by the Smart Middleware Platform, an open marketplace for EU resources will be created that enables energy-efficient reuse of efforts achieved by other EU participants. Chapter 3 further examines the vision of the Smart Middleware Platform and how this vision comes into fruition through common EU data spaces.

Furthermore, this document details the conceptual architecture of the Smart Middleware Platform. Chapter 4 describes the design approach of the conceived architecture, the principles on which the architecture is based, and the key business requirements that drive the architecture. Chapter 5 continues with a breakdown of the conceptual architecture, the identified capabilities, and building blocks that support the use cases of the SMP. Chapter 6 ends with a mapping of the building blocks of the platform to system components in a data space that uses the SMP.

This document describes the latest version of the conceptual design. The main objective is to present the end results and to give a comprehensive overview of the Smart Middleware vision, including its logical architecture. This document will be completed in next steps of the project, identifying information flows, detailed architectural diagrams, and technological components.

2 Terminology and definitions

Definitions	
SMP	Smart Middleware Platform
Data space	A data space is considered an open ecosystem of distributed, federated actors that share data, applications, and infrastructure amongst each other.
Edge computing	The exact definition of edge computing can vary greatly in different contexts and perspectives. Edge can be considered as all infrastructure that is not in hyperscaler cloud providers. Another definition is provided by Gartner explaining this term as "*a part of a distributed computing topology where information processing is located close to the edge, where things and people produce or consume that information*".[1] In this study, edge infrastructure (regardless of the infrastructure typology) will need to be capable of supporting internal orchestration of high-level services (containers, virtual machines, …) on top of the operating system. Direct integration with operating systems or infrastructure is considered out of scope for the middleware.
Capabilities	Capabilities describe the functionalities that the SMP should be capable of.
Building blocks	Building blocks are smaller units, they are the components serving a certain functionality.
Epic use case	Epic use cases are EOSC, Destination Earth, Data Spaces and AI on demand. They can be considered as the main use cases of the SMP as they provide the requirements for it and serve as a basis for the SMP design.
Sub-use case	Sub-use cases are the detailed use cases for the epic use cases. They describe the scenarios in which the SMP may be used by an epic use case.
HPC	High Performance Computing
IAA	Identification, Authentication, Authorization
RBAC	Role-Based Access Control
CSIRT	Computer Security Incident Response Team

[1] Source: Definition of Edge Computing - IT Glossary | Gartner

3 Smart middleware vision

In the long term, the Smart Middleware Platform should federate data, software, and infrastructure providers and consumers in the European Union. The Smart Middleware Platform allows EU stakeholders to pool together resources to create more business value, efficient resource usage, reduce costs and the duplication of efforts made towards a common goal. The middleware will facilitate the connection between isolated data centres and EU actors who can capitalise on underutilised infrastructure; will open up data assets to public institutions, SMEs, organisations and industry to improve services in the general public interest; and can be leveraged in the commission's ambition to create an open marketplace for EU resources which leads to efficient reuse of efforts made by other EU parties. In line with the European Green Deal, the Smart Middleware Platform will be based on low-power, energy efficient software. Its services also aim to optimize energy usage across industries.[2]

This long term vision will be implemented through a gradual step-by-step approach. This study takes one of the first steps to achieve the long term vision, by federating assets related to specific sectors and industries. The study forms the preparatory work to procure a minimal viable product (MVP) of the Smart Middleware Platform as a start. Additionally, a longer term roadmap for the middleware will be developed that defines the steps to achieve the broader vision.

The first objective of this project should be to allow participants with common interests, standards, or business needs to share valuable assets amongst each other. This objective translates to the middleware enabling the construction of sector-specific common EU data spaces. Additionally, the Smart Middleware Platform shall also open the door towards resource sharing across sectors or industries. Interoperability stands at the core of the SMP, which enables the exchange of data, cloud resources, AI tools and more within and across data spaces.

Note that this study takes a broad definition of the term "data space". It considers a data space as an open ecosystem of actors with mutually aligned interests that share valuable assets. These assets can be data, but also applications, algorithms or provided infrastructure. As such, the term "data space" is used to describe an ecosystem where more than just data is shared amongst participants.

3.1 Architecture stakeholders

Many stakeholders benefit from the increased transparency, interoperability and openness of assets within various sectors. Some examples can be:

- Data processing by the public sector should translate to a societal benefit for citizens. More transparency of data gathered by public institutions to other administrations, citizens, or private companies, can serve this common good. The other way around, data sharing towards public institutions can support decision making to create a data-driven government[3]. The SMP enables public institutions to integrate with other public institutions as well as private enterprises.
- Data, infrastructure, and application sharing aids the European mission described in the European Green Deal. Through extensive data gathering and data processing, environmental impact of legislation can be monitored and predicted. These insights can guide decision making throughout the EU[4]. For this purpose, a European Green Deal Data Space will be developed, which will be facilitated by the SMP.
- The openness of healthcare data can enhance research, accelerate healthcare innovations, and improve patient outcomes. This requires extensive collaboration between healthcare providers, research institutions, and private healthcare innovation centres. In this setting, special attention should be given to patient privacy and GDPR compliance. The data exchange cannot disclose personal identifiable

[2] European Commission, https://digital-strategy.ec.europa.eu/en/activities/work-programmes-digital, "The digital Europe work program", 10 November 2021
[3] Gaia-X Association, https://www.gaia-x.eu/sites/default/files/2021-08/Gaia-X_DSBC_PositionPaper.pdf#page=141, "*Data Space Business Committee – Position Papers",* 13 August 2021
[4] European Commission, https://ec.europa.eu/info/strategy/priorities-2019-2024/europe-fit-digital-age/european-data-strategy_en, "*A European strategy for data*", 19 February 2020

information of patient records unless there is explicit consent from the person from who the personal data originates[5].

The stakeholders of the SMP were previously divided into four actor groups[6]. These actor groups represent the 'roles' that stakeholders can assume when interacting with the Smart Middleware, depending on what they can offer to or use from the SMP ecosystem. A stakeholder does not need to limit itself to a single actor group. If the need arises, a stakeholder can switch its role or even assume multiple roles at once. The four actor groups are briefly summarised as follows:

- *Data providers* hold valuable data assets that they share to the SMP ecosystem;
- *Infrastructure providers* allow users to run applications on their infrastructure;
- *Application providers* share their applications and algorithms towards interested end users; and
- *End users* consume assets and resources.

This summary heavily simplifies the interactions and relations between these actors. Complex processes, such as usage agreements or usage billing, accompany these interactions. A more in-depth overview of the different actors is provided in the Business Use Cases and Business Processes document.

It is important to note that data sharing should comply with the well-established regulation. Annex II gives a brief overview on how the Data Governance Act applies to the SMP actors described above.

3.2 SMP architecture objectives and scope

Figure 1 displays how the architecture vision of the SMP maps the four actor groups and different data spaces. At the core of data spaces lie the four types of actors the SMP considers. These actors are a symbolic representation for a distributed network of cooperating parties in an open ecosystem. The Smart Middleware Platform, represented by the Smart Middleware Platform (SMP) Agent, spans across these actors[7], enabling asset sharing between them. It provides common services on which data spaces can be built. The middleware stays agnostic to the specifics of a particular data space, allowing additional data space specific services to be added on top of the Smart Middleware. This added layer can, for example, contain standards on data representation, enforce common quality certifications, or define peer review rules to assess data quality. The data space specific services tailor the ecosystem beyond simple sharing of assets; they make sure those assets become valuable to participants.

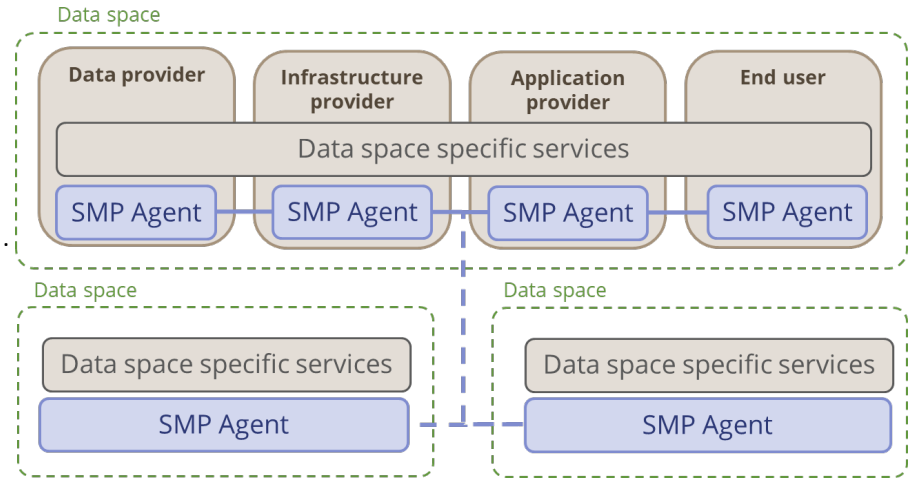

Figure 1. Illustrative representation of how the SMP provides services within and across data spaces.

As discussed above, the middleware does not only aim to be used to build data spaces, it also creates interoperability between data spaces. As multiple data spaces incorporate the SMP, data spaces become more connected. This enables services to cross the boundaries of specific data spaces. Such services will initially be

[5] Open DEI, https://design-principles-for-data-spaces.org/, "*Design Principles for Data Spaces – Position Paper*", April 2021
[6] For more details, please refer back to the previously submitted Business Use Cases and Business Processes document.
[7] Note that the representation of the SMP Agent at this stage does not accurately represent the system components of the SMP architecture. It serves to represent the connection module that actors need to install to become part of the platform. Chapter 6 later describes the exact system architecture.

more limited, as the middleware cannot capture the details of all different data spaces. It will be up to the user to deal with the specifics of each data space in interpreting the assets that it obtains.

To make this illustrative view more tangible, Figure 2 presents an example of how a set of distributed actors might interconnect to form a data space. It is important to note that this figure displays one possible scenario of many possible ways different participants might interact. The number of participants of a data space or the number of stakeholders behind a single actor is only limited by its technical feasibility. This implies that large numbers of participants and stakeholders can interacts simultaneously.

The *SMP Agent* in the figure serves as an abstract component that actors need to deploy to become part of the ecosystem[8]. Chapter 5 describes the actual building blocks of the Smart Middleware Platform. As modular design is a key architecture principle, actors may even choose to deploy only a partial set of the SMP's building blocks.

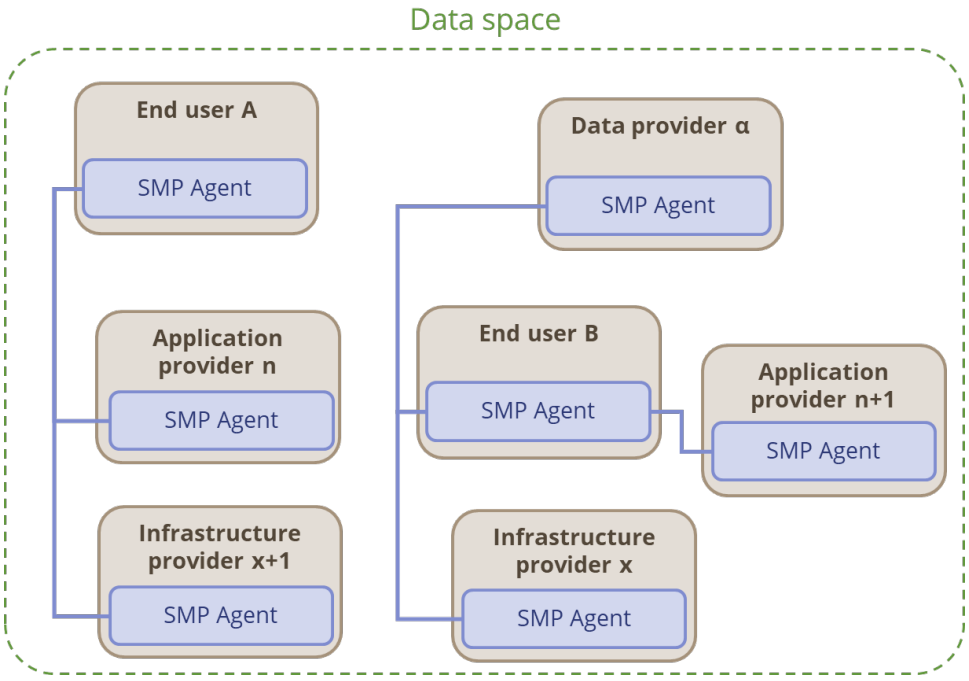

Figure 2. Practical example of how the SMP gets deployed within a data space.

It is important to note that each of these displayed actors are an abstraction to the internal systems of one or more stakeholders. The deployment of the middleware in a data space can have various degrees of granularity. The stakeholder behind an actor can be an individual user that has the capabilities to deploy an *SMP Agent*, or can be an entire data sharing initiative on its own. It is up to the data space governance to decide how the SMP best provides value, and what level of granularity of the deployment fits best. Some conceivable examples clarify this central discussion:

- Imagine that cities throughout Europe want to create smart cities to monitor traffic within their city centre[9]. As there is an interest towards this data from several European stakeholders – individual citizens, regulators, researchers that model traffic patterns, etc. – the collected information is made available in the EU Mobility Data Space. Various possibilities arise when considering how this data is shared to the data space. Deploying the SMP directly on the IoT sensors that collect the data seems infeasible, as SMP requirements to cover the challenges of deploying on resource-constraint devices will only be addressed later. Therefore, the raw data will be consolidated at some centralised servers. One can make the decision to deploy an *SMP Agent* on a server that consolidates the data for a single city. One can also decide to deploy a single agent on a data lake that consolidates city centre traffic information for a whole mesh of cities per Member State. Which granularity level fits best, cannot be

[8] This architecture does not accurately represent all system components of the Smart Middleware Platform. Chapter 6 describes the exact system components in detail.

[9] Developing smart cities and communities is an active domain of work for the European Commission. An example in line with this work is the development of an interoperability framework for smart cities (https://joinup.ec.europa.eu/collection/nifo-national-interoperability-framework-observatory/news/eif4scc-smart-cities-communities)

stated at this point in time so the SMP will allow both possibilities and any intermediate level in between.

- Optimisation of the supply chain can be a major driver behind cross-company data exchange. As part of the EU Industrial Manufacturing Data Space, several companies might look to increase the efficiency of the maritime supply chain[10]. In this setting, the SMP can serve to enable the logistics data exchange between companies from which more efficient transportation services can be calculated. A reasonable deployment of the middleware might be a single *SMP Agent* per company. Alternatively, transportation companies might first build an internal federation, and open this federation to industrial producers via a single *SMP Agent* instead of an agent per transportation company.
- Energy producers, network operators, and energy suppliers have a common interest to share data for example to monitor energy grid load and minimise the risk of grid failure. Partnerships between such entities already exist today, such as the partnership of energy companies in Gaia-X[11]. The SMP has the ability to connect isolated initiatives across Europe to form an EU Energy Data Space. However, in this deployment, each existing initiative will have to assess how best to connect to the SMP and to what degree the SMP becomes part of already existing federations.
- The Destination Earth initiative deals with the complex tasks of creating a digital twin of the Earth on which resource-intensive simulations can be run. Due to the complexity of this initiative – the high-performance computing resources required as well as the immense scale of the data size – an internal federation between resources is under development. This internal federation handles the specific complexities that come forward in this use case. Therefore, for Destination Earth, a connection to the SMP through a single gateway may fit best. The internal middleware of Destination Earth abstracts the internal complexity away from the SMP. However, the Destination Earth initiative does open up towards an increased user base by connecting to the SMP.
- The European Open Science Cloud is an example of an already existing data space that aims to host and process research data. EOSC already builds upon an internal middleware, specifically tailored to its use cases. Again, in this instance, a shallow deployment of the Smart Middleware Platform may be suited best. Similar to Destination Earth, a single gateway can serve as an interoperable connection between EOSC and other EU data spaces.
- The SMP and AI initiatives will come together in two contexts. First, the AI-on-demand stakeholders can be stakeholders within a given data space. As such, by deploying SMP, they will participate in the data space. For example, in the Mobility Data Space, they could then run AI algorithms for traffic prediction. Second, the AI4EU initiative will expose its tools to one or more data spaces. This would typically be done through the marketplace financed through a DIGITAL grant or through other means. In this case, existing stakeholders of a given data space would have access to the AI tools through the marketplace, and use them over data accessed through their SMP agent.

