D3.4 Data Marketplaces with Interoperability Solutions I

Authors: **Dr. Stefan Gindl**

December 2020

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 871481

TRUSTS Trusted Secure Data Sharing Space

D3.4 Data Marketplaces with Interoperability Solutions I

Document Summary Information

Grant Agreement No	871481	Acronym	TRUSTS
Full Title	TRUSTS Trusted Secure Data Sharing Space		
Start Date	01/01/2020	Duration	36 months
Project URL	https://trusts-data.eu/		
Deliverable	D3.4 Data Marketplaces with Interoperability Solutions I		
Work Package	WP3		
Contractual due date	31/12/2020	Actual submission date	15/12/2020
Nature	Report	Dissemination Level	Public
Lead Beneficiary	RSA		
Responsible Author	Stefan Gindl		
Contributions from	SWC, IDSA, DIO, FORTH, EMC		

Revision history (including peer reviewing & quality control)

Version	Issue Date	% Complete[1]	Changes	Contributor(s)
v0.1	27/10/2020	5	Initial Deliverable Structure	Stefan Gindl (RSA)
v0.2	10/11/2020	60	1st Version	Stefan Gindl (RSA)
v0.3	30/11/2020	75	2nd Version	Stefan Gindl (RSA)
v0.4	3/12/2020	85	Review 1	George Margetis (FORTH)
v0.5	6/12/2020	90	3rd Version	Stefan Gindl (RSA)
v0.6	10/12/2020	95	Review 2	Benjamin Heitmann (FhG)
v0.7	11/12/2020	97	Interim Final Version	Stefan Gindl (RSA)
v0.8	14/12/2020	98	Final review	Dominik Kowald (KNOW)
v1.0	14/12/2020	100	Final version	Stefan Gindl (RSA)

Disclaimer

[1] According to TRUSTS Quality Assurance Process:

1. to be declared

Copyright message

Table of Contents

List of Figures

List of Tables

Glossary of terms and abbreviations used

Abbreviation / Term	Description
DIO	Data Intelligence Offensive (www.dataintelligence.at)
DM	Data Market/Datamarket/Data Marketplace
DMA	Data Market Austria (datamarket.at)
IDS	International Data Spaces (www.internationaldataspaces.org)
IDSA	International Data Spaces Association (www.internationaldataspaces.org)
RoD	Registry of Datamarkets

1 Executive Summary

Interoperability with third party data markets (DMs) is an essential step towards making TRUSTS a DM federator. Interoperability provides all technical and methodical solutions that are required for TRUSTS to exchange data with external DMs. Exchanging data encompasses listing data assets available in other DMs, making them accessible via the TRUSTS platform, and trade data assets within TRUSTS.

Interoperability in the context of Task 3.3 "Data marketplaces interoperability solutions" explicitly focuses on the exchange of data with third party DMs. It does not pertain to the interoperability between components and nodes within TRUSTS, e.g. in the context of a micro-services architecture or of a distributed architecture. Federation, i.e. a distributed architecture, is one of the core goals of TRUSTS, however this is not part of Task 3.3.

Task 3.3 provides technical solutions where possible. This is often limited by the features available in third party DMs, i.e. the richness of APIs exposed. However, based on the available amount of data, Task 3.3 will derive guidelines and best practices for interoperability of interested DMs with TRUSTS. Similar to a manual or a reference architecture, the guidelines will provide recommendations, instructions, and clear and concrete guidance on how to implement, deploy, and connect interoperability solutions for their own DM to operate with TRUSTS. Task 3.3 will provide the connectivity points where external DMs can hook their system.

The strategy that was followed within Task 3.3 was to first identify a set of relevant third party DMs using existing listings. Subsequently, the technical specifications of the DMs were examined with regard to their suitability for interoperability. For example, a DM with a rich API catering for a wide variety of consumers' data needs would have been the ideal candidate for Task 3.3. In contrast, a DM where data consumption is exclusively transacted via a Web interface would have been filtered out for further consideration. Mere web interfaces are not machine operable, which consequently means Task 3.3 cannot possibly produce an interoperability solution for it. Thus, Task 3.3 provides an interoperability solution for two relevant DMs with an existing API and furthermore defines guidelines for DMs willing to connect to TRUSTS in the future. Most DMs are at an intermediate level, i.e. they provide a form of machine operable interface, but not necessarily to an extent desired for TRUSTS. An indicative example regarding DMs providing facilities to download data but do not give access to their underlying data catalog. This limits the extent of possible interoperability, since full access to the underlying data catalog would provide the means to give overview of existing datasets in the DM. The exploration of the technical features was accomplished on four DMs (Namara[2], Otonomo[3], HERE[4], CARTO[5]). This involved the creation of small software prototypes connecting to the APIs exposed by the DMs. The data returned was investigated to draw conclusions about the features delivered by the DMs.

The software prototypes, together with the study of the IDS Reference Architecture Model, as well as the deliverables and software components of the Data Market Austria, provided the basis to design and conceptualize the interoperability solution. This interoperability component provides all functionality required for TRUSTS to interoperate with a selected subset of two DMs (the final DMs will be determined throughout the further advancement of Task 3.3). The interoperability component will be generalized

[2]Namara: app.namara.io/#/sign_in

[3]Otonomo: otonomo.io/

[4]HERE: www.here.com/

[5]CARTO: carto.com/

into a reference architecture, providing the technical and methodological overview for third party DMs to implement their own interoperability solutions. This will give DMs all necessary information at hand that is needed to interoperate with the TRUSTS ecosystem beyond project lifetime.

The examination of existing DMs delivered a diverse landscape regarding their technological readiness to interoperate with them from within TRUSTS. A significant number of them does not provide a machine-operable interface. Instead, data exchange, i.e. upload and download of data, is managed via a web interface for manual usage only. Furthermore, the data catalogs as the backbones of DMs are not easily possible to query. For example, Namara, a many-to-many DM, provides the download of datasets but does not support any facilities to browse through their data catalog via an API. Browsing through the data catalog is possible via its Web interface. From there, dataset ids can be acquired to download the respective dataset via the API. Lastly, other DMs are not markets in the sense that they provide a platform where sellers can offer their datasets for purchase by consumers. Instead, they deliver their own data product, as for example in the case of Otonomo, HERE, or CARTO. These examples are providers of location-based data, which can be consumed via the API. However, they are not marketplaces where data providers can register and offer their own data assets. The decision about what to consider is not part of Task 3.3.

D3.4 “Data Marketplaces with Interoperability Solutions I” is the first report on Task 3.3 “Data marketplaces interoperability solutions” and summarizes the efforts taken so far. It details the selection of DMs, the creation of the software prototypes connecting to DM APIs, the proposed architecture of the interoperability solution, as well as the “Registry of Datamarkets”, a platform envisioned to become a central point for information related to DMs. The Registry of Datamarkets is planned to exist beyond project life time and supposed to help TRUSTS turn into the vibrant data ecosystem it is planned to become.

2 Introduction

This deliverable reports the status of Task 3.3 “Data market interoperability solutions” (direct quotation from the task description). The goal of Task 3.3 is to develop an “interoperability solution for TRUSTS” and “to ensure interoperability with other industrial data marketplaces”. Furthermore, “interoperability solutions with the European Open Science Cloud (EOSC) will be evaluated and implemented where possible”. Task 3.3 aims to solve these sub-tasks in the following manner: a dedicated interoperability component will be conceptualized, and, in a later phase of the project, implemented and deployed within TRUSTS. The development of new standards will encompass the creation of guidelines similar to a reference architecture model. These guidelines will constitute an assistance for existing DMs in case they desire to connect to TRUSTS. They provide a set of blueprints for DMs to implement and deploy their own TRUSTS-compliant interoperability solutions. Lastly, interoperability with EOSC will be evaluated. EOSC is under strong development, Task 3.3 will start the development of respective solutions or best practices once EOSC has reached an appropriate level of maturity. Outcomes of this work will be reported in the second version of the deliverable.

This deliverable summarizes the efforts accomplished so far and outlines insights gained throughout the process as well as conclusions from these insights. Furthermore, it provides the currently envisioned architecture of an interoperability component, i.e. a software component for data exchange and handling of related activities, e.g. harvesting of metadata, or the connection to the clearinghouse as the module responsible for transaction logging and handling of payments. The interoperability component serves as an interface between third-party DMs and TRUSTS. As it is likely that further DMs will emerge after the project's lifetime, a special focus will be laid on Task 3.3 to derive guidelines for DM interoperability. These guidelines will help future DMs to establish their own interoperability components and participate in the data sharing economy of TRUSTS. The "Registry of Datamarkets" described later will serve as the platform to share the interoperability guidelines.

The tasks accomplished so far and described in this deliverable:

- Identify a set of relevant third party DMs based on existing DM listings. An essential criterion is the availability of a machine-operable interface, i.e. an API to explore the whereabouts of data exchange.
- Develop software prototypes connecting to the available APIs. Explore functionality of the DMs using these prototypes.
- Condense a tentative architecture from the insights gained through API exploration. Merge these requirements for an interoperability component with the IDS Reference Architecture Model and the expertise and available software components of the Data Market Austria.
- Lay the basis for the development of a set of guidelines and best practices. Further and final development of these guidelines continues at a later phase of the proposal and will blend in insight gained by attempts to connect to EOSC.
- Initiation of the development of a so-called "Registry of Datamarkets". This is a repository envisioned as a central for all information related to DMs, i.e. background information such as the interoperability guidelines as well as listings of existing DMs. This registry is supposed to become a tool to maintain TRUSTS as a lively data ecosystem also beyond project lifetime.

2.1 Mapping Project's Output

The purpose of this section is to map TRUSTS Grand Agreement commitments, both within the formal Deliverable and Task description, against the project's respective outputs and work performed.

Table 1: Adherence to TRUSTS GA Deliverable & Tasks Descriptions.

TRUSTS Task		Respective Document Chapter(s)	Justification
T3.3 Data marketplaces interoperability solutions	*Based on the findings of D2.1: Definition and analysis of the EU and worldwide data market trends and industrial needs for growth, and by analyzing existing interfaces and standards, and even developing new relevant standards (see T7.4 Standardisation), the interoperability solution for TRUSTS will be designed in this task. This means the definition of interfaces to ensure*	*This is accomplished by participation in Task 2.1 and work on D2.1, which is due in M12 of the project.* *Interfaces of third-party DMs have been investigated in Task 3.3 and are reported in D3.4*	*Interfaces of third-party DMs have been investigated by developing software prototypes connecting to their APIs*

	interoperability with other industrial data marketplaces. In addition, interoperability solutions with the European Open Science Cloud (EOSC) will be evaluated and implemented where possible. Thereby this task has strong interdependencies with T3.2 Smart Contracts, T3.4 Data Governance & Metadata and the overall Work Packages: WP7 Business Plan and WP6 Legal Framework, to ensure interoperability solutions are reflected technically, legally and business wise.	*Methodology, Conceptualization of an interoperability prototype* *This is accomplished by participation in the respective tasks.*	*The sections describe the approach of selecting relevant DMs as well as a conceptual architecture of an interoperability solution*
TRUSTS Deliverable			
D3.4 Data Marketplaces with Interoperability Solutions I This is the first version of a series of three deliverables (D3.4, D3.5, D3.6), which will summarize the integration requirements as well as guidelines for both the TRUSTS platform to interact with existing platforms, EOSC, and for future platforms to integrate with TRUSTS.			