The deployment model of the Smart Middleware Platform is a discussion that has to be deepened on a per-use-case basis. Note that the deployment granularity will affect the type of information end users will experience when gathering for example data. The more data that is consolidated and processed by internal federation, the more raw data becomes information. Taking the example of smart cities, a deployment of the *SMP Agent* close to the IoT nodes will expose very raw data, while a deployment at a higher level will expose interpreted data that has been transformed into traffic information.

The granularity of the deployment that fits best for a use case, heavily depends on the federation that is already in place in a data space, the complexity of the internal resources, the level of consolidation that makes sense to end users, and policies of stakeholders. Additionally, a roadmap can be carved out for existing systems on how the SMP can further penetrate into the internal federation to the extent that makes sense.

As the Smart Middleware design cannot take into account all the intricate details of every possible use case, the SMP architecture vision abstracts them into the four actor groups that are presented. What these actors practically represent can differ greatly per use case. As such, no assumptions are taken on the size of an actor. This makes the middleware agnostic to specific use cases, and increases flexibility to be suitable for a broad range

[10] An example of projects to increase logistics efficiency is *The Physical Internet* project by *Imec*. More information can be found at https://www.imec-int.com/en/articles/physical-internet-next-gen-vision-logistics

[11] Gaia-X Association, https://www.gaia-x.eu/sites/default/files/2021-08/Gaia-X_DSBC_PositionPaper.pdf#page=50, "*Data Space Business Committee – Position Papers",* 13 August 2021

of use cases. Different initiatives that can leverage the Smart Middleware Platform are being developed in different timeframes. The SMP's flexibility allows it to integrate with both existing systems, as well as new initiates that can use the SMP from day one. Figure 3 visualises how the flexibility can support a widespread set of use cases.

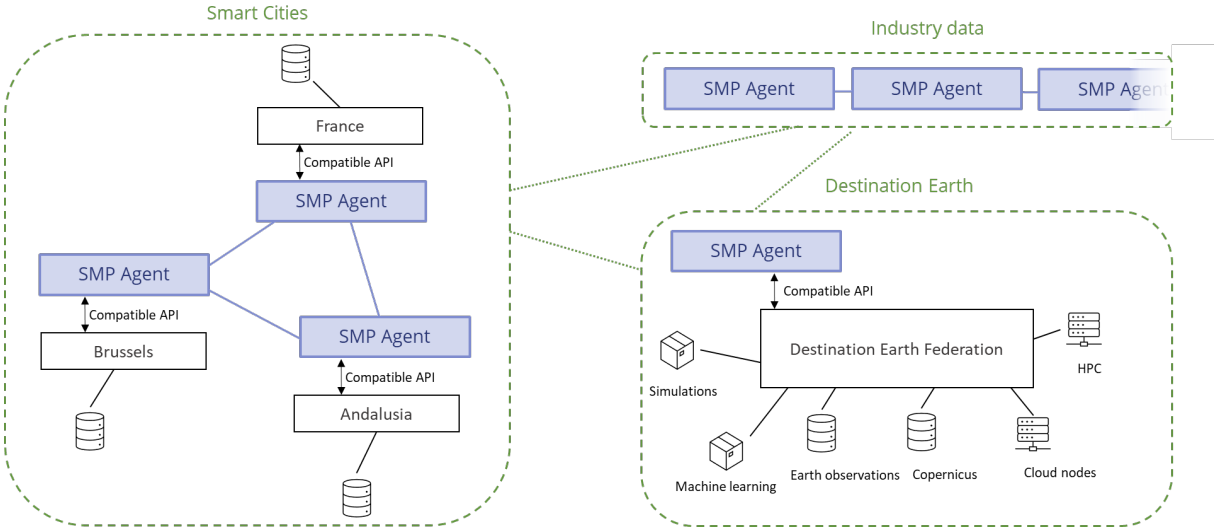

Figure 3: SMP agent deployed and connected over multiple data spaces

4 Architecture design approach

4.1 Approach and methodology

The architecture design was created through multiple interviews and interactive workshops with the active involvement of the project stakeholders[12]. The entire process – from the identification of the actors to the finalisation of the conceptual architecture design – can be divided into six steps:

1. Create context diagrams → the actors interacting with the Smart Middleware Platform were identified through brainstorm exercises with the project stakeholders. This was followed by the creation of the context diagrams displaying the Smart Middleware Platform and its actors (both system and human actors);
2. Identify epic use cases and sub-use cases → through brainstorm exercises and interviews with the project stakeholders, a representative (but non-exhaustive) list of sub-use cases was established for the epic use cases in scope[13]; which comprise the different actors and their potential interactions with the Smart Middleware Platform;
3. Create use case diagrams → context diagrams of the epic use cases and sub-use cases were created depicting the various interactions of the identified users;
4. Identify functional blocks → with the illustration of the various use cases, the functionalities that could be translated into common functional blocks of the SMP (e.g. search, authentication, billing, etc.) were listed;
5. Analyse system components → the identified common functionalities were mapped to the SMP's system components;
6. Design SMP's conceptual architecture → with the mapping of the functionalities to the system components, a high-level conceptual architecture was defined with detailed explanations on the building blocks that support the required functionalities.

The conceptual design was presented to the project stakeholders, who validated the design and provided additional input to fine-tune it.

4.2 Architecture principles

The design of the conceptual architecture of the Smart Middleware Platform is built upon ten architectural principles. Each of these principles is applied throughout the Smart Middleware's design. They are all equally important to the design. Figure 4 provides an overview of these principles:

[12] The project stakeholders include DG CONNECT and an internal ad-hoc taskforce dedicated to this project (people in charge of cloud, edge, data spaces, AI on demand, EOSC, HPC and eGov). Various meetings were held with these stakeholders to present, discuss and iterate on the use cases, their context diagrams and the smart middleware platform's business processes.

[13] Four main use cases are currently in scope for the preparatory work in view of the procurement of the Smart Middleware Platform: (i) Data Spaces, (ii) Destination Earth, (iii) EOSC and (iv) AI on demand.

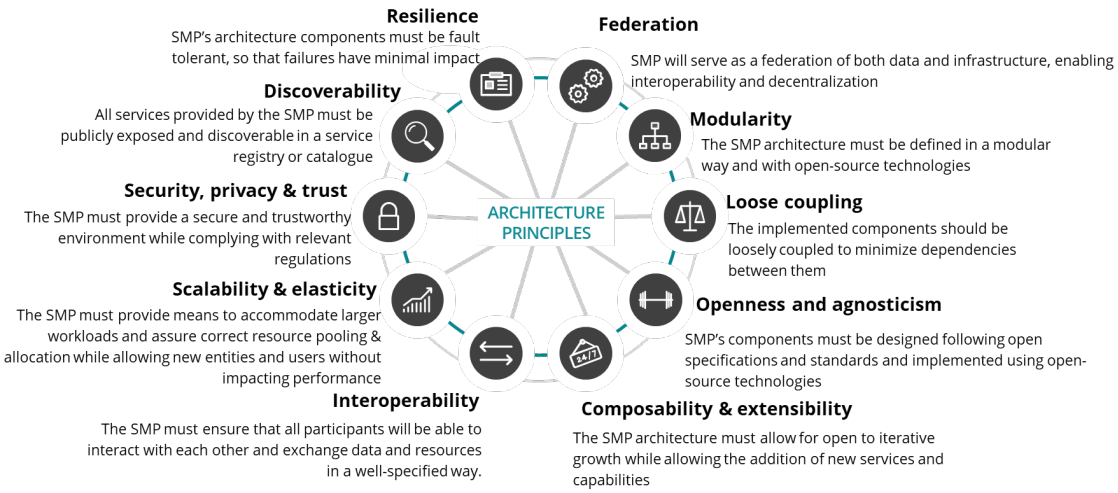

Figure 4. Architecture principles for the design of SMP

- **Federation**: Federated systems describe autonomous entities, tied together by a specified set of standards, frameworks, and legal rules. The Smart Middleware Platform should federate data, infrastructure and applications. This principle is key to enable interoperability and information sharing among the different entities that will be part of the SMP, while giving maximum autonomy to service owners.
- **Modularity**: The architecture of the Smart Middleware Platform needs to be defined in a modular way which allows the replacement or addition of components without affecting the rest of the system. This also provides the possibility to implement every component with a different open-source technology. Through modularity, SMP users are able to deploy a specific subset of components that are tailored for their purposes.
- **Loose coupling**: Components and services should have minimal dependencies on each other. Standardised, business-oriented APIs make sure consumers are not impacted by changes to services. This allows service owners to change implementation, switch out components, or modify data records behind the APIs without downstream impact to end users. This principle ties in with the *modularity* and *resilience* principle.
- **Resilience:** Components of the architecture must be fault tolerant, such that failures in one of them will have minimal impact on other components. Single points of failure need to be avoided to the maximum extent possible as the main objective is achieving a distributed architecture.
- **Openness & agnosticism**: The open specification allows insights into all parts of the architecture without any proprietary claims. It makes adding, updating or changing components easy for all users. Services should be provided irrespective of specific technologies and should be executable in all environments.
- **Composability & extensibility**: The Smart Middleware Platform architecture should allow services to deliver value to the business in different contexts, providing the necessary tools to facilitate their composition together with other services to form new aggregated services. The Smart Middleware Platform remains open to iterative growth allowing the addition of new services and capabilities that fit future use cases to the platform. An open development community should be promoted in order to enable the contribution of new features that extend the Smart Middleware's functionalities by its members.
- **Interoperability**: The Smart Middleware Platform enables interoperability between its participants to share resources in a well-specified manner. The architecture should describe the technical means to achieve this and be agnostic to the specific implementation details of each participant.
- **Scalability & elasticity**: The Smart Middleware Platform provides the means to accommodate larger workloads and allow new entities and users on the platform without affecting the performance. Both vertical scaling – i.e. the practice of adding more resources to a single node – and horizontal scaling – i.e. the process of duplicating nodes – should be possible. The SMP's performance should be able to follow user demand without deteriorating.

- **Security, privacy & trust**: Users of the Smart Middleware must be confident that when they interact with other entities they are doing so in a secure and trustworthy environment and in full compliance with relevant regulations. Data confidentiality, availability and integrity must be guaranteed. Privacy of data subjects, SMP users, or individuals must be assured.
- **Discoverability**: All services that are deployed in the Smart Middleware Platform will be 'publicly' exposed and discoverable in a service registry or catalogue. In this context, 'public' is seen as visible by all approved participants of a data space, not the public internet. Services will adhere to a service description, providing interested parties with a clear understanding of their business purpose and technical interface.

4.3 Fundamental requirements driving the architecture

In a previous phase of this study, a set of business requirements have been listed through desk research, stakeholder interviews and workshops. This section emphasises some of the functional business requirements that drive the design and building blocks of the Smart Middleware Platform[14]. Only a small part of the business requirements are reiterated in this section, focussing on functional business requirements. Business requirements that are consolidated into the architectural principles or that address legal compliance are left out. The following requirements are translated into capabilities in the design of the conceptual architecture:

- **ID-6** The SMP shall provide access to secure, resource-efficient data transfer services.
- **ID-24** The SMP shall enable data consumers and data providers to define data quality rules.
- **ID-25** The SMP shall enable to share data within and across data spaces.
- **ID-29** The middleware shall provide search functionalities within data spaces.
- **ID-30** The middleware shall enable access to real time/streaming data in a controlled way.
- **ID-31** The middleware shall provide tools to handle metadata and vocabulary transparently.
- **ID-33** The middleware shall provide data anonymization and masking services.
- **ID-79** The middleware shall list and document the available services and datasets.
- **ID-81** The middleware shall provide catalogue and resource discovery services.
- **ID-13** Authorised users shall be able to run AI models, simulations or algorithms through the SMP in their chosen IT infrastructure.
- **ID-14** Authorised users shall be able to run AI models, simulations or algorithms through the SMP in 3rd party HPC.
- **ID-40** The middleware shall enable infrastructure providers to provide storage services/ access through the SMP.
- **ID-41** The middleware shall offer an abstraction layer to give access to (edge, cloud) infrastructure services (computing & storage services).
- **ID-18** The middleware shall enable to audit the use of the SMP.
- **ID-19** The middleware shall enable to monitor the SMP performance (both technical and environmental) and ensure performance and quality of service in the execution of applications across multiple cloud and edge providers.
- **ID-20** Middleware reports shall be generated for reporting to data space governance bodies and SMP distributed actors.

[14] For a full list, refer to "D01.04 – Final business requirement"

- **ID-26** The middleware shall enable authorised users to download data.
- **ID-48** The middleware shall provide a mechanism to define and share service level agreements between its participants.
- **ID-53** The middleware shall provide a licensing and license asset management.
- **ID-66** The middleware shall federate the authentication which enables all external systems to maintain their level of autonomy, there shall NOT be a single centralised authentication method implemented.
- **ID-67** The SMP shall enforce access control rules for the infrastructure running the middleware.
- **ID-71** The middleware shall handle the authorisation of the end users by enforcing the access rights policies.
- **ID-73** The communication with and within SMP components shall be encrypted.
- **ID-85** hall make sure that specified usage restrictions and obligations are realized even after access to data has been granted (data-centric usage control).
- **ID-78** The CSIRT shall be established. The SMP CSIRT will conduct Threat Vulnerability & Risk Assessments (TVRAs) for identifying relevant attack vectors, with the aim to reduce the attack surface of the SMP. All proper incident response methods will be followed.

5 Conceptual architecture

The following sections elaborate on the conceptual architecture of the Smart Middleware Platform. They present the capabilities of the SMP and the building blocks that support these capabilities. It is important to remark that the conceptual architecture lays out the capabilities of the Smart Middleware Platform as a whole. Realizing these capabilities may require several system components within a data space network. Chapter 6 discusses how the conceptual architecture maps to several system components.

5.1 Architecture overview

Four architectural layers describe the Smart Middleware Platform (Figure 5): the data layer, the infrastructure layer, the administration layer and the governance layer.

The **data layer** encompasses the building blocks to enable the exchange of data assets and applications. The SMP offers data consumers the means to access different types of data from different providers, enabling interoperability between providers and consumers. The layer contains the services to share as well as manage data and applications, and perform basic analysis.

The **infrastructure layer** has similar responsibilities for managing infrastructure assets. This layer allows the SMP to connect to third-party infrastructure services, such that end users can execute applications on them. The middleware does not provide infrastructure itself, but allows infrastructure providers to open up internal infrastructure towards data space participants. Both elemental computing and storage resources (e.g. virtual machines, file system storage) as well as PaaS services (e.g. databases, AI hardware) can be provided. The infrastructure layer allows end users to discover, utilize, and manage infrastructure services offered by infrastructure providers.

The **administration layer** vertically spans the data and infrastructure layer. It provides services that are required for the well-functioning of those layers, as well as the SMP as a whole. These services regard security, identification and access control, monitoring, and more. The administration layer enables data spaces to federate the different actors and allows actors to operate their components in the data space.

Last, the **governance layer** addresses transversal functionalities that apply to all the forementioned layers as it provides means of protentional contingency against issues the participants might find while using the SMP. The two main capabilities in this layer are composed of human taskforces that either support the participants in finding solutions to technical problems or trigger a response action against security threats in the form of a Computer Security Incident Response Team (CSIRT).

Each of these four layers is further detailed in the following sections. The services of each layer are subdivided into capabilities. These capabilities group a set of technological building blocks that are required for the SMP to provide that capability to users. The next sections describe how the capabilities interact and offer functionalities to end users and asset providers. Details on the smallest scale building blocks are explained on Annex I .