This section provides an outline of the deliverable's structure. It summarizes the steps taken towards the achievement of the goals of Task 3.3 so far and outlines future steps identified throughout the process.

The deliverable is structured into the following sections:

- **Methodology:** This section outlines the strategy used to understand requirements for DM interoperability. It describes the sources used to identify potentially relevant DMs. It explains the filter criteria used to select DMs from the sources. Furthermore, it outlines a preliminary analysis and definition of components required to build the interoperability prototype conceptualized in the subsequent section. The identification of components relies on comparable research attempts, i.e. the IDS Reference Architecture Model as well as the insights gained from the project Data Market Austria.
- **Conceptualization of an interoperability prototype:** One of the goals of Task 3.3 is to ultimately develop and deploy a component to establish interoperability with third party DMs. First steps towards a conceptualization of this component are taken here, outlining a preliminary system architecture as it seems useful and feasible at the current stage of knowledge and insight into the TRUSTS requirements and the components of both IDS and DMA.
- **Conclusions and next steps:** This section summarizes the insights gained throughout the current phase of Task 3.3. It also explains caveats detected upon closer inspection of DMs and available components.

3 Methodology

The following sections describe the approach followed within Task 3.3, starting with a literature-backed definition of DMs. An overview of relevant sources as well as the description of software prototypes implemented to connect with existing third party DMs follows. The Section closes with a description of components from the IDS relevant for Task 3.3.

3.1 Mapping the Area of Data Markets

An important aspect for Task 3.3 “Data marketplaces interoperability solutions” is the selection of platforms fulfilling the requirements to be considered as DMs. This is the essential first step before the design, conceptualization, and implementation of an interoperability software component. The literature provides guidance in this process by giving technical definitions and the provision of databases of DMs. We give an overview of technological features, characteristics, and business aspects of existing DMs based on our research and explain the considerations that were the basis for the decision, which DM would be used for implementing interoperability prototypes.

DMs exist in a wide variety of flavors and for different domains. Definitions of DMs are diverse and heterogeneous. Stahl et al. [1] divide electronic DMs into different categories based on their supplier/buyer relations. DMs can serve single or multiple buyers and single or multiple suppliers (see Figure 1). In some cases, the single supplier is equivalent to the DM operator itself. DMs are categorized in three different sectors: private, consortium and independent DMs. A private DM has either a single buyer or single supplier, i.e. it offers one-to-many or many-to-one relations to the sides of buyers/suppliers. Consortium DMs are unions of either suppliers and buyers, or both. The third type, the independent DM, serves as an arbiter for buyers and suppliers, i.e. providing a platform and infrastructure for successful search, acquisition, accounting, and transfer of data. They allow for many-to-many relations between independent players in the DM.

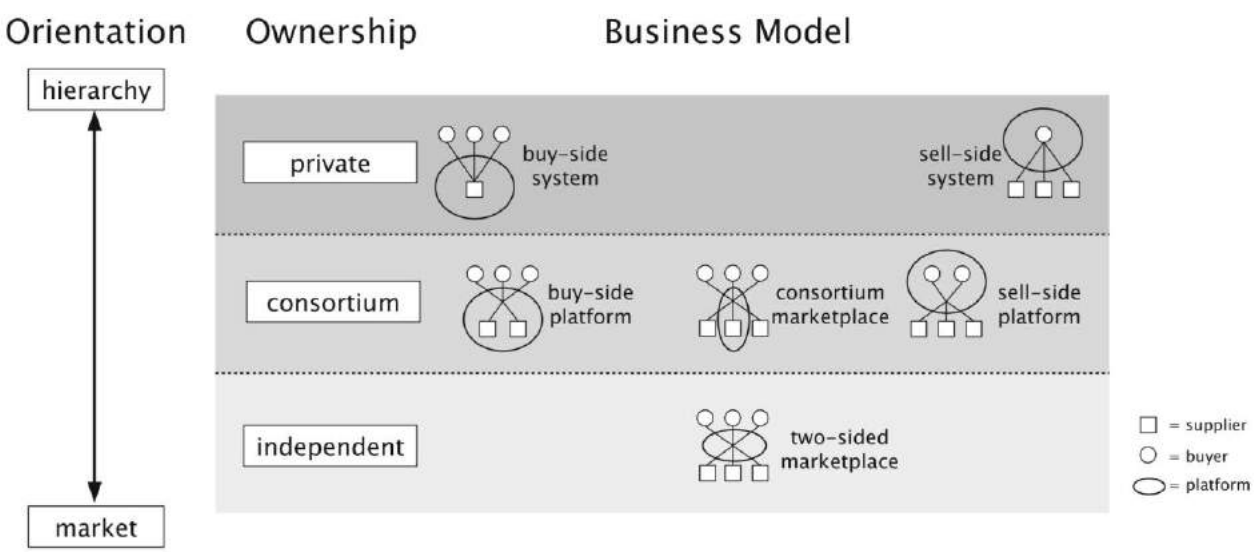

Figure 1: Categorization of electronic DMs based on their supplier/buyer relations [1].

In contrast to this lenient categorization of DMs, the IDS Reference Architecture Model [7] provides a much stricter definition. DMs are exclusively limited to the independent type, i.e. require the conceptual availability of m:n relations between independent buyers and suppliers. Data providers such as Facebook, who sell their data products to external organizations, are consequently not considered as a DM. Alternatively, the definition of Stahl et al. [1] would allow those to be considered as a DM.

DMs cover a wide range of different domains, such as agriculture, finances, healthcare, location services, etc. According to [7], the by far most prominent domain is the so-called audience data, providing data from the marketing and advertising industry. Table 2 illustrates an overview of the distribution of DMs by their domains.

Table 2: The number of DMs by type [6].

Labelling of cases by type(s) of data traded on the marketplace (based on labelling of datarade.ai, Fruhwirth et al., 2020 and Spiekermann, 2019)	Number of cases in the database (N)
Agriculture data	2
Alternative data	5
Any data	8
Audience data	112
B2B data	9
Connected car data, automotive data	4
Data for AI and machine learning	3
Environmental data	1
Financial data	2
Financial data, alternative data	5
Financial data, market data	2
Healthcare data	8
Location data	6
Parking data	1
Personal data	4
Real estate data	1
Sensor data	2
Satellite data	1
Traffic data, petrol price data, parking data	2
Total	**178**

Another important aspect to differentiate DMs are features regarding their business model [6]. For example, their value propositions range from “Easy data access and/or tooling” over “Secure data sharing” and “High quality and unique data” to “All services in a single platform”. They differentiate with regard to their marketplace participants (B2B, C2B, or any), their geographic scope (global, regional, local), from a technological perspective regarding their architecture (centralized vs. decentralized), or in the financial domain via their pricing models (freemium, pay-per-use, flat fee tariff, package based pricing, multiple). Table 3 gives a taxonomy of DM business models (cited from [6]).

Table 3: A taxonomy of DMs with respect to their business model [6].

	Dimension	Characteristics					
Service domain	Value proposition	Easy data access and/or tooling	Secure data sharing	High quality and unique data	All services in a single platform		
	Enterprise data marketplace	Yes	No				
	Data processing and/or analytics tools	Yes	No				
	Marketplace participants	B2B	C2B	Any			
	Industry domain	Any data	Geo data	Financial & Alternative data	Health & Personal data	Audience data	Sensor & Mobility data
	Geographic scope	Global	Regional	Local			
	Time frame	Static	Up-to-date	(Near) real-time	Multiple		
Technology domain	Platform architecture	Centralized	Decentralized				
	Data access	API	Download	Specialized software	Multiple options		
	Data source	Self-generated	Customer provided data	Acquired data	Multiple sources		
Organization domain	Matching mechanism	One-to-one	One-to-many	Many-to-one	Many-to-Many		
	Platform sponsor	Private	Consortium	Independent			
Finance domain	Revenue model	Commissions	Subscriptions	Usage fees	Asset sales		
	Pricing model	Freemium	Pay-per-use	Flat fee tariff	Package based pricing	Multiple	
	Price discovery	Set by buyers	Negotiation	Set by marketplace provider	Set by external sellers		
	Smart contract	Yes	No				
	Payment currency	Fiat money	Cryptocurrency				

There is also a plethora of software solutions for operating self-hosted data management platforms, such as CKAN[6], Invenio[7], or Amundsen[8]. They are open-source, on-premise solutions for organizations to manage and administrate their data. CKAN, for example, is a widely known platform used for the administration of research data and provides all functionality of a data catalog, including search facilities. Invenio, which was created and is maintained by CERN, provides similar features. In addition to keyword-based search, it features a recommendation engine for cases where the search engine was not successful. Invenio is more a framework than a standalone platform and lets operators flexibly add components required for their data management. There are lightweight installations of Invenio with a reduced feature set and deployments that are more functional with a wider range of functionality. Lyft, the taxi summoning company, develops Amundsen. It provides a rich feature set comparable to CKAN

[6] CKAN: ckan.org

[7] Invenio: invenio-software.org

[8] Amundsen: github.com/amundsen-io/amundsen

and Invenio and leverages big-data technologies such as Apache Hive[9] and Elasticsearch[10]. These platforms, however, are more equivalent to data catalogs and do not provide functionality required by DMs, such as billing functionality or tracking of transactions. Thus, they can serve as modules used to build a DM but are not substitutes of them. Furthermore, they are not shipped pre-loaded with data but have to be populated with data by the operators or their network of customers and suppliers. Table 4 gives a comparison of the features provided by a set of prominent data management platforms, assembled by [5].

Table 4: Comparison of features provided by a selected set of data management platforms [5].

	Feature	DSpace	CKAN	Figshare	Zenodo	ePrints	EUDAT
Architecture	Deployment	Installation package or service	Installation package	Service	Service	Installation package or service	Service
	Storage Location	Local or remote	Local or remote	Remote	Remote	Local or remote	Remote
	Maintenance costs	Infrastructure management	Infrastructure management	Monthly fee	Monthly fee	Infrastructure management	Monthly fee
	Open Source	✓	✓	×	×	✓	×
	Customization	✓	✓	×	Community policies	✓	×
	Internationalization support	✓	✓	×	×	✓	×
	Embargo	✓	Private Storage	Private Storage	✓	✓	✓
	Content versioning	×	✓	×	×	✓	✓
	Pre-reserving DOI	✓	×	✓	✓	✓	✓
Metadata & Dissemination	Exporting schemas	Any pre-loaded schemas	None	DC	DC, MARCXML	DC, METS, MODS, DIDL	DC, MARC, MARCXML
	Schema flexibility	Flexible	Flexible	Fixed	Fixed	Fixed	Flexible
	Validation	✓	×	×	✓	✓	✓
	Versioning	×	✓	×	×	✓	✓
	OAI-PMH	✓	×	✓	✓	✓	✓
	Record license specification	✓	✓	✓	✓	✓	✓

3.1.1 Identification of Relevant Existing Data Markets

For the identification of relevant third party DMs, we relied upon existing relevant collections. One source was the EU Data Landscape, an online platform to monitor players in the European data economy, while the other was developed in [6]. The EU Data Landscape was a result of the European Data Market study[11], which aims to quantify the European data ecosystem. Besides the EU Data Landscape, this study also produced the Data Monitoring Tool, which helps to assess future growth of the data economy by the EU member state and economical sector. Figure 2 shows the EU Data

[9] Apache Hive: hive.apache.org

[10] Elasticsearch: www.elastic.co

[11] European Data Market study: datalandscape.eu/

Landscape providing search facilities to identify components of the EU data economy. For Task 3.3, the "Marketplaces" were relevant. At the moment of writing this deliverable, 14 marketplaces were listed under this category.