Two types of capabilities are distinguished: user services and supporting services. The user services represent the services that are offered by the users of the Smart Middleware Platform. Not all users will need the provided services of the SMP. Organisations acting as a data provider will, for example, not require the user services of the infrastructure layer. The supporting services enable a correct functioning of the user services. They do not add immediate value to SMP actors, but run in the background to support the user services. SMP actors also do not directly interact with the supporting services.

It must be noted too that another distinction is made in each of the layers between services that can be accessed through the SMP and services that are built into the Smart Middleware Platform. The former are depicted with round edges and the latter with squared edges in the following sections (starting from Figure 6). This is an important distinction, as services that are accessed through the SMP translate to software components that are compatible with existing solutions through their APIs. Services that are built into the SMP require the absorption of possible existing solutions into the software stack of the platform.

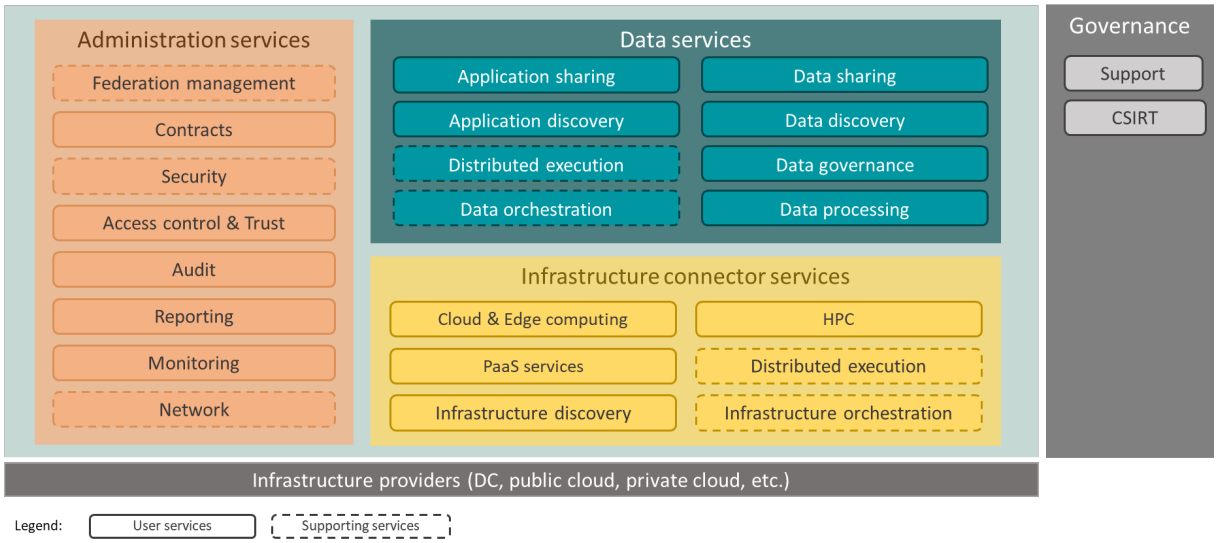

Figure 5. High-level overview of SMP capabilities and architecture layers.

5.2 Data layer architecture

Figure 6 depicts the technological building blocks that form the data layer of the Smart Middleware Platform. The building blocks are divided into user services and supporting services. User services represent the services that data consumers and providers are offered and can leverage. The supporting services relate more to background services that support the offering of the user services. The services presented in the visual are also differentiated between services that can be accessed through the middleware (rounded edges) and services that are built into the Smart Middleware platform (squared edges).

Two prominent capabilities are the **Application sharing** and **Data sharing**. These contain the building blocks required for providers and consumers to exchange both data and applications. The capabilities create the connection between stakeholders to share data and application assets. The data sharing capability encompasses several functions including management of various types of data sharing, from transferring a single, few megabyte file, to transferring a terabyte-sized data dump. The SMP architecture foresees a simple data transfer mechanism and two special types of data transfer: bulk transfer and data streaming. A datastore connector handles the connection to the backend data store of the data provider, which can vary from a simple file storage to a relational database system. Additionally, **Data processing** tools address in number of scenarios where it will be desirable to process data as near as possible to the source. Among these tools, data anonymization tools support data providers and consumers to protect the privacy of data owners. On top, **Data governance** tools are offered by the middleware for end users who can verify the integrity and quality of the required assets.

Sharing applications is similar to sharing data. Indeed, at their core, applications are no more than a collection of data that is marked to be executable. However, specifics of application sharing come in terms of formats and the fact that usually multiple files need to be combined correctly to be able to run the application. It also adds additional considerations to handle the security of executable code and the trust that End users have in the provided application. The SMP will define the procedures to use for sharing applications. The application sharing capability considers three types of applications: full-fledged software packages, isolated algorithms, and machine learning models. Each type of tool comes with different specifics and runtime environment requirements that the SMP should adhere to.

Consumers can find assets of a data space through the tooling provided by the **Data discovery** and **Application discovery** capability. First, providers make their data assets discoverable by submitting well-structured metadata description of their services – in a standardised format. Then consumers can query the catalogues to find suitable assets within the data space. These catalogues describe the content of the assets, how to consume them, and the policies that apply on this usage.

Additional services of the data layer orchestrate data assets across SMP actors. The **Data orchestration** and **Distributed execution** capabilities allow actors to pool together data from different sources and manage partial sets of data across infrastructure providers when executing distributed applications. The combination of these

capabilities allows end users to gather data from different providers and spread it over distributed infrastructure where data is fed into an application.

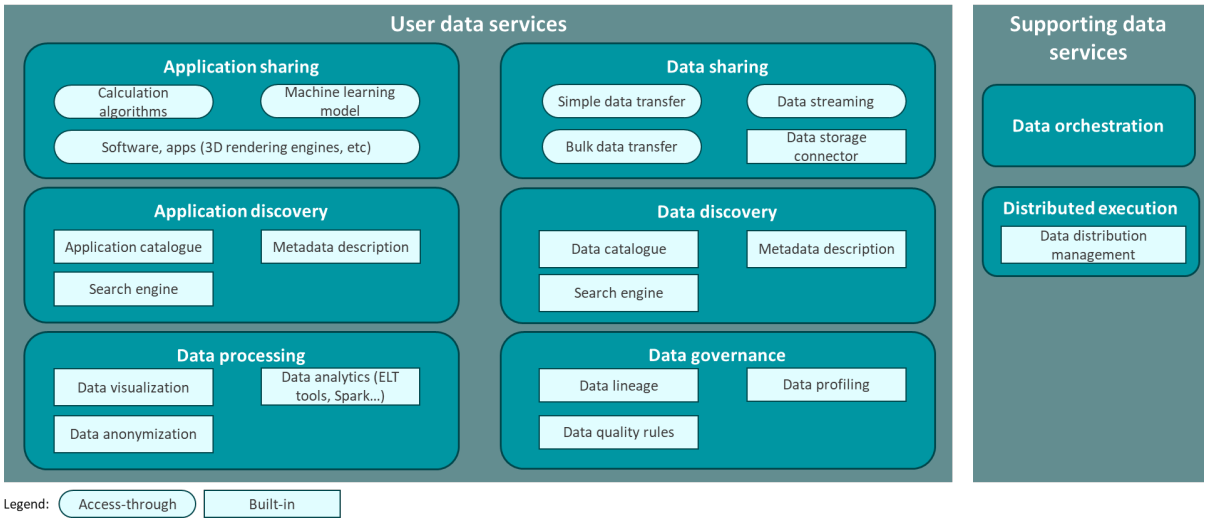

Figure 6. High-level view on the data layer building blocks

5.3 Infrastructure layer architecture

This chapter focuses on the building blocks of the SMP's infrastructure layer, as well as the dependencies and interactions between the different infrastructure building blocks on a high level. The services provided by this layer enable the end users to easily find and provision the necessary computing and storage resources to execute their workloads in a secure and energy-efficient way. The services presented in the visual are also differentiated between services that can be accessed through the middleware (rounded edges) and services that are built into the Smart Middleware platform (squared edges).

Figure 7 gives an overview of the infrastructure services provided by the SMP. As the SMP is responsible for the orchestration of the services from various infrastructure providers, these are depicted at the bottom of the picture, forming a base for the services. Regarding the infrastructure services, it is important to differentiate the user infrastructure services from the supporting ones; while the former can be accessed by a wide range of users, the latter are services executed in the background, without directly being called by a user.

The **supporting services** of the SMP are grouped into two different capabilities. The *infrastructure orchestration* and the *distributed execution* are services executed in the background whenever they are required by any of the core services. The first capability, the *infrastructure orchestration*, automates the provisioning of the infrastructure services to enable the various infrastructure providers to interconnect and get exposed via a standard interface. The second one, the *distributed execution*, allows the user to deploy applications and execute computations close to the data.

The **user services** available for the end users can be divided into four capabilities. The *infrastructure discovery* can serve as an optimal entry point for the user, as its purpose is to make it easier to discover all the available infrastructure services within a data space. Users can find the provided service that fits their needs by searching and filtering specific metadata, such as information on the particular infrastructure provider or the description of the service. The infrastructure catalogue, and therefore the SMP, will offer a wide range of services. The *cloud & edge computing* capability provides the opportunity to provision various resources to execute computations or store data in the environment of their choice. The *platform-as-a-service services* provide several database engines and other platform-level resources. Finally, the *HPC* capability permits the user to perform complex calculations at high speed by providing a cluster of high-performance computers.

The infrastructure services can be easily combined with each other to create even more value for the end user. For instance, after successfully analysing certain data with the help of the provisioned *PaaS* analytical services or the *HPC* capability, the user may want to store the used datasets and/or the results of their calculations. In this case, the *PaaS* storage services can easily fulfil the storage needs of the end user. In case a user would like to develop a stand-alone application, they may also use various *PaaS* services at the same time. They can leverage the different storage options to store each sort of data in the most efficient manner (e.g. the transactional data

in a transactional database, while the sensor data in a NoSQL database). Besides, they can use the *cloud & edge computing* capabilities to deploy and run their applications, and the *distributed execution* capability even enables them to run the code close to the edge.

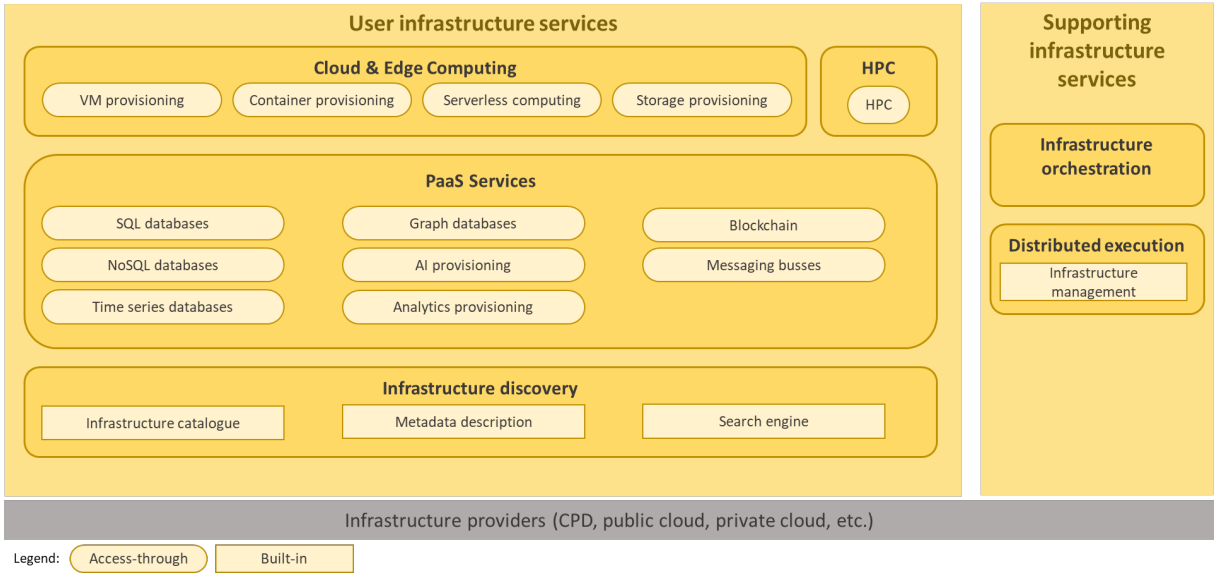

Figure 7. High-level view on the infrastructure layer building blocks

5.4 Administration layer architecture

The administration layer depicted in Figure 8 addresses the main capabilities that will assure the correct delivery of services within the SMP. Despite being depicted as isolated bubbles, capabilities and building blocks are strongly related and further discussed (see Annex I). The function of this layer can be described as a set of supervision and management functions that will lead to a better control and interoperability of the rest of the SMP services. Along with the later described governance layer, it will play an executive role within the SMP.

Whenever a user accesses a service through the SMP, the **Contracts** capability will determine if the licenses are correctly delivered. It will also register the billing terms and will manage the service level agreements as specified by the providers, as well as managing the permissions related to data sharing. The usage contracts will be accessible through this Administration layer every time an end user requests a data or computing service.

In the widespread context of the SMP, **Security** is crucial to protect EU resources. As it will be described in Annex I, security capabilities are constantly active. The agreed level of security encryption and integrity on every data transmission or service deployment must be granted for the consumer at any time. For this reason, any provision of a service will be inevitably deployed along with the security block, providing strong end-to-end security guarantees of all data that is handled by the middleware.

Related to the security capabilities, the functions gathered by **Access control & Trust** capability will be constantly required whenever an end user or a provider accesses the SMP. Attributes and user roles such as Role-Based Access Control (RBAC), as well as authorizations to proceed with an action are addressed here. In this sense, every relation of the user with a data layer service or infrastructure provisioning is closely screened by the Access Control & Trust. The middleware will provide identification, authentication and authorization (IAA) services for communication between data space participants, and integrate existing IAA systems of participating organisations for IAA of users within the organisation.

The **Reporting** and **Monitoring** capabilities are strongly interconnected. In the case of Monitoring, it is regarded as the real time information collection and screening that will register alerts and usage information concerning other layer´s services, as well as energy and quality of service optimization. On the other hand, the Reporting capability will handle the historical record of such information, as well as the general platform usage, allowing the relevant stakeholders to export and log the extraction of the information obtained. These two capabilities could be visualised as a supervision of the processes taking part at every layer level within the SMP. Should additional action be taken, the Governance board will resolve the situation with the assistance of other Administration layer blocks such as Security or Contracts.

In order to analyse if the services given by the SMP are meeting contracts, access or security requirements, an auditing tool in the **Audit** capability will be capable of receiving information inputs from the reporting and monitoring building blocks. By comparing what is expected from the SMP services and what is actually happening at a service delivery level, the audit capability will interact with the Reporting and Monitoring capabilities.

The **Federation management** capability encompasses the general administration orchestration, as well as the supervision of the services and administration layer connection. It will oversee that the main principles of federation and interoperability are met by providing the means to connect resources. The federation management will encompass the needed configuration parameters for a well-functioning of the SMP components. Such parameters may include the servers to connect to, rules concerning the lifecycle of recorded data, network configuration, and other parameters.