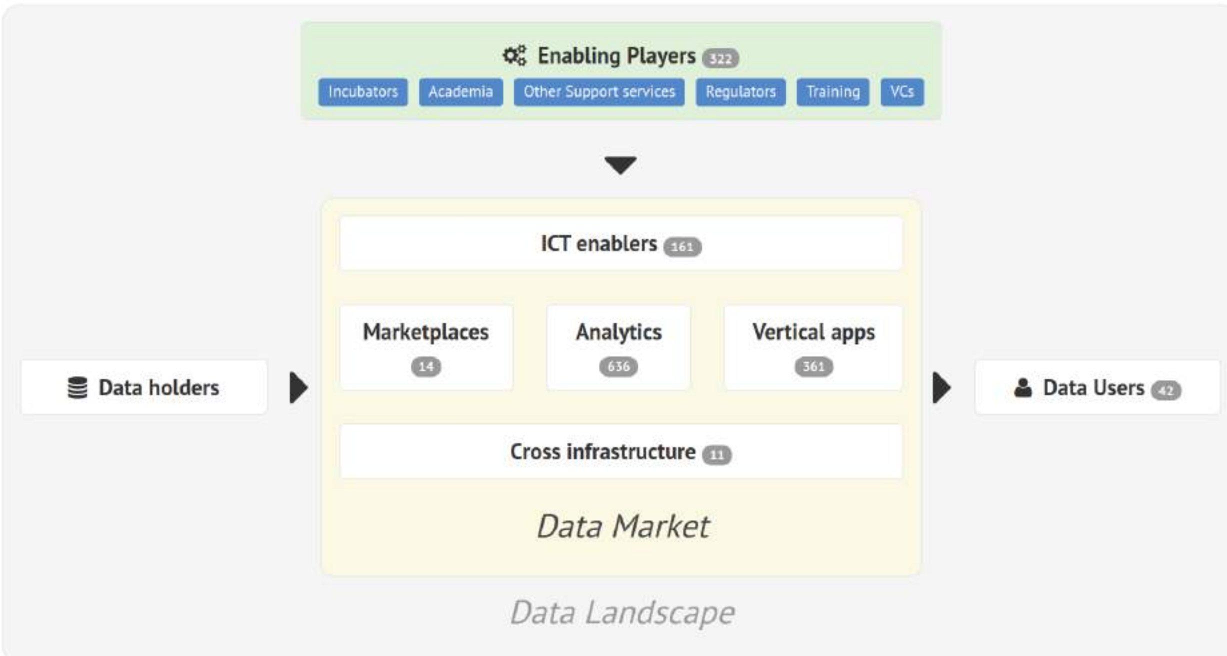

Figure 2: Key components of the EU data landscape indicating 14 available DMs (last accessed Oct 29, 2020).

Another feature of the EU Data Landscape is a map showing the geographic location of key players, as shown in Figure 3. Interestingly, there is also one marketplace listed residing outside of the EU (BlueTalon[12]).

Figure 3: Geolocations of the DMs in the EU data landscape (last accessed Oct 29, 2020).

12 Blue Talon: bluetalon.com

As mentioned, we used the EU Data Landscape as a source to systematically review the DMs listed there. Unfortunately, the list seems out-of-date, as many of the DMs listed do not exist anymore. The registered websites were not available or there was a notification on the website about the company being out of business (e.g. as in the case of dmi.io). Table 5 outlines a summary of this systematic review. From the list of 14 DMs, DAWEX was identified as a DM relevant for investigation. It is an m:n DM and also provides the technical facilities for data providers to register external APIs within DAWEX. Data consumers can retrieve authentication tokens via DAWEX and use them to acquire data from the provider via their API. For the TRUSTS platform, this approach is highly relevant, as TRUSTS can also conceptualize, implement, and deploy an API within Task 3.3 and make it accessible on DAWEX.

However, DAWEX itself does not provide its own API to allow for machine interaction. Consequently, it is not possible to harvest DAWEX' data catalog and make it available within the TRUSTS platform. Furthermore, it is not possible for TRUSTS to become a reseller of DAWEX datasets. Instead, DAWEX will operate as a reseller of TRUSTS datasets. The DAWEX example can constitute a case study for the business-related work packages of TRUSTS towards developing appropriate business models, as DAWEX is a prominent DM within the European data ecosystem.

Table 5: Status of the DMs listed in the EU data landscape.

Name	URL	Note
dmi.io	dmi.io	Company out of business.
BlueTalon	bluetalon.com	Website not available.
Smart Jobs S.L	www.jobinow.com	Website not available.
DAWEX	www.dawex.com	Eligible for interoperability.
WhoApi	whoapi.com	Provider of IP related information.
qDatum	www.qdatum.io	Language barrier: website is in Hebrew.
Rocketgraph	rocketgraph.com	Company out of business since 2017.
City Context Open Data API	www.citycontext.com	Website not available.
Helix Nebula Science Cloud	www.helix-nebula.eu	Cloud initiative related to the European Open Science Cloud. Seems outdated, latest news from 2017.
DataScouts	www.datascouts.be	Website not available.
Datalayer	datalayer.io	Repository of curated data science notebooks. API availability unclear.
GLOBMOD	www.globmod.com	Health intelligence lab. No API available.
Spaziodati	www.spaziodati.eu	Provision of a knowledge graph via REST API.

Open Corporates	opencorporates.com	Database on companies. API availability unclear.

From the DM list included in [6], we examined the DMs listed in Table 6. Finding DMs with appropriate mechanisms for developing an interoperability solution is a challenging task. Some of them do not provide APIs, e.g. Kraken. Other DMs mention the availability of an API on their website, but upon closer inspection it turned out that there is nothing implemented, e.g. in the case of Advaneo and DAWEX. Some DMs do not use REST APIs but cryptocurrency-based solutions instead, which requires an implementation effort beyond the scope of T3.3, e.g. Streamr and Databroker. Furthermore, there are DMs exposing an actual API, but it is behind a paywall and consequently inaccessible now, such as in the case of Caruso and Otonomo, where the latter's free API key expires after 30 days. Merely three of the examined 16 DMs provide direct access to their APIs. At the current stage of the project we focus on these three DMs, i.e. Namara, HERE and CARTO. Based on insights and analysis results, we will develop (i) an interoperability prototype and (ii) guidelines and best practices for other DMs to shape their APIs in a way that allows for simple and robust connection to the TRUSTS ecosystem. Table 6 gives a detailed overview of the examined DMs and the status of their REST APIs. In the future course of the project, we will examine further DMs for suitable APIs and increase the number of software prototypes connecting to them. Furthermore, we extend our knowledge and expertise of the technical background of DM with a survey launched in November 2020. The survey contains technology- and business-related questions. The technology-related part will help us to learn about the technology stack used by DMs, and examine common characteristics. Based on the insights gained we will conceptualize and implement an interoperability solution that fits their needs. The survey was targeted at 25 DMs, which were select from the list of DMs compiled by [6] in a manual selection process by two annotators. See Section 5.1 for more information. We also plan focus groups with DM operators and stakeholders for the beginning of next year. The focus groups will give access to knowledge of the invited experts.

Table 6: Listing of the examined DMs and their API availability.

DM Name	API Available?	Notes
Advaneo	?	Mentions an API, but there is no description.
Skywise	✓	Skywise has an API, but it is not possible to create new accounts.
Kraken	X	No API available
BDEX Marketplace	?	Mentions an API, but there is no description.
Caruso Dataplace	✓	Request approval required.
Streamr	✓	REST API is deprecated, exchange possible via cryptocurrency (Ethereum).
Databroker (formerly Databroker DAO, after the	✓	No REST API, exchange possible via cryptocurrency (Ethereum).

rebranding this documentation may be outdated)		
Opendatasoft	X	Request approval required.
Otonomo	✓	The API access expires after 30 days.
Caruso	✓	API keys are available in paid licenses only.
Namara (ThinkDataWorks)	✓	Connection via API is possible.
HERE	✓	Connection via API is possible.
CARTO	✓	Connection via API is possible.

For this deliverable, we built prototypes connecting to their API for Namara, HERE, and CARTO, from the abovementioned list. We also built a prototype for Streamr. However, here a further investigation of its REST API was futile, because Streamr is going to discontinue the provision of a REST API and will feature interconnectivity based on cryptocurrencies instead. The possibility of connecting via a cryptocurrency will be investigated in the forthcoming period, as a next step, exploring firstly its relevance with the TRUSTS objectives, since the incorporation of such a feature in TRUSTS would introduce highly increased complexity as compared to simple authentication via a REST API. Namara allows many-to-many relations between data suppliers and consumers. In other words, suppliers can share their datasets on the market and multiple consumers can acquire them under the defined license clause. Data suppliers include public sources, such as the U.S. Bureau of Labor Statistics or Data.gov, but also private entities such as AirBnB, Nasdaq, or Best Buy. This results in a rich variety of available datasets, e.g. on consumer prices, household indicators, or datasets for machine learning. Datasets can either be downloaded manually via the Web interface, e.g. as CSV files, but also programmatically using their API. In the free version, up to 100 downloads per month are possible.

HERE, CARTO, and Otonomo provide location intelligence data, for example for fleet planning. They have convenient and easy-to-use APIs. Otonomo has the caveat that the license for free trial expires after 30 days. This complicates development effort, making it impossible to test later code changes with the expired API. Instead, it would require repeated creation of accounts on Otonomo, which is a legally questionable approach.

3.1.2 Analysis of Existing Data Markets

To better understand the characteristics of data exchange provided by third party DMs we developed a set of four code prototypes to investigate the functionality of these DMs. The DMs used for this analysis were Namara, CARTO, HERE, and Streamr. Authentication on these DMs works via API keys. This requires the creation of an account on the respective platform. Subsequently, the API keys can be retrieved from the online interface.

Google Colaboratory[13] served as the platform to implement the code prototypes. The advantage of this platform is its ease of setup. The platform is made for the programming language Python and provides a programming environment closely resembling Jupyter Notebooks[14] (a tool widely used for data science).

Figure 4 shows a code snippet retrieving data from HERE. One of the features of HERE is to allow searching the proximity of a given geolocation. A potential use case is the identification of restaurants close to a location, which is shown in the code snippet. The interface of Google Colab is easily visible. Two variables define latitude and longitude, another variable refines the search for “restaurant”. Subsequently, a request is sent to HERE and the respective data is transmitted back. The result is parsed out of the request response, and 15 restaurants close to the given geolocation are listed.

The other selected DMs were examined in a similar way. Google Colab prototypes include enough functionality to assess the features and characteristics of the DMs. In the later phase of the project, these characteristics will be analyzed in more detail and aligned with the ongoing TRUSTS architecture specification.