Finally, the **Network** capability contains building blocks for the establishment of secure network connections using technologies like virtual private networks. Additionally a firewall protects unwarranted access to the SMP components and backend services. These network capabilities are relevant for connecting to infrastructure services, as well as setting up the communication channels for data and application transfers

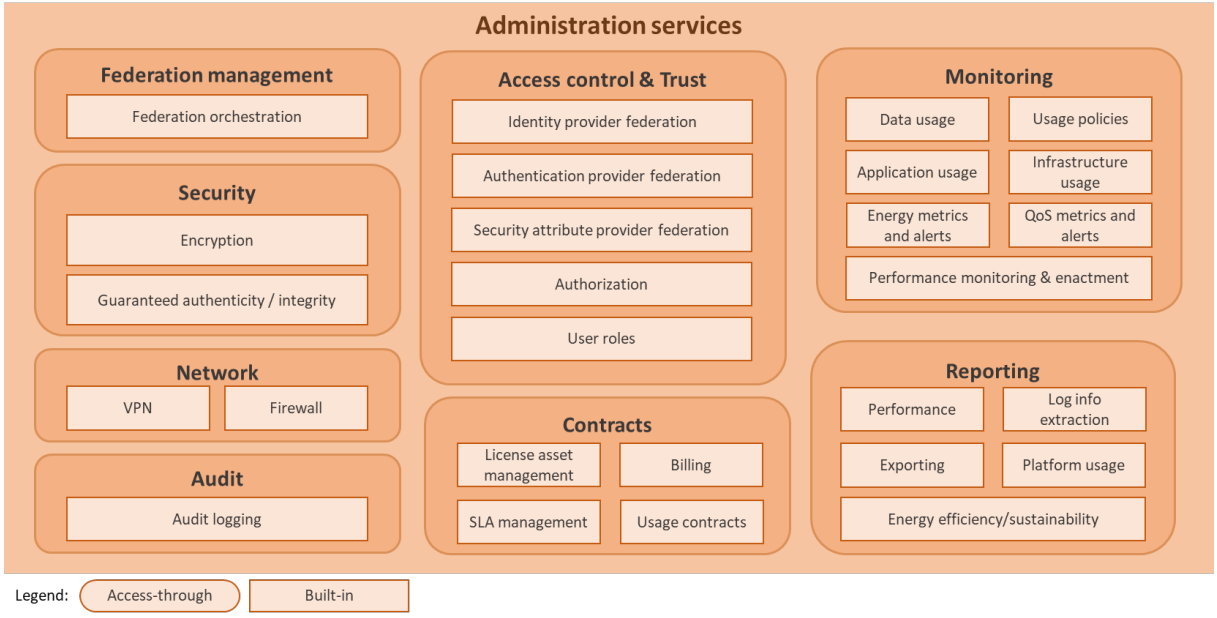

Figure 8. High-level view on the administration layer building blocks.

Crucially, the Administration services layer interacts with the Data and Infrastructure as well as the Governance layer. Previous sections have discussed how the Data and Infrastructure layer are organised. The data layer provides services related to applications and data, whereas the infrastructure layer provisions computing, storage, and other infrastructure services. When a data consumer needs to access any of the services belonging to the Data layer, the Administration Access control & Trust capability is triggered to assure the right permissions are given. Additionally, authentication and authorization mechanisms are a prerequisite for executing the Data layer services, such as the Application or Data sharing capabilities.

Once the Administration layer confirms the identity and the role of the user, the contract capability is activated in order to verify the terms agreed and the Service Level Agreements. After this process, data services can be executed. The explanation above concerning data services is applicable to the Infrastructure layer services in case the end user is intending to access computing, storage, or other infrastructure resources. Nevertheless, it will be common to combine data and infrastructure resource usage by means of a distributed execution. It must be noted that beyond the usage of data and infrastructure services in combination, independent use of infrastructure or data alone will be a possibility as well. In these cases too, the Administration layer will perform consistent and continuous analysis of the usage with the help of the Reporting and Monitoring capabilities. Along with the Auditing capability, the Administration services will extract valuable information about the data and services exchanged between the consumer and the data, application, and infrastructure provider(s). This will include analysis of the usage kept in metrics logging and real-time monitoring, which will be transferred to the reporting capability for a wider history data recording. In this reporting function, performance of deployed services, the level of efficiency or the usage of the platform and the sustainable utilization will be supervised by the administration.

When considering the security and orchestration needs for the global usage of any of the Data and Infrastructure layers building blocks, the Administration Security and Federation management will be responsible of the correct encryption and verification deployments, assuring as well a correct allocation and interoperability of the resources. It must be noted however that both the Data and the Infrastructure layers will hold their own local orchestration blocks as explained in their respective sections.

On balance, the Administration layer will play the role of a process supervisor, assuring that any resource usage by the consumer is correctly delivered in terms of security, policies and contracts. With the addition of Reporting, Monitoring and Auditing capabilities, any misconduct or abnormal behaviour will be addressed by this layer, having therefore a sort of executive role.

5.5 Governance layer architecture

The following section presents the SMP's architecture building blocks on the governance layer. The services provided in the governance layer enable the support and management of the Smart Middleware Platform. The governance layer can be divided in two categories of governance services: Support services and CSIRT (Computer Security Incident Response Team).

The **Support services** in the governance layer are associated with the administration layer of the architecture and will assist the SMP users when issues arise during installation of the SMP and during the use of other services from the SMP. Three services are currently foreseen in the support category, the first service is a Support webpage where SMP users will have access to. The second support service is a Ticketing system where SMP users will be able to log and keep track of issues regarding the SMP and the third service is a Helpdesk where SMP users can connect to in case issues remain unsolved.

The **Computer Security Incident Response Team** services is the second category of governance services which includes the incident response and threat monitoring building blocks. The Incident response building block will have procedures to respond to security incidents to restore the compromised SMP functionality as soon as possible. The second service, Threat monitoring, will proactively monitor and follow-up on possible malicious activities that could affect the SMP to avoid potential security breaches.

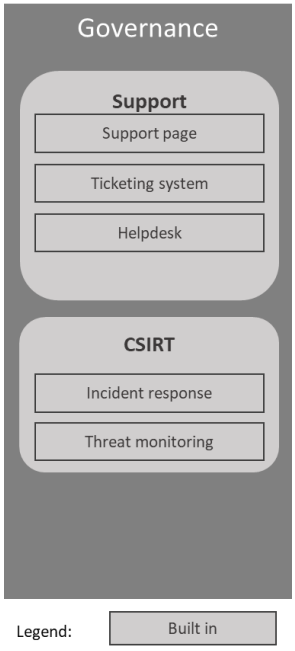

Figure 9. High-level view on the governance layer capabilities

6 System architecture

The previous chapter described the capabilities of the Smart Middleware Platform as a whole. Some of the functionalities however, require several system components in the architecture to realise the capabilities. This chapter elaborates on the needed system components and how the capabilities are mapped to these components.

The Smart Middleware Platform requires some centralised services in data spaces to support particular capabilities. However, it should be noted that these centralised services are given minimal responsibility as the goal of the Smart Middleware Platform is to become as decentralised as possible.

6.1 Centralised system components

Three types of centralised components were identified to support the distributed functionalities of the Smart Middleware Platform: (i) data, application and infrastructure catalogues, (ii) vocabulary providers, and (iii) identification authorities. While these components are centralised for each data space, it should be noted that they are not necessarily exclusive. Multiple instances of these components can be deployed by multiple organisations or even, be implemented by distributed components to increase availability and scalability, however always offering a logically centralised view of service. Nevertheless, a peering mechanism must be put in place such that distributed actors receive a harmonised service. Note that the centralised services are not centralised across data spaces, each data space has its own set of centralised services to support it. There are no central services above different data spaces.

6.1.1 Data, application, and infrastructure catalogues

The three catalogues are best supported through centralised components. Each of the catalogues can be considered a separate system component, but all catalogues may be hosted by the same organisation. The catalogues should be built in a scalable manner and should be designed and deployed for high-availability with built-in redundancy and failover procedures. These catalogues thus support the *catalogue* and *search engine* building blocks of the *Data discovery*, *Application discovery*, and *Infrastructure discovery* capabilities.

The catalogues accept the submitted metadata from data, application and infrastructure providers and collect this submitted information in a structured data storage. The catalogues are responsible for verifying whether the submitted information complies with the metadata standards of the data space. Additionally, the integrity of the service descriptions must be verified, through digital signatures of the provider, before persisting the submitted information in the catalogue. End users can query the content of the catalogues through the search engine and accompanying interface. End users should prove possession of a valid identity in the data space before accessing the catalogues' information.

As stated in the introduction of this chapter, the central catalogues are not intended as exclusive instances within a data space. Multiple organisations within a data space can deploy a catalogue. Peering these multiple catalogues together can be done by duplicating the structured storage across the multiple catalogues. This allows query resolution at all instances of the catalogue because all needed information is stored locally, but close attention needs to be paid on synchronising the content of the different instances.

A fully decentralised catalogue was considered in the design of this component. While this is technically possible, it adds complexity to the cataloguing service with little benefits. A fully decentralised service catalogue would induce a high network load for every query that an end user wants to resolve. For every query, the end user would have to contact all other data space participants for their fragment of the service catalogue. Additionally, a mechanism to discover peers in the data space would need to be implemented, as data space participants can join and leave continuously. The added complexities of a fully distributed catalogue are not deemed to outweigh the drawbacks of a centralised catalogue.

6.1.2 Vocabulary provider

The vocabulary provider serves to harmonize the ontologies and vocabularies in the data space. It provides the definition of metadata representation and, if required, the data representation standards. These ontologies are provided to the network in machine and human-readable formats. The standardisations should be laid out by the data space governance, and communicated to the data space participants through the vocabulary provider.

The vocabulary provider thus ensures a correct functioning of the *metadata description* building blocks of the *Data discovery*, *Application discovery*, and *Infrastructure discovery* capabilities.

The SMP Agents remain up to date with current metadata description standards. The data, application and infrastructure catalogues check the submitted services descriptions against the ontologies provided by the vocabulary provider.

6.1.3 Identification authority

Identification, authentication and authorization are of paramount importance within a data space. The identification must be supported by a central authority. This authority is in charge of reviewing the identity details of organisations that want to participate in the data space. If the authority approves the organisation, it provides a proof of identification that the organisation installs in its SMP Agent. With this proof, the participant authenticates itself to other data space participants and other participants define authorization rules based on the verifiable identity. The technologies and cryptographic protocols to support this identification process is outlined in detail in Annex III .

The identification authority plays a role in the *identification provider federation, authentication provider federation,* and *security attribute provider federation* building blocks of the *Access control & Trust* capability.

6.2 Decentralised system components

The remaining capabilities of the Smart Middleware Platform are supported through the decentralised SMP Agents. This component is installed by participating organisations on the infrastructure of their choosing. The Smart Middleware Platform aims to keep the functionality at the edge to the fullest extend to reflect the decentralised purpose of a data space.

Recall that an organisation can assume multiple roles within a data space. Depending on the role that the organisation assumes, a different part of the software stack of the SMP Agent is addressed. As a data provider for example, the organisation does not need the connector to compute resources. When an organisation plans on only assuming a subset of the possible roles, it can choose to install only a part of the SMP Agent's software stack.

6.3 System architecture overview

Figure 10 shows the resulting system architecture overview of a data space that is built leveraging the Smart Middleware Platform. The core of the platform's functionality is supported through the connections between the SMP Agents of the data space participants. The central components are connected to the data space participants to support the full functionality of the data space.

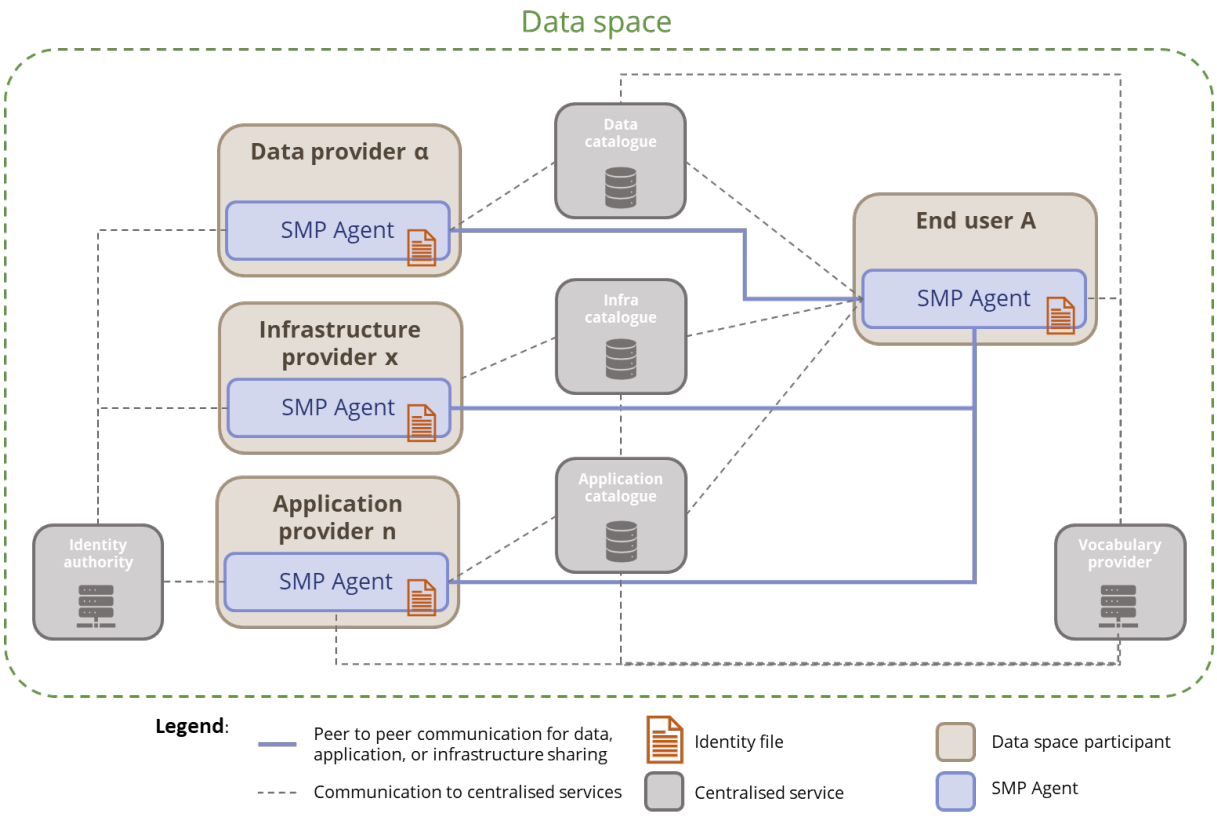

Figure 10. An overview on the system architecture of the Smart Middleware Platform

Annex I Architecture building blocks

This Annex provides detailed descriptions of each of the building blocks that form the SMP conceptual diagrams.