[13] Google Colab: colab.research.google.com

[14]Jupyter Notebook: jupyter.org/

Proximity-based search for points of interest

```python
# Geo-location of Thurngasse 8
latitude = 48.21882
longitude= 16.3601
# Search for a restaurant
query = 'restaurant'
# Include 15 search results
limit = 15

url = URL_DISCOVER.format(latitude=latitude, longitude=longitude,
                          query=query, limit=limit, api_key=API_KEY)
req = requests.get(url)
res = json.loads(req.content)
```

```python
for item in res['items']:
    print(item['address']['label'])
```

```
Dreiklang - Herbert Hofer Essen und Trinken, Wasagasse 28, 1090 Vienna, Austria
Fleischerei Otto Mayerhofer Gesmbh, Liechtensteinstraße 27, 1090 Vienna, Austria
Asiatisch, Liechtensteinstraße 23, 1090 Vienna, Austria
Restaurant Oishi, Liechtensteinstraße 23, 1090 Vienna, Austria
Rocco Gastronomie GesmbH, Liechtensteinstraße 24, 1090 Vienna, Austria
Rocco Gastronomie GmbH, Liechtensteinstraße 24, 1090 Vienna, Austria
I Vecchi Amici, Liechtensteinstraße 24, 1090 Vienna, Austria
Koperwas Restaurant, Liechtensteinstraße 26, 1090 Vienna, Austria
People's, Thurngasse 4, 1090 Vienna, Austria
Intermezzo, Thurngasse 4, 1090 Vienna, Austria
Maximoff, Thurngasse 17, 1090 Vienna, Austria
Necov KEG, Liechtensteinstraße 33, 1090 Vienna, Austria
Pizzeria Pronto, Liechtensteinstraße 33, 1090 Vienna, Austria
Valentino-Panella OG, Liechtensteinstraße 33, 1090 Vienna, Austria
Restaurant Skopje, Liechtensteinstraße 33, 1090 Vienna, Austria
```

Figure 4: Results of proximity-based search on HERE.

3.2 Analysis of Required Components

3.2.1 The Role of IDS

An essential aspect of the TRUSTS project is the reuse of components from both the IDS and the DMA. The IDS provides an extensive reference architecture model [7] for orientation when building data spaces. It defines key concepts, constituents, their relationships, as well as requirements for data exchange in trusted business environments. The IDS reference architecture model considers a set of participants, for example data providers, data consumers, data users, the clearinghouse, or service and software providers. They interact with each other in the ecosystem of the IDS, exchange data, sell it, purchase it, process and analyze it.

The IDS reference architecture model provides a detailed vision of the different roles interacting in a data ecosystem and their relations among each other (see Figure 5). The central role is taken by the infrastructure of the DM itself, which is hosted by a DM owner. The owner interacts with both data providers and data buyers by hosting data assets from providers and selling them to data buyers. There are both commercial and non-commercial data providers. On the buyer side, there are consumers or businesses making purchases. Third-party service providers also interact with both the data marketplace as well as data providers and buyers. They increase the value of data assets, e.g. by providing preprocessing or analytics functionality in the form of applications or services.

A lively data ecosystem requires the participation of all of these stakeholders. Data assets are not limited to just mere datasets. They also include the aforementioned applications and services, which potentially can take many forms such as analytics applications or machine-learning models pre-trained on existing datasets.

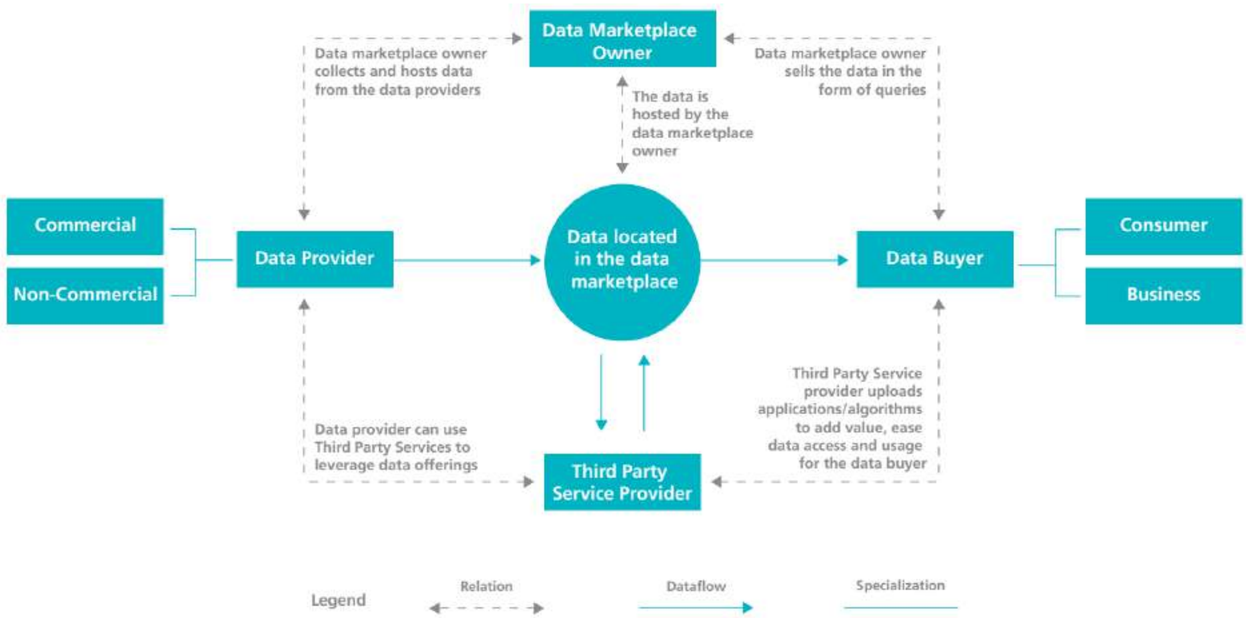

Figure 5: High-level view on the different roles in data ecosystems [3].