I.1 Data layer architecture

Table 1. Detailed description of data layer building blocks

Data sharing	
Simple data transfer	Simple data transfer services are used for the exchange of small to medium sized sets of data between participants. The size of the data is typically less than a few MB's to 100 MB's. Simple data transfers can happen synchronously using a single network connection between participants.
Bulk data transfer	Bulk transfer services are used for the exchange of large chunks of data between participants. The size of the data is in the range of 100 MB's and above. Bulk transfer services typically proceed as asynchronous processes where data is transmitted in digestible chunks and recombined at the receiver side. Bulk data transfer can be sped up by using multiple network connections.
Data streaming	Data streaming concerns a specific use case where the data exchange is not a singular occurrence, but should happen periodically. Small chunks of data are continuously being transmitted from provider to consumer. Possible sources for data streaming can be sensors or server logs. The use of data streaming typically coincides with real-time and near real-time data processing.
Data store connector	The data store connector foresees the integration with the internal data federation of a data provider. This connector foresees integrations with multiple popular data management solutions, such as NTFS file systems, MySQL or PostgreSQL relational databases, MongoDB key-value databases, …
Application sharing	
Calculation algorithms	The SMP will provide the means to share basic data science algorithms (e.g., linear regression, logistic regression, Random Forest, K-means, …). The participant can download these tools in the infrastructure of their choosing.
Machine learning models	The SMP provides access to an ecosystem to share machine learning models. This is possible if the providers make these models discoverable and apply the correct access control policies such that consumers are authorised to access these machine learning models.
Software & apps	In general, the SMP could provide access to any type of software application. An example is a 3D rendering engine that can process the data from Destination Earth. This is possible if the providers make these apps discoverable and apply the correct access control policies such that consumers are authorised to access these machine learning models.
Data discovery	
Metadata description	Data assets need a metadata description to be discoverable by data consumers. This metadata description informs data consumers about the content of the data, as well as its location, author, usage policy, …

Data catalogue	There will be nodes that expose functional and non-functional attributes that define the data. These nodes expose the metadata description of the data. Such nodes will serve as data catalogues that can be browsed by consumers.
Search engine	The federated catalogue mentioned above implements query algorithms to facilitate filtering to make it possible to find the best assets based on query parameters and matching metadata descriptions.
Application discovery	
Metadata description	Application assets need a metadata description to be discoverable by consumers. This metadata description informs application consumers about the content of the application, as well as its location, author, usage policy, …
Application catalogue	There will be nodes that expose functional and non-functional attributes that define the shared applications. These nodes expose the metadata description of the applications. Such nodes will serve as application catalogues that can be browsed by consumers, much like an App Store where users find information on offered algorithms, machine learning models, and full software or applications.
Search engine	The federated catalogue mentioned above implement query algorithms to facilitate filtering to make it possible to find the best assets based on query parameters and matching metadata descriptions.
Data processing	
Data visualisation	The SMP provides access to open-source data visualizations tools like d3.js or leaflet.
Data analytics tools	The SMP provides access to basic data processing tools like Hadoop Ecosystem, Spark and Jupyter notebook to create scripts for processing data and execute this scripts on shared data.
Anonymization	To ensure the privacy of data subject, data possibly has to be anonymised before it can be shared to certain data consumers. This anonymisation typically involves aggregating data from multiple data subjects such that individual records cannot be recovered.
Data governance	
Data lineage	To monitor and ensure data integrity, tracking errors during data processing, it must be possible to show the complete flow of the data, from start to finish. Data lineage provides a complete overview of the actions that have been taken on the data by all participants.
Data profiling	Data profiling is essentially the execution of the data governance strategy. It involves collecting descriptive statistics on the data, collecting data types, tagging data with keywords, and other steps. It helps in analysing data, implementing a data governance strategy, and determining data quality.
Data quality rules	The consumers can assess the data quality of the offered data. The data quality will be described in the data catalogue according to certain data quality rules. These rules can contain attributes like data accuracy, whether data is complete, up to date, or available.
Distributed execution	
Data distribution management	As the SMP connects data, applications and infrastructure, it should enable a distributed execution of algorithms. This building block manages the distribution of the data assets in such process.

Data orchestration	
Data orchestration	The main building block for taking siloed data from multiple data storage locations, combining them, and making them available to data consumer applications.

I.2 Infrastructure layer architecture

Table 2. Detailed description of infrastructure layer building blocks

Cloud computing & Edge computing	
VM provisioning	The SMP provides an abstraction layer to provision Virtual Machines on the underlying infrastructure. The end users may select the virtual machine(s) which are the most tailored for their needs regarding a set of parameters (e.g. operation system, memory size, network bandwidth, etc.).
Container provisioning	Allows the provisioning & deployment of container (images) in order to launch and stop the execution of container-based algorithms, generic applications or custom code. The containers may be preferred over the VMs for the end users in case the application portability is the most important aspect for them.
Serverless provisioning	Serverless computing abstracts away most of the OS configuration, maintenance tasks, and notion of the underlying instances that software runs on. The SMP offers serverless computing services provided by the infrastructure.
Block storage	The SMP provides an abstraction layer to provision various kinds of storage on the underlying infrastructure. Block storage provides low level storage capabilities to end users to store any type of binary data in ordered blocks.
File system provisioning	The SMP provides an abstraction layer to provision various kinds of storage on the underlying infrastructure. File systems are provided to end users to store files in structured directories.
PaaS services	
SQL databases	Standard relational databases. These are the most suitable option for storing primarily structured, transaction-oriented data, in case the structure does not change frequently and when the data integrity is important. The database options provided by the SMP can be open-source or commercially licensed. A relevant tool example include MySQL as the most prominent open source solution in the current market.
NoSQL databases	NoSQL databases are the best option in case the user needs more flexible schemas and/or horizontal scaling. They also enable faster queries due to the data model. The SMP may offer cloud-native or open-source non-relational databases for the users. Among others, tools like Apache Cassandra or MongoDB provide this type of database management
Time series databases	Databases designed specifically for the storage and retrieval of data associated with a timestamp (time series data). These databases are typically the most suitable for storing sensor data, as they can compress, manage and summarize the time series data, and handle time-aware queries. As this type of data is useful for monitoring and reporting capabilities, tools like Prometheus and Graphana combined become a powerful stack for this purpose.

Graph databases	Graph databases are built to allow an easy/performant way to query relationships. Graph databases use nodes to store data entities and edges to store relationships. The best examples are (social) networks or supply chains.
AI provisioning	The SMP may provide abstractions to allow for a simple deployment and execution of AI models. The end users may select from the AI services provided through the SMP and use them to gain insights from their own data set(s). Many solutions can be found in the LF & AI landscape founded by Linux, for instance TensorFlow.
Blockchain	The SMP will provide blockchain services to enable building applications where a decentralised, shared system of records is needed. Its most important advantages are data integrity, reliability, the speed of the storage, immutability and transparency, therefore, it is ideal for, for example, recording logs for audit and regulatory compliance or documenting payments.
Messaging busses	The messaging busses mediate the message exchange between different systems via a shared set of interfaces (message bus). The sender may publish messages on a queue, which will be transferred to the subscribing receiver(s). Apache Kafka is a relevant solution that specifically implements a messaging system as described.
Analytics provisioning	This building block enables the end users of the SMP to access high-level data analytics services without the need to provision the tooling manually (e.g., ETL services, Apache Spark).
Infrastructure discovery	
Metadata description	The metadata descriptions make the search for the infrastructure components easier and user-friendly by giving the opportunity for the user to retrieve the infrastructure services by searching for the metadata (e.g., keywords in the description, the author/publisher of the infrastructure, the compute power of the provided infrastructure, …).
Infrastructure catalogue	The infrastructure catalogue contains the list of all the available infrastructure services for the end users. It ensures that the end users can discover the existing services in an easy way.
Search engine	The federated catalogues implement query algorithms to facilitate filtering to make it possible to find the best assets based on query parameters and matching metadata descriptions.
HPC	
HPC	In a later phase, the SMP will provide high performance computing power to enable parallel data processing and the conduction of complex, resource-intensive calculations.
Distributed execution	
Infrastructure management	The infrastructure management runs and clusters the underlying networked computers from different providers to create the impression that a single and reliable machine is processing the computations. It allows computations to be executed close to where the data is located.
Infrastructure orchestration	
Infrastructure orchestration	This building block is responsible for automating the provisioning of the infrastructure services needed for the computations conducted on the middleware. It allows the

	various infrastructure providers to interconnect and exposes them via a standard interface.

I.3 Administration layer architecture

Table 3. Detailed description of administration layer building blocks

Contracts	
License asset management	Administrates any topic related to recommended licenses by both the SMP and the Members States, compatibility and harmonization between the different data spaces.
SLA management	Hosts the mechanism that defines and shares the Service Level Agreements between contractors. Administration services will handle SLA-related issues as well as SLA sharing when required.
Usage contracts	It administrates the types of services that will be provided to the consumers by the services providers, and the conditions which consumers must adhere to.
Billing	The administration of any topic related to the contract billing, such as conditions or specifications are included in this building block. Specifications may include procedures, due times, quantities, periods of payments, etc.
Access control & trust	
Identity provider federation	Provides an identity federation for the participants of the SMP. The service of this block include the identity information validation, creation and management.
Authentication provider federation	Federates existing or data space specific authentication mechanisms. Participants will access with a specific token whenever consumption or data search is needed, avoiding unapproved access to data. Additionally, it addresses Infrastructure authentication mechanisms.
User roles	The roles needed to grant permissions and access to determined services within the SMP are assigned at this stage of the Access Control & Trust. Common roles across SMP enabled data spaces will provide interoperability.
Authorization	This building block is responsible for handling the permissions of the different users so that it can be defined, what users what actions are allowed to perform on a specific resource. This building block is of crucial importance, as giving a limited access to the necessary users is one of the ways to keep a system secure.
Security attribute provider federation	For each type of resource, different policies will apply. Often, the providers will define the access control policies by assigned security attributes to users. In general, security attribute providers can be third-parties. This building block federates the attribute providers.
Security	
Encryption	Supports the creation and management of encryption and decryption, as well as key management in secure vaults.

Guaranteed authenticity and integrity	Supports the measures in place to ensure end-to-end data integrity, such that actors can validate the authenticity of the delivered information. This building block links to the key management services.
Monitoring	
Data usage	Supervision of the amount and type of data flowing through the Middleware; and what it is being used for.
Application usage	Supervision of what are the apps being used by any of the end users; and how they are being used.
Infrastructure usage	Supervision of the particular infrastructure resources that are being used.
Usage policies	Integration of usage control capabilities regarding the policies that can be defined in the environment. These policies will describe the terms and conditions under which apps, data or infra can be used on the consumer side.
Energy metrics and alerts	Monitoring supervision will be reflected in a functionality that gathers metrics and gives alerts regarding the general state of the SMP in terms of data, app and infra usage in order to optimize energy usage and meet sustainability goals.
QoS metrics and alerts	Monitoring supervision will be reflected in a functionality that gathers metrics and gives alerts regarding the general state of the SMP in terms of data, app and infra usage, providing information about the quality of service that is being given.
Performance monitoring & enactment	As a previous stage to the performance reporting building block, usage and service availability performance is monitored in a real-time basis to feed the information sets needed by the reporting block.
Reporting	
Performance	Allows the analysis of the performance of each of the functionalities to serve as a source for the general report building block. Along with the monitoring block, provides past performance and history records of the metrics gathered at the monitoring stage.
Platform usage	Creates reports and supervises the use of the global SMP platform in both the infrastructure and data layer by the consumers and providers.
Energy efficiency & sustainability	To support the objective of the Green Deal, this component will provide insights into the energy efficiency and environmental performance of the SMP's building blocks to ensure that the services operate in a low power mode.
Log info extraction	Provides the information needed obtained from sources such as the audit logging helping to generate a comprehensive report of the functionalities.
Exporting	Exporting building block allows information gathered by the reporting processes to be external and available outside of the SMP environment.
Audit	
Audit logging	The administrator can investigate the activities taking part in each of the layers of the SMP to further develop a reporting about issues or non-compliant conditions of the services at data or infrastructure stages. This building block gathers the information of auditing actions performed in the SMP context, allowing to undertake organizational or legal measures that ensure the compliant functioning of the SMP.
Network	

VPN	Virtual private networks may be created in order to protect the data traffic of the middleware from external threats. The VPNs are also to be used to transfer data or applications within and across data spaces over the public internet.
Firewall	This building block allows the creation of firewall rules in order to restrict the inbound and outbound traffic of the private networks. It provides an additional protection layer on top of identity and access management.
Federation management	
Federation orchestration	Provides the means to connect resources in a service network. Manages the operation of networks including processes for monitoring, runtime issues, evaluation, etc. to meet the main principles of interoperability and federation. It also configures the system components of the SMP network to smoothly collaborate in a data space.

I.4 Governance layer architecture

Table 4. Detailed description of governance layer building blocks

Support	
Support page	Consumers will have access to a support webpage with collected and useful documentation regarding the SMP with a FAQ format.
Ticketing system	Users will have availability of a ticketing system that logs the issues regarding SMP to be reported to the administration.
Helpdesk	Third party contractors can connect with the consumers and providers in case the issues remain unsolved. The governance board will coordinate these three sub-blocks.
CSIRT	
Incident response	Coordination at the administration layer will give a response to security incidents with proper procedures to restore full SMP functionality as soon as possible.
Threat monitoring	Proactive threat monitoring and follow-up of possible malicious activities affecting the SMP to avoid potential security breaches before they happen.

Annex II Data Governance Act compliance

The Data Governance Act[15] (DGA) was adopted by the European Commission on November 2020. It aims to boost data sharing within the European Union, while still protecting the sovereignty of data owners. The SMP will push the next evolution of data sharing initiatives in Europe. Therefore, it is paramount that the SMP concepts are compatible with the Data Governance Act regulation. This annex maps the definitions of the regulation on the vocabulary and concepts used in the design of the Smart Middleware Platform. The annex only briefly touches on some concepts of the DGA regulation, and how they map to the SMP. A deep analysis of the regulation is out of scope for this report.

The Data Governance Act defines 'data subjects', 'data holders', and 'data users' as the actors involved in a data sharing transaction:

- A 'data subject' defines any natural person that can be identified directly or indirectly through a direct identifier, such as a name, or one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person.
- The 'data holder' refers to a legal person who is not the data subject of the data in question, but holds the right to grant access to or share the data, in accordance with Union or national law.
- The 'data user' is the legal or natural person who has lawful access to certain personal and non-personal data and to use that for commercial or non-commercial purposes.

In the context of SMP, the *End user* becomes the 'data user' of the regulation, and the *Data provider* forms the 'data holder' in the transaction. The *Data provider* holds the data of possibly multiple 'data subjects'. The regulation that applies on 'data subjects', 'data holders', and 'data users' thus applies on *Data providers* and *End users* following the given mapping.

Additionally, the Data Governance Act defines the concept of 'data intermediation services'. These services aim to establish commercial relationships for the purposes of data sharing between an undetermined number of 'data subjects' and 'data holders' on the one hand, and 'data users' on the other hand. 'Data intermediation services' can be technically facilitated by the services that the SMP offers. A published data sharing service in SMP can be a form of a 'data intermediation service', to which the DGA regulation applies.

Finally, the Data Governance Act provides a definition of a 'secure processing environment'. Such environment represents the virtual or physical environment to ensure compliance with Union law. It is important to mention that the SMP shall represent a 'secure processing environment' and as such, the burden of being compliant with the law falls on the SMP and this requirement should always guide the developers.

[15] Proposal for a Regulation of the European Parliament and of the Council on European data governance (Data Governance Act), COM/2020/767 final, available at: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52020PC0767

Annex III Identification and authentication

This annex explores different identification, authentication and authorization (IAA) mechanisms that can be used for the Smart Middleware Platform. These identification systems should provide the required flexibility to deploy the platform in various topologies and use cases. The annex starts by overviewing three possible technologies for IAA and the required system components for these mechanisms. For each mechanism, the information flow during the on-boarding of a new data space participant is shown. Additionally, as an illustrative example, the authentication process is detailed when an End user accesses data from a Data provider.

III.1 Identification in the SMP ecosystem

Before diving into the technologies for identification, authentication and authorization, this section sketches the context of the identification mechanisms within SMP. The identification system will play an important role in two functionalities of the SMP: (i) to establish secure communication channels, and (ii) to provide the information on which participants can base themselves to define access control policies.

To keep the identification system manageable in a large scale environment, the identification is split into two tiers. The first tier manages the identification, authentication and authorization of organisation employees to use the *SMP Agent* of their organisation. The second tier identifies and authenticates the organisation as a whole in the SMP network. In the first tier, the *SMP Agent* connects to the preferred IAA system of the organisation: EU Login, eID, Microsoft AD, OpenID Connect, … This mechanism is already well established and not unique to the SMP. The rest of this chapter focusses on the second tier, which involves the machine to machine authentication and identification of an organisation in the ecosystem. Figure 11 depicts this two-tier approach.

Each organisation will hold an *Identity file* to support the identification, authentication and authorization of the organisation in the network. Recalling the two functionalities that the IAA supports, the necessary content of this *Identity file* becomes apparent. For the establishment of a secure communication channel between participants (i), the *Identity file* should contain a proof of the organisation's public key. Each data space participant will create a cryptographic public/private keypair that is used in the asynchronous authentication mechanisms needed to establish a communication channel. An example of how such a secure communication channel can be established is the well-known TLS/SSL protocol. The *Identity file* associates the public key of an organisation to its identity. Proving the identity of the organisation then becomes proving the possession of the private key that belongs to the respective public key. This way, the organisation can be authenticated in the network and a secure communication channel can be established.

Access control and authorization by providers (ii) can be performed based on custom identity attributes of an organisation. Examples of such attributes are the organisation name, geographical location, whether it is a private or public institution, … Based on these attributes, providers can define access control policies for their resources. For example, a provider can open a resource to all public institutions, or to all participants from a specific Member State. On the other hand, the access control policies can be more stringent and access is only allowed for a specific organisation. The *Identity file* proves the attributes of an organisation, and, as such, ensures the trust on which a provider can rely to enforce their access control. The following sections deep dive into the technologies that can be used to prove an organisation's identity attributes and their public key.

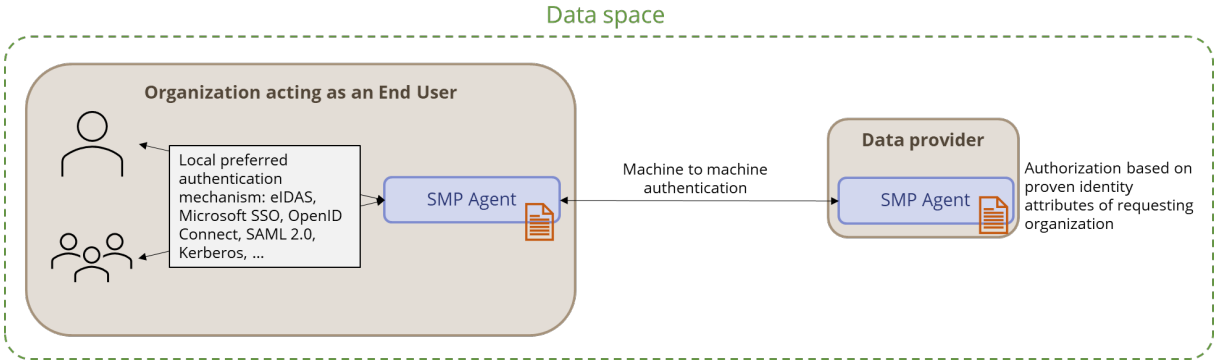

Figure 11. Visual illustration of the two-tier IAA

III.2 Overview of IAA solutions

Through desk research on identification systems and research on the IAA of existing data space initiatives, three possible solutions were identified. The first option is based on X.509 certificates[16] with embedded identity attributes. The second option removes the attributes out of the X.509 certificate and uses a dynamic attribute provisioning service. The third option decentralises the identification mechanism as much as possible and uses novel Self-Sovereign Identities with a distributed ledger.

The next sections explore the three options' technical details and processes. First, the overall system architecture that is needed to support the IAA option is described. Second, the information and process flow of the on-boarding of a new data space participant is detailed. Third, as an illustration to the identification, authentication, and authorization in practice, the example of an End user requesting a data resource from a Data provider is shown. Last, it is shown how the IAA system allows to federate trust across several data spaces or within a single data space.

Note that this analysis makes no assumptions on how a data space is constructed. In order to not limit to a specific data space topology, the analysis considers that a data space itself can be a federation of sub-data spaces. Proving how such a topology is supported, demonstrates the flexibility of the proposed IAA system to support a wide range of data space topologies.

III.2.1 X.509 certificates with embedded identity attributes

X.509 is a widely used standard for defining the format of public key certificates. It defines the information fields that are needed to link a public key to a specific endpoint. The validity of X.509 certificates is attested by a digital signature from a trusted certificate authority. In the SMP, that certificate authority will be the *Identity authority* that issues the certificate as part of the identity provisioning. The custom identity attributes can be embedded in the X.509 certificate through the optional extensions field (available as of version 3 of the X.509 standard).

The overall architecture of a data space is identical to Figure 10 described in Section 6.3. The *Identity authority* is in charge of validating the identity attributes of a data space participant – through secondary channels – and signs the X.509 certificate. The other participants of the data space trust the *Identity authority* and therefore trust the information that is attested by it. This identification model is followed by X-Road, where the *Identity authority* is called the certification authority[17].

On-boarding process

The information flows when an End user wants to on-board in a data space using embedded identity attributes is shown in Figure 12 below. The on-boarding process involves four steps. Note that the End user is only an example of an actor that wants to on-board, any other type of actor undergoes the same process.

1. The End user generates a public/private keypair and safely stores his private key (SK) in a secure digital key vault.
2. The End users submits the on-boarding documents to the *Identity authority* along with its public key. The content of these on-boarding documents is defined by the data space, based on what the data space governance requires from its participants. The required identity attributes will be given through the on-boarding documents.
3. The *Identity authority* is required to validate the submitted information. This can happen through several possible secondary channels. Some examples can be a phone call, secure email, a physical meeting with organisation representatives, etc. It is important to confirm the public key with the data space representative.
4. If the submitted information is correct, the *Identity authority* generates the X.509 certificate with the embedded attributes and public key, by a digital signature.

[16] More information can be found at https://datatracker.ietf.org/doc/html/rfc3280attested
[17] More information can be found at https://x-road.global/architecture and on the 'Preliminary analysis on existing solutions and public sector users report (D03.01)' document

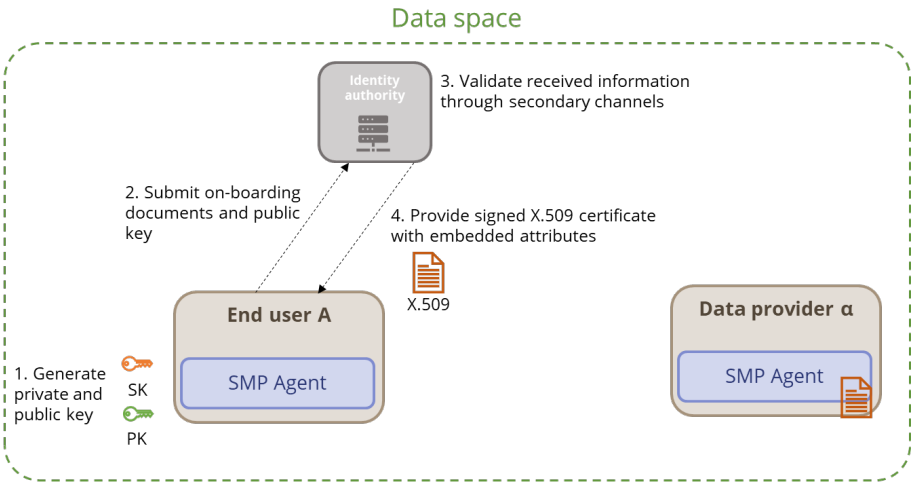

Figure 12. On-boarding process in the first option for SMP IAA

End user requesting access for data resource

After two participants are on-boarded (in this example an End user and a Data provider), they can authenticate each other and start sharing resources. The steps of the identification, authentication and authorization are described as follows (Figure 13):

1. The End user initiates the communication to the Data provider. A secure communication channel is established using the TLS mutual authentication protocol. This protocol uses the X.509 certificates and asymmetric public key cryptography to establish a symmetric encryption key that can be used for future communication.

2. Part of the TLS authentication is the validation of the X.509 certificates by both participants. The need for this step is illustrated by explaining the certificate revocation mechanism of this IAA system. When the data space governance wants to retract the permission of a participant to partake in the data space, it cannot take away the certificate of that participant. The X.509 certificate is stored in the local infrastructure of the participant and is not controlled by the data space governance. The X.509 certificates have an expiration date, but cannot be invalidated offline before the expiration date of the original issuance. Therefore, an online protocol is set up, by which the *Identity authority* can mark a previously issued certificate as no longer valid, even if it has not yet expired. Consequently, data space participants are required to check the status of the counterparty's certificate before trust can be assured. Checking the certificate status happens over the Online Certificate Status Protocol or OCSP. Thus, this step of the information flow consists of this status check with the *Identity authority* to make sure the other party's certificate is not revoked.

3. After the communication channel is established, the Data provider checks if its access control policy allows access by this particular End user to the requested data set. The embedded identity attributes are run against the defined policies, and access is granted or denied accordingly.

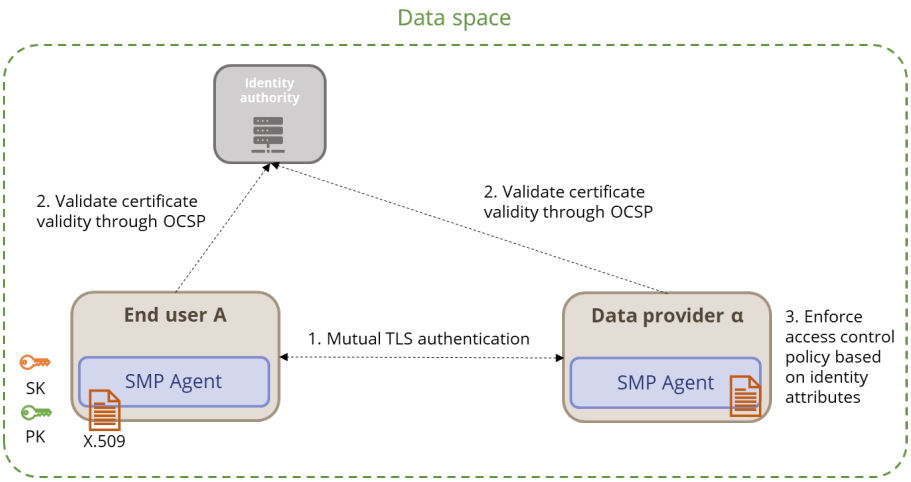

Figure 13. End user requesting access in the first option for SMP IAA

Federation within and across data spaces

The last topic to explore is how the X.509 certificates with embedded identity attributes support the federation of trust within and across data spaces. To create trust within a data space, a tree of trusted *Identity authorities* is created. Figure 14 visualizes such a trust tree. The root *Identity authority* attests the identity of sub *Identity authorities*. This trust can be handed down as deep as needed for the data space. When a data space participant wants to validate the identity of another participant, it verifies not only the certificate of the participant, but also the chain of certificates that were used to attest the trust of the issuing *Identity authority*. A single *Identity authority* within the data space forms the root of the trust tree. This root *Identity authority* is known and trusted by the data space participants. Such a model of a trusted tree of *Identity authorities* resembles the model used for attesting SSL/TLS certificates in the public Internet.

Federating trust and identities across data spaces becomes more complex. One could conceive the option of putting another *Identity authority* above the root *Identity authorities* of the different data spaces. However, for the EU common data spaces, this model is not the most suitable. Such a model creates a governing authority above all the EU data spaces, which is an undesired and cumbersome role. For the EU common data spaces it is important that each of them can be governed independently, without an overarching governance across data spaces. The highest level of the trust tree must therefore be within the data space itself. The interoperability between data spaces should instead be created through establishing peering relationships between data spaces. This governance model can be supported in the identification system by allowing multiple trusted root *Identity authorities* to be configured for the participants. The root *Identity authority* of a data space manages a list of trusted authorities. This list contains the root *Identity authorities* of the other data spaces with which the data space wants to be interoperable. A participant of data space A then can trust the identity of a participant of data space B, because the root *Identity authority* of data space B is trusted within data space A.

Note that the established trust between data spaces through this mechanism is not transitive or symmetric by default. It is not symmetric as data space A can configure to trust the root *Identity authority* of data space B, without data space B needing to do the same for the root *Identity authority* of data space A. The relation is neither transitive, because trust between data space A and data space B, and trust between data space B and data space C, does not entail trust between data space A and data space C. The lack of these properties allows more flexible trust networks to be created, with more independence for each data space governance.

There is a limitation in this mechanism of trust across data spaces, or even within data spaces. Because the identity attributes are embedded in the X.509 certificates, the same X.509 certificate can only be reused for identification in other data spaces if the other data space only requires the same identity attributes for authorization. If the other data space requires identity attributes that are not present in a participant's current X.509 certificate, a new on-boarding will still be needed under the *Identity authority* of the other data space.

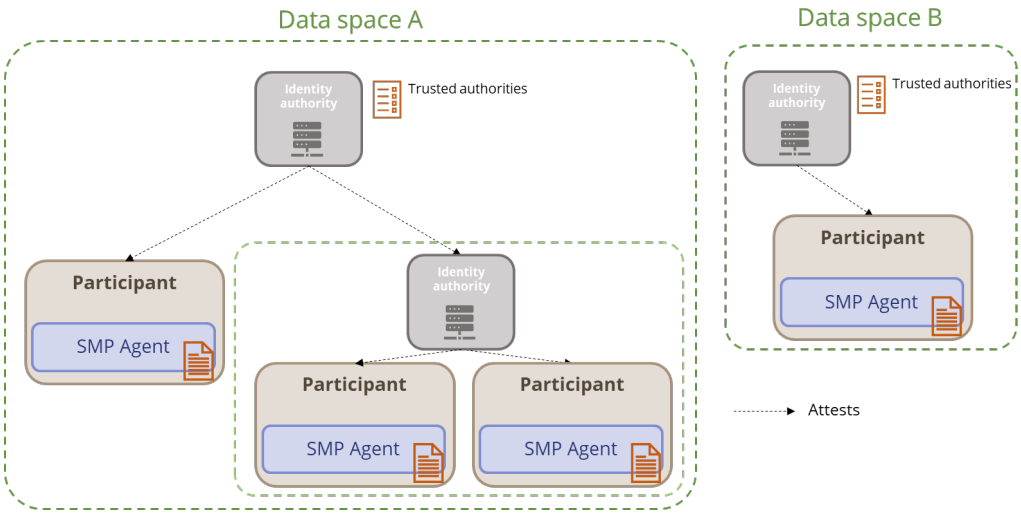

Figure 14. Federation of trust in the first option of SMP IAA

III.2.2 X.509 certificates with dynamic attribute provisioning

The second mechanism that can be used for the identification, authentication and authorization in SMP, separates the X.509 certificates from the identity attributes. Rather than embedding the attributes, a separate entity, the *Attribute provider*, manages the identity attributes of the data space participants. The X.509 certificate only attests the public key of the data space participant. The identity attributes that are needed for the data space authorization are stored and attested by the *Attribute provider*. This system for IAA is recommended by the IDSA reference architecture[18], where the *Attribute provider* is called a Dynamic Attribute Provisioning Service.

The *Identity authority* acts largely the same as in the previously described system. It is still the *Identity authority* that is in charge of verifying the on-boarding information prior to issuing the X.509 certificate. Instead of embedding the identity attributes in the certificate, the *Identity authority* registers the verified attributes at the *Attribute provider*. These give ephemeral proofs to data space participants of their identity attributes. Participants can be authorized to the services from other data space participants by providing the ephemeral proofs. Important is that these proofs remain valid only for a short time, such that no OCSP on them is required. A typical value is 60 minutes, however the exact value has to be configured by the *Attribute provider* based on the needs of the data space.

On-boarding process

The information flows when an End user wants to on-board in a data space is shown in Figure 15 below. The on-boarding process involves five steps. Note that the End user is only an example of an actor that wants to on-board, any other type of actor undergoes the same process.

1. The End user generates a public/private keypair and safely stores his private key (SK) in a secure digital key vault.
2. The End users submits the on-boarding documents to the *Identity authority* along with its public key. The content of these on-boarding documents depends on the data space and what the data space governance requires from its participants. The required identity attributes will be given through the on-boarding documents.
3. The *Identity authority* is required to validate the submitted information. This can happen through several possible secondary channels. Some examples can be a phone call, secure email, a physical meeting with organisation representatives, etc. It is important to confirm the public key with the data space representative.
4. If the provided information is correct, the *Identity authority* registers the identity attributes at the *Attribute provider*. The *Attribute provider* stores the identity attributes and to which public key they belong. Based on the public key, the attributes can later be searched and located.
5. If the registration of the identity attributes is successful, the End user is provided with a signed X.509 certificate that attests its public key. The X.509 certificate only contains the essential information to attest the public key, and does not contain custom extensions with the identity attributes.

[18] International Data Spaces Association, https://internationaldataspaces.org/wp-content/uploads/IDS-Reference-Architecture-Model-3.0-2019.pdf, "*Reference Architecture Model version 3.0*", April 2019

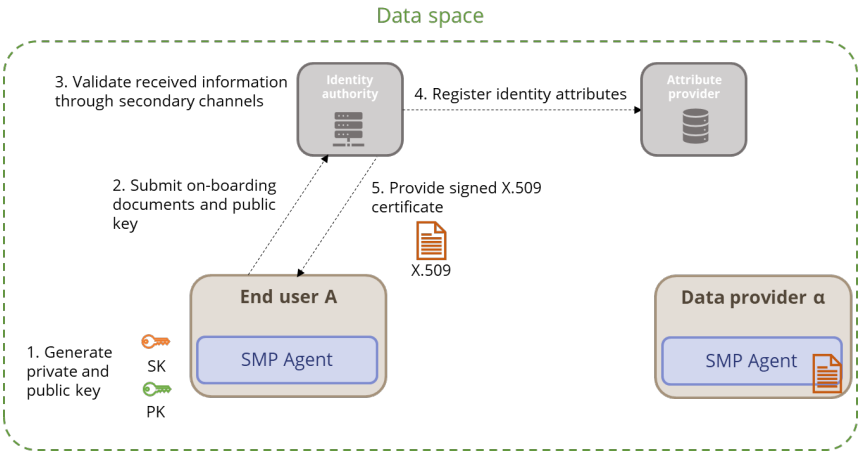

Figure 15. On-boarding process in the second option for SMP IAA

End user requesting access for data resource

After two participants are on-boarded (in this example an End user and a Data provider), they can authenticate each other and start sharing resources. The steps of the identification, authentication and authorization are described as follows (Figure 16):

1. Assuming the End user does not have a valid ephemeral proof of its identity attributes, the process starts with the End user requesting such a proof at the *Attribute provider*.
2. The *Attribute provider* verifies whether the requesting party possesses a valid X.509 certificate, through a mutual TLS procedure, and looks up the identity attributes belonging to that requestor's public key. Note that part of validating the X.509 certificate happens again through OCSP with the *Identity authority*, but this is not shown to keep the figure clear.
3. The *Attribute provider* generates an ephemeral proof of the identity attributes. This ephemeral proof also contains the public key to which the identity attributes belong, as well as a validity period of the proof. The *Attribute provider* digitally signs the ephemeral proof.
4. When the End user has a valid ephemeral proof of its identity attributes, it initiates a mutually authenticated TLS connection with the Data provider. Part of this mutual authentication is again the certificate check through OCSP. Note that this is not shown to maintain the figure's clarity.
5. The End user provides the ephemeral proof to the Data provider. The Data provider checks the signature of the ephemeral proof and validates whether the public key corresponds to the public key that was used in the mutual TLS authentication.
6. After checking the validity of the ephemeral proof, the access control policies of the Data provider are enforced based on the identity attributes that the ephemeral proof attests.

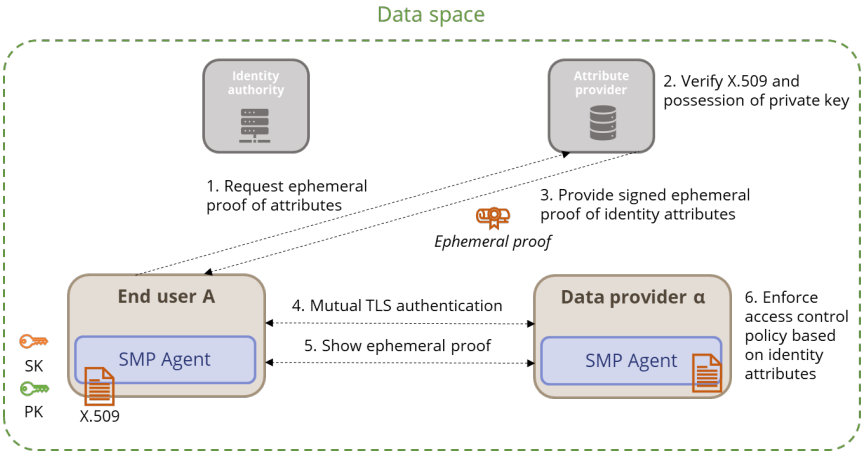

Figure 16. End user requesting access in the second option for SMP IAA

Federation within and across data spaces

Federating identities and trust within and across data spaces in this IAA mechanism largely follows the same principles as described in the first option. In this option (Figure 17), a tree of trust is also built, with its root in the root *Identity authority* of the data space. New is the addition of the *Attribute providers* as leaf nodes in the tree. They are trusted through attestation by an *Identity authority*, but cannot extend this trust to other nodes of the network. They are part of the trust tree, such that their signatures can be validated by data space participants by working up the chain of trust, similar as to how trust is validated for other data space participants.

Federating trust across data spaces happens over the same peering mechanism as described in the previous option. A list of trusted *Identity authorities* is managed by the root *Identity authority* of a data space. This list is distributed to the data space participants and *Attribute providers* within a data space. With this mechanism, peering data spaces can trust the X.509 certificates of participants of other data spaces in the TLS authentication.

The main difference for the federation of identities across and within data spaces is that the identity attributes are no longer per se attested by the *Identity authority* close to the End user, but instead can be attested by the *Attribute provider* close to Data, Infrastructure or Application provider. In the previous option, the identity attributes have to be attested by the *Identity authority* that issues the X.509 certificate. In this option, this is no longer fixed. Data space participants can actually have identity attributes attested by several *Attribute providers*. The need for this can arise when a particular data space or sub data space needs different attributes for their authorization. For example, it is imaginable that the Manufacturing Data Space will use different attributes for their access control polices than the Language Data Space. In such a setting, interoperability can still be guaranteed when a participant of one data space simply on-boards in several other data spaces or sub data spaces. A second on-boarding will however be more lightweight, as the participant already possesses a valid X.509 certificate. Only the new identity attributes have to be validated and registered in the appropriate *Attribute provider*. A practical example clarifies this discussion.

Assume an End user of the Manufacturing Data Space wants to obtain data from the Energy Data Space and the Language Data Space. Second, assume that there is a peering trust between these data spaces through the list of trusted identity authorities. Third, assume that the Manufacturing Data Space and Energy Data Space control access to resources based on the same identity attributes; however, the Language Data Space requires different identity attributes of an organisation that wants to obtain resources. The following mechanisms allow the End user to obtain the required data:

- The End user holds an X.509 certificate that has been provided by the root *Identity authority* of the Manufacturing Data Space. Therefore, its identity attributes are also registered in the *Attribute provider* corresponding to the root *Identity authority* of the Manufacturing Data Space.
- The End user obtains access to the data from the Energy Data Space, by obtaining an ephemeral proof of its identity attributes from the *Attribute provider* of the Manufacturing Data Space. Since the Energy Data Space requires no additional identity attributes, and the attributes attested by this *Attribute provider* are trusted through the peered trust, access can be granted based on this ephemeral proof. In this case, a Data provider of the Energy Data Space thus not only trusts the X.509 certificate of another data space, but also the ephemeral proof of another data space. All this happens because of the peered trust between the Energy Data Space and the Manufacturing Data Space.
- The End user can not as easily obtain data from the Language Data Space. Since different identity attributes are used in the access control policies of the Language Data Space, the End user must first provide these identity attributes. To enable this, the End user performs a lightweight on-boarding in the Language Data Space. The End user provides its X.509 certificate, ephemeral proof of overlapping identity attributes, and the new identity attributes to the *Identity authority* of the Language Data Space. The *Identity authority* can trust the X.509 certificate and the ephemeral proof, and thus only needs to manually verify the new identity attributes. When verified, the identity attributes are registered at the *Attribute provider* of the Language Data Space. When the End user now wants to request data from the Language Data Space, it uses its X.509 certificate of the Manufacturing Data Space to request an ephemeral proof of identity attributes from the *Attribute provider* of the Language Data Space. This ephemeral proof is then given to a Data provider of the Language Data Space, who then grants access to the resources.

Note that the decision to separate the X.509 certificates and the identity attributes creates much more flexibility to federate identities and their attributes. In the first IAA mechanism, different identity attributes would entail the End user needing to fully on-board in a new data space, and needing to manage multiple X.509 certificates.

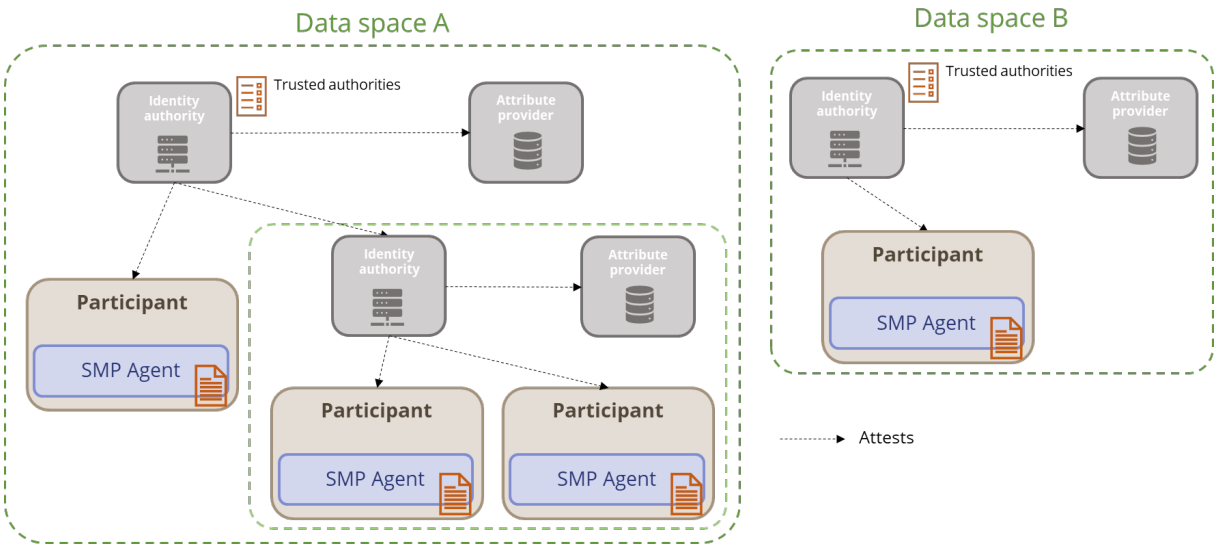

Figure 17. Federation of trust in the second option of SMP IAA

III.2.3 Self-Sovereign Identities with a distributed ledger

The most novel paradigm in identity and access management is the emergence of Self-Sovereign Identities (SSI). The core idea of SSI is to move control over identity data back to the natural or legal persons. The paradigm shares similarities with the previous option when it comes to the trust in the *Identity authorities*, but differs drastically when it comes to the data formats and storage of credentials. An SSI solution uses distributed ledger technology to indisputably attest the issued credentials. It allows the decentralisation of the *Identity authorities* and *Attribute providers*. Later explanation on the specific information flows clarifies how this is achieved.

The data models used in SSI are defined by the World Wide Web Consortium (W3C)[19]. This report does not provide the details on these data models, but briefly touches on what these data models serve in the SSI framework.

- W3C Distributed Identifiers (DID)[20] are used to uniquely represent any natural or legal persons[21]. In the context of SMP, the DID will represent the organisations that participate in a data space. A DID on its own only serves as an anonymous pseudonym for the organisation, therefore a link is made to the real-life identity behind the DID by a trusted authority. This link is proven through public key cryptography by linking the DID to a public/private keypair managed by the organisation. Proof of possession of the private key then proves that the DID belongs to that particular organisation.
- W3C Verifiable Credentials (VC)[22] represent the verifiable identity attributes that belong to a DID. The VCs are granted to the organisation behind a DID by the *Identity authority*. A VC acts similar to the ephemeral proof that was discussed earlier. If attested by a trusted *Identity authority*, the validity of the identity attributes can be verified offline by checking the digital signature of the VC.
- W3C Verifiable Presentations (VP)[23] add the additional functionality to morph Verifiable Credentials into a representation that shows only what another party needs to know. The underlying cryptographic protocols[24] allow to create a VP without any new involvement of the *Identity authority*. In a practical setting, this mechanism can be used to combine attested identity attributes from multiple *Identity authorities* into a single document that can be presented to another participant of a data space. It can also limit the information that is given to another party, providing only the required identity attributes to gain access, and nothing more.

[19] More information can be found at https://www.w3.org/
[20] More information can be found at https://www.w3.org/TR/did-core/
[21] Note that DIDs can actually be used in a much broader context to identify 'things', like individual computers, data sets, services, … In this study, this abstraction is not made, and DIDs will be used for the identification of legal persons.
[22] More information can be found at https://www.w3.org/TR/vc-data-model/
[23] idem
[24] These protocols rely on Zero-Knowledge Proofs, which are mechanisms that allow a party to prove a statement without revealing any more information apart from the fact that the statement is indeed true

In contrast to the two previously described options for SMP IAA, the SSI framework should not be considered in an SMP isolated use case. The previous two options aimed to provide an identification solution specific to participants of data spaces built on SMP, while dealing with their specific business needs. The SSI paradigm on the other hand, has the potential to become a much larger ecosystem for identification, authentication and authorization for many IT systems across Europe. The previous options required the creation and management of *Identity authorities* specific to a data space, managing the identity attributes that participants of the data space need. These attributes are however likely documented and attested by other legal authorities within the Member State of the participant.

The full potential of SSI comes into play when the ecosystem around SSI grows far beyond SMP. If these existing legal authorities become part of an European SSI ecosystem, the data space will no longer need to set up a specific *Identity authority*. Participants can obtain several VC's from several suitable legal authorities, each VC proving a subset of identity attributes for the organisation. These VC's can be combined into a VP, that is presented to other data space participants. Because of their trust in the legal authorities of Member States, they can establish trust in the verifiable identity attributes of the data space participant.

Self-sovereign Identity is emerging[25] as a powerful contender for future digital identity infrastructure. Efforts made by major industry players (e.g., Microsoft[26]), smaller enterprises (e.g., Evernym[27]), non-profit organisations (e.g., Sovrin[28]), and public sector (e.g. the European Union creating an eIDAS compatible European Self-Sovereign Identity framework on the EBSI[29]) indicate the momentum towards SSI that should not be neglected by SMP. Additionally, Gaia-X is also moving in the direction of SSI for the identification of the Gaia-X Federation Services[30]. Therefore, compatibility with Gaia-X might depend on the adoption of SSI principles in SMP.

Because of the still emerging efforts by both industry and public sector on SSI solutions, the rest of this chapter assumes that the SMP cannot tap into an existing SSI ecosystem. The information flows are explained from the perspective that the *Identity authority* role is fulfilled by an assigned party in a data space. In the rest of this chapter, the *Identity authority* has no purpose outside of the SMP context.

On-boarding process

The on-boarding process of an End user in the SSI participant can be divided into two parts: obtaining an attested DID and registering the identity attributes. Both sub-processes can happen independently and can later be replaced independently by a similar process in a wider SSI ecosystem. The on-boarding process happens as follows, with the first sub-process being represented in step 1-5 (Figure 18) and the second being represented in step 6-9 (Figure 19):

1. The End user generates a new private/public keypair and stores the private key (SK) in a secure digital key vault.
2. The End user creates a DID that contains the public key, the unique identifier string, and a link to the participants real-life identity.
3. The End user requests the *Identity authority* to attest his DID. The *Identity authority* verifies the real-life identity of the End user.
4. The *Identity authority* attests the DID by digitally signing the it.
5. The End user records its attested DID on the distributed ledger, by which the DID becomes discoverable for other participants.
6. After obtaining a valid DID, the End user registers its identity attributes at one or more *Identity authorities.* This can be a different authority as the one that previously attested the DID. As described above, in a full SSI ecosystem this step might no longer be necessary as the identity attributes can be attested by authorities outside the data space. This explanation limits, however, to the vision of an SSI framework only in SMP.

25 Deloitte, https://www2.deloitte.com/global/en/pages/risk/articles/solving-the-public-sector-identity-crisis.html, "*Solving the public sector identity crisis*", 2021
[26] More information can be found at https://www.microsoft.com/en-us/security/business/identity-access-management/decentralized-identity-blockchain
[27] More information can be found at https://www.evernym.com/about-evernym/
[28] More information can be found at https://sovrin.org/
[29] More information can be found at https://ec.europa.eu/digital-building-blocks/wikis/display/ebsi
[30] More information can be found at https://www.gxfs.eu/authentication-authorisation/

7. The *Identity authority* retrieves the DID from the distributed ledger to validate the identity of the End user.
8. The received identity attributes are validated through secondary channels, like phone, email, or a physical meeting.
9. If the information proves correct, the attributes are stored by the *Identity authority* with a link to the DID they belong to. Note that a data space participant can choose to register its attributes with different *Identity authorities* to disperse its information across multiple entities.

An additional remark must be made on the attestation of the DID. This information flow regarded the need for an *Identity authority* within the data space to attest the DID. However, an interesting synergy arises when considering the eIDAS initiative in Europe. eIDAS aims to harmonise the digital identities of natural and legal persons. In practice, this entails that natural and legal persons are given private and public keypair by their legal authorities that are verifiable across Member States[31]. This eIDAS keypair of a legal person can be used to attest the DID of the SMP. This means that the data space participant can attest their own DID, by signing it with the trusted keypair of eIDAS. The trust that is set up through eIDAS thus transfers to the DIDs of an SMP data space. This reduces the complexity of on-boarding a new participant, as no *Identity authority* is involved in the first sub-process.

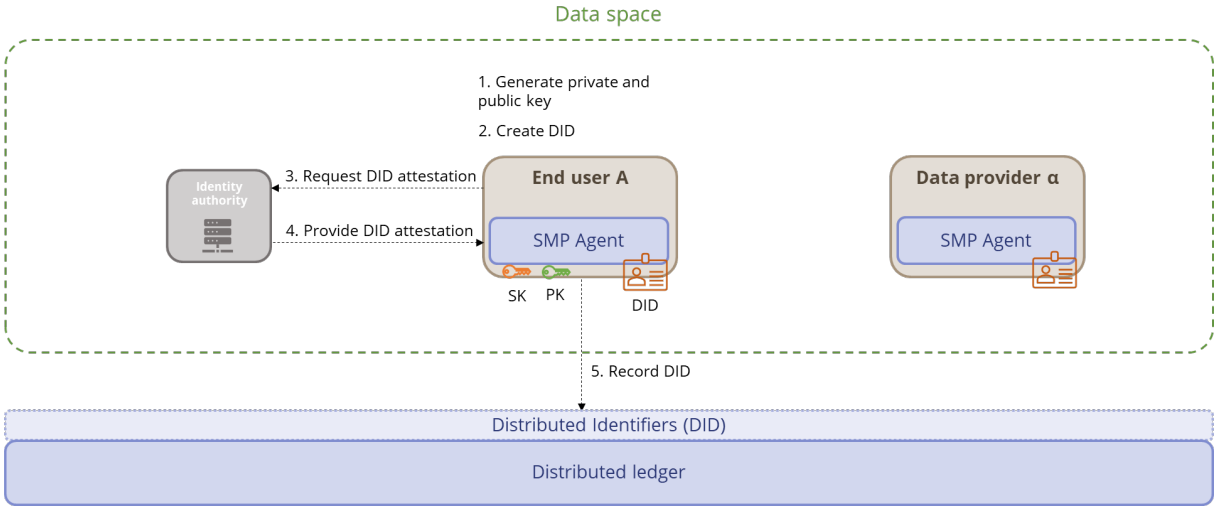

Figure 18. On-boarding process in the third option for SMP IAA - Step 1 to 5

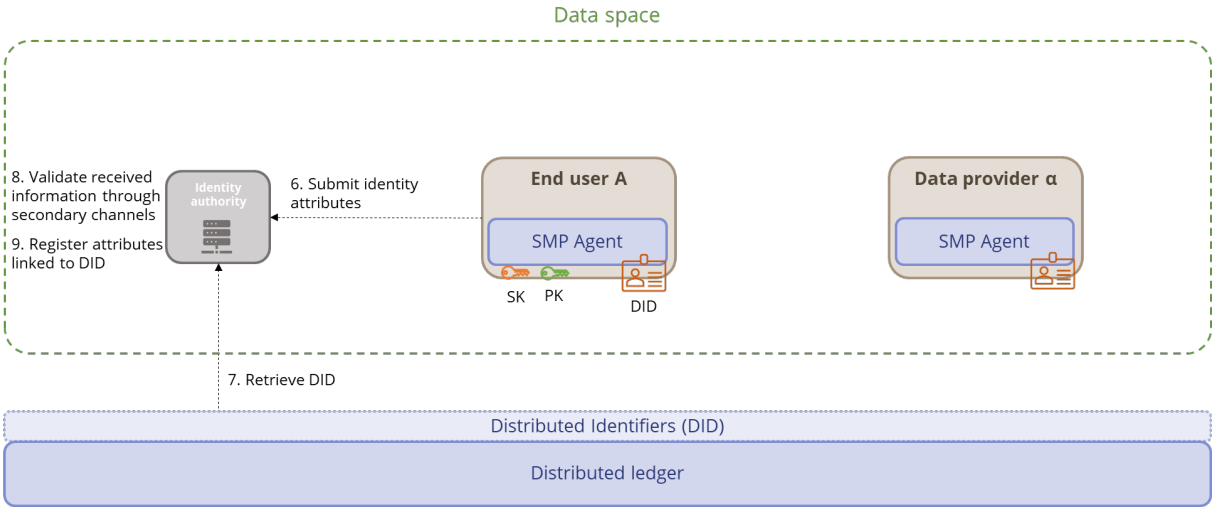

Figure 19. On-boarding process in the third option for SMP IAA - Step 6 to 9

[31] Note that the eIDAS Regulation is not yet implemented in an operational system for the identification of legal persons.

End user requesting access for data resource

After obtaining a valid and trusted DID and having its identity attributes registered, a data space participant can interact with other participants of the data space. This information flow details how an End user can request access to a data service of a Data provider, and how the data provider authenticates the End user (Figure 20):

1. The End user requests a Verifiable Credential from one or more *Identity authorities* that can attest certain identity attributes that the End user needs proof of.
2. The *Identity authority* retrieves the DID from the distributed ledger to assure the identity of the End user.
3. The *Identity authority* looks up the identity attributes belonging to the DID and creates the Verifiable Credential on the identity attributes.
4. The *Identity authority* provides the Verifiable Credential to the End user and records a proof of issuance on the distributed ledger. This proof of issuance is recorded such that other participants can validate whether the VC is still valid. It is through the distributed ledger as well that *Identity authorities* can later revoke credentials, which would also be visible to the whole network.
5. The End user morphs one or more Verifiable Credentials into a Verifiable Presentation that only reveals the information that the Data provider needs to enforce its access control policy.
6. A mutual TLS-authenticated channel is established between the End user and Data provider.[32]
7. Part of the TLS authentication involves retrieving the DID of both the End user and Data provider and verifying the validity.
8. With the secure communication channel established, the End user presents the Verifiable Presentation to the Data provider.
9. The Data provider reads the information about the issuance of the underlying Verifiable Credentials to make sure none are revoked.
10. In the last step, the Data provider enforces the access control policy based on the proven identity attributes of the End user.

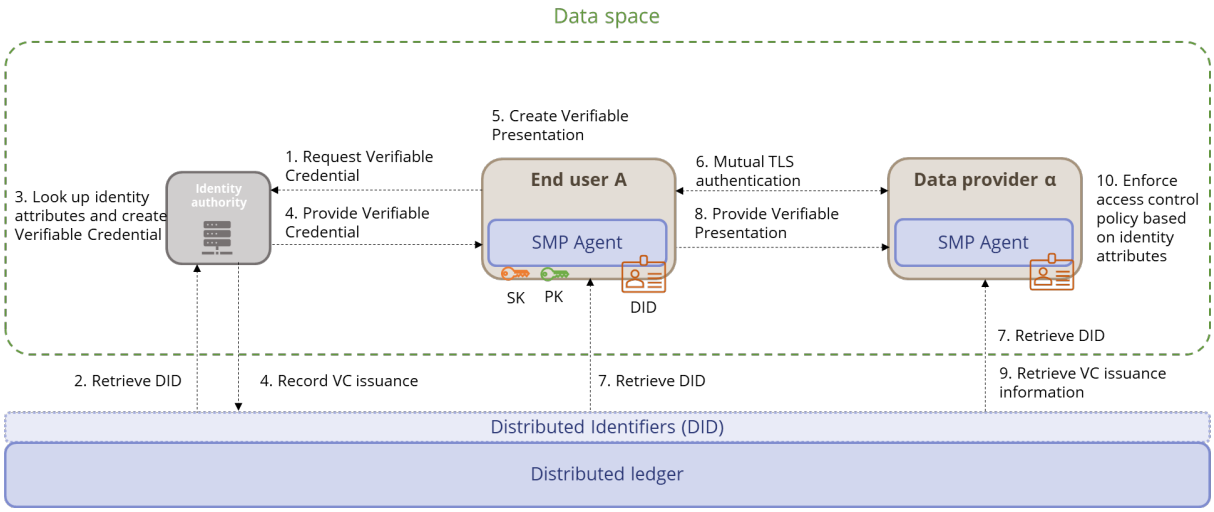

Figure 20. End user requesting access in the third option for SMP IAA

Federation within and across data spaces

[32] Note that TLS implementation based on Distributed Identifiers are uncommon in industry today. Common TLS implementation work using the well-established X.509 certificates. However, this is only a practical issue as there is no inherent incompatibility between TLS and DID for public key certification. The mutual authentication algorithm of TLS is a variant of the Station-to-Station protocol, which only requires verified public keys for both parties of the communication. Whether these public keys are verified through an X.509 certificate or through a DID is practical consideration that can be overcome by amending existing TLS implementations.

Federation of trust with SSI has already been discussed in previous sections. The SSI framework allows for a very flexible distribution of trust within an ecosystem. Note that the distributed ledger in the previous figures has distinctively been placed outside of the data space. This distributed ledger is common across data spaces, or an even wider ecosystem. This entails that the DIDs of participants can be trusted across data spaces or sub data spaces.

It has already been pointed out several times that multiple *Identity authorities* can coexist within a data space. Each *Identity authority* might specialise in verifying a specific identity attribute. Trust in these *Identity authorities* is again managed by a root governance of the data space through a list of trusted *Identity authorities*. In a wider SSI ecosystem, trust in the *Identity authorities* can be managed by other parties, depending on how the ecosystem is configured. *Identity authorities* can also be part of multiple data spaces, attesting to the attributes that multiple data spaces need.

Overall, the SSI frameworks offers a large flexibility to federate trust and identities across and within data spaces, or in an even wider ecosystem. The developments of these wider ecosystems need to be monitored, and connecting the SMP to these ecosystems can provide valuable synergies and offload certain responsibilities away from the data space governance.

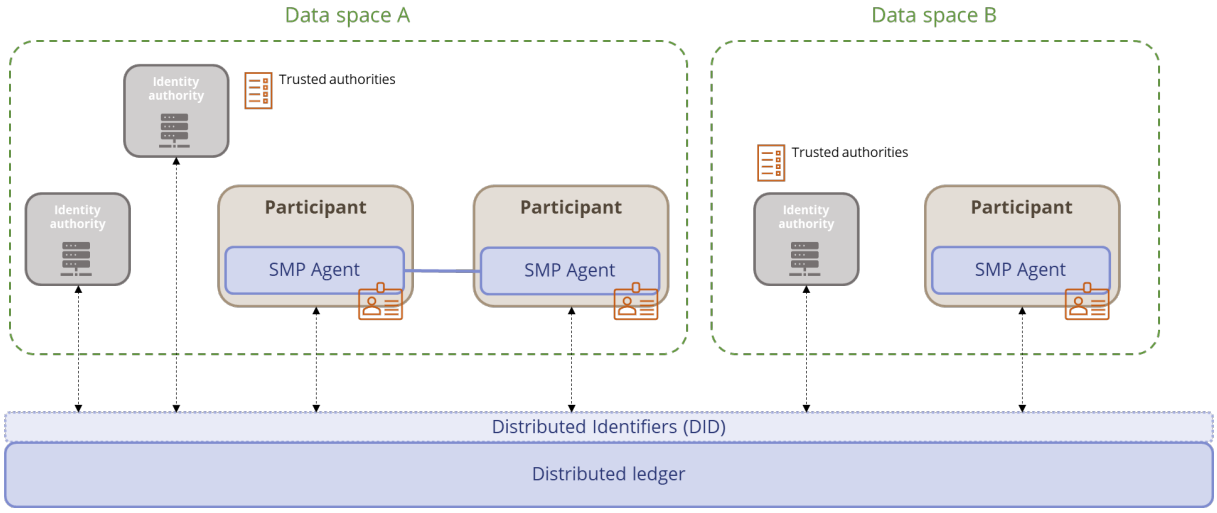

Figure 21. Federation of trust in the third option of SMP IAA

III.3 Comparison of IAA solutions

The last part of the analysis on identification, authentication and authorization compares the three proposed options for SMP. Each option presents several benefits and drawbacks that are debated in this section. A final topic is discussed on how the IAA system can evolve within the SMP.

The first option – the X.509 certificates with embedded identity attributes – benefits from the least complexity. It is a well-established paradigm in industry, with many example implementations to be found. It also results in the least amount of operational effort for the *Identity authority*, as this component does not need to do the complex operations that are involved in creating ephemeral proofs of identity attributes. Because the X.509 certificates are issued for a long time period, the overall load on the *Identity authority* comes predominantly from the OSCP messages. The big drawback is the lack of flexibility of this system. If the identity attributes change, a whole new X.509 certificate has to be issued, and the old one has to be revoked. Additionally, if different identity attributes are needed in an other data space, a whole new certificate has to be issued and more than one certificate has to be managed by the participant.

The second IAA option – the X.509 certificates with dynamic identity attribute provisioning – solves the flexibility issues of the first system. The X.509 certificates are detached from the identity attributes. This means changes to the attributes can happen without needing to re-issue the X.509 certificate. Different identity attributes can

also be registered at different *Attribute providers* under the same X.509 certificate. This simplifies the process for organisations to become part of multiple data spaces, needing to manage only a single X.509 certificate. The drawback of this option is the operational effort of the *Attribute providers*. These have a significantly higher load than in the first option, as they continuously need to look up identity attributes and provide ephemeral proofs. A second drawback is the centrally consolidated identity information about organisations. This may become an interesting attack vector for malicious adversaries, and thus needs to be carefully secured.

The third option – the Self-Sovereign Identities – decentralises the trust in the data spaces. It provides the most flexibility for a topology of multiple *Identity authorities*. It also opens the door to a much wider IAA ecosystem, from which data spaces can benefit. Ideally, most identity attributes will be attested by *Identity authorities* outside of the data space ecosystem, which removes the question of who should manage the centralised *Identity authorities*. Moreover, the novel W3C data models provide a desirable privacy-preserving mechanism which allows data space participants full control on which identity attributes they make known to other organisations. The key drawback of SSI, is the fact that it is not yet widespread in industry. Many initiatives are moving in this new direction, but a pervasive adoption has yet to emerge.

An evolution from the second IAA option towards the Self-Sovereign Identities can be considered., when an ecosystem around SSI develops within Europe. The most prominent initiative to monitor in this respect are the European Blockchain Services Infrastructure (EBSI) initiative and other practical development of the eIDAS Regulation. To smooth the transition from the second option towards the third option, it can be investigated if the W3C Verifiable Credentials and W3C Verifiable Presentations can already be used as the ephemeral proofs of the second IAA option. This would mean that transitioning towards a full SSI framework, would mean the replacement of the X.509 certificates by the Distributed Identifiers, but the platform would already be familiar with the other W3C data models.