The IDS reference architecture model also sketches constituents and participants in addition to the roles stakeholders can take in data ecosystems (see Figure 6). Core participants are data owners, data providers, data consumers, and data users. Data owners and data providers might be a single entity, similarly to data consumers and data users. In case they are separate entities, data owners authorize providers to offer their data on the data market. Core participants interact with each other via a set of constituents, which provide the functionality for smooth interaction within the data ecosystem. The broker component connects data consumers with those data providers supplying the data required for their business needs. A broker might contain a data catalog to systematically file available data. A clearinghouse component facilitates accounting, closes purchases between buyers and sellers, and logs transactions. Producers of applications can offer them from within an app store, where they are ready to be downloaded by interested data providers or data consumers. Applications might serve a variety of purposes, for example pre-processing and homogenisation of data, data analytics and model creation, visualizations and dashboards, etc. Service providers can offer their services on top of available data assets and support organizations in the implementation of their own data-driven projects. Vocabulary providers supply taxonomies and ontologies to data providers, ensuring the adherence to naming conventions and metadata standards.

With regard to interoperability, third-party DMs operate as both data providers and data buyers. The interoperability component envisioned in Task 3.3 will provide functionality for both roles. On the one hand, it will access data assets in DMs and publish their availability within TRUSTS, increasing the visibility of DMs and their data assets. An appropriate business model is required to incentivize DMs to participate in this form of data sharing, which is not part of Task 3.3 and it will be elaborated within WP7. On the other hand, the interoperability component as well as the implementation guidelines derived during its development process will allow DMs to acquire data assets from TRUSTS and offer them for purchase on their premises. The exact details of both modes of interaction require both legal and economic affirmation by the respective consortium members within TRUSTS. Based on their decisions, respective functionality will be provided within the interoperability component.

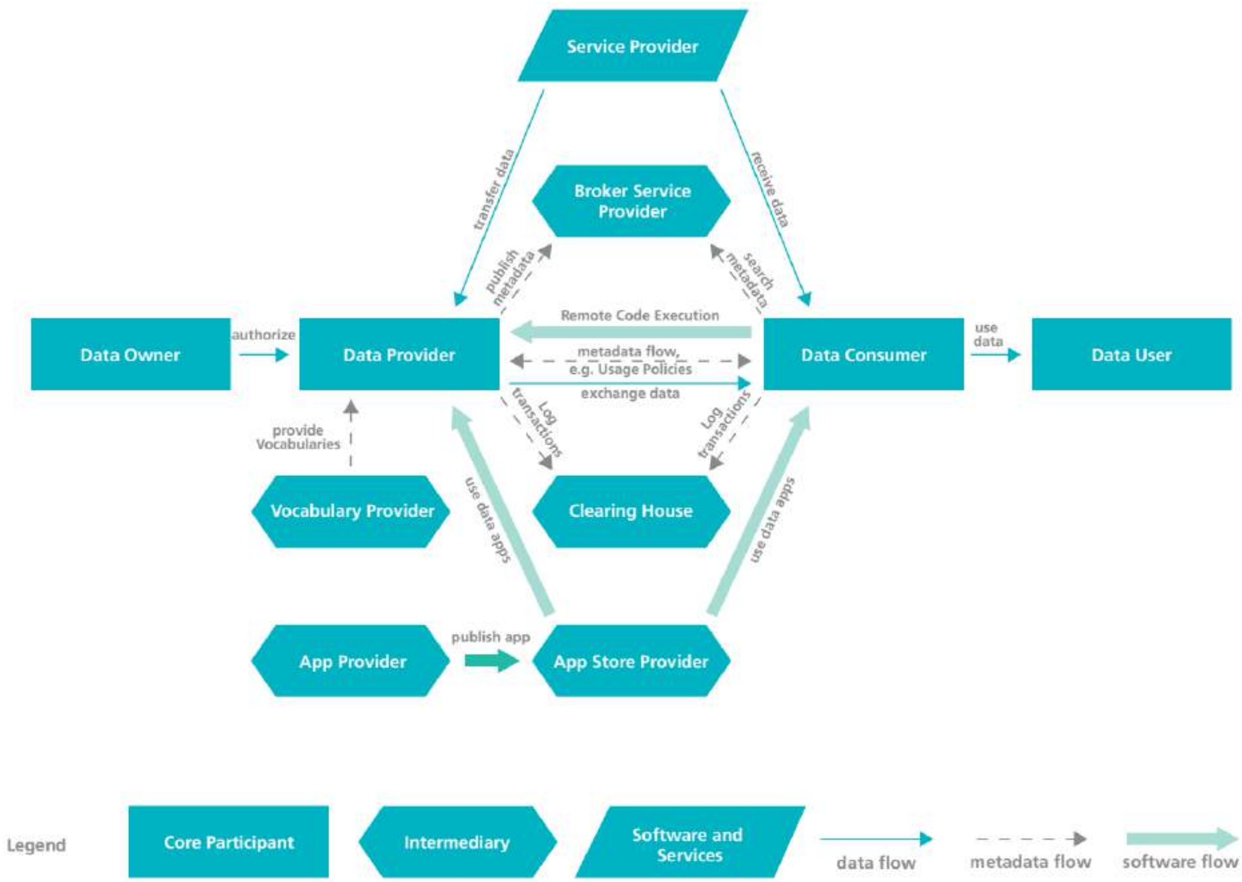

Figure 6: The constituents and participants of data ecosystems [3].

The TRUSTS platform is envisioned as a federated architecture, in contrast to a centralized architecture where the majority of functionality is delivered by a single provider or node. Constituents run on their own nodes or on nodes hosted by infrastructure providers, respectively. Similarly, participants run their software on their own or hired nodes. This kind of architecture requires a technology to link them with each other, exchange requests, and transfer data. The IDS foresees the connection of constituents and participants via a dedicated interface, the connector. Figure 7 illustrates the role of the Connector as a link between different constituents of the IDS, e.g. the catalog connects to the broker and the app store.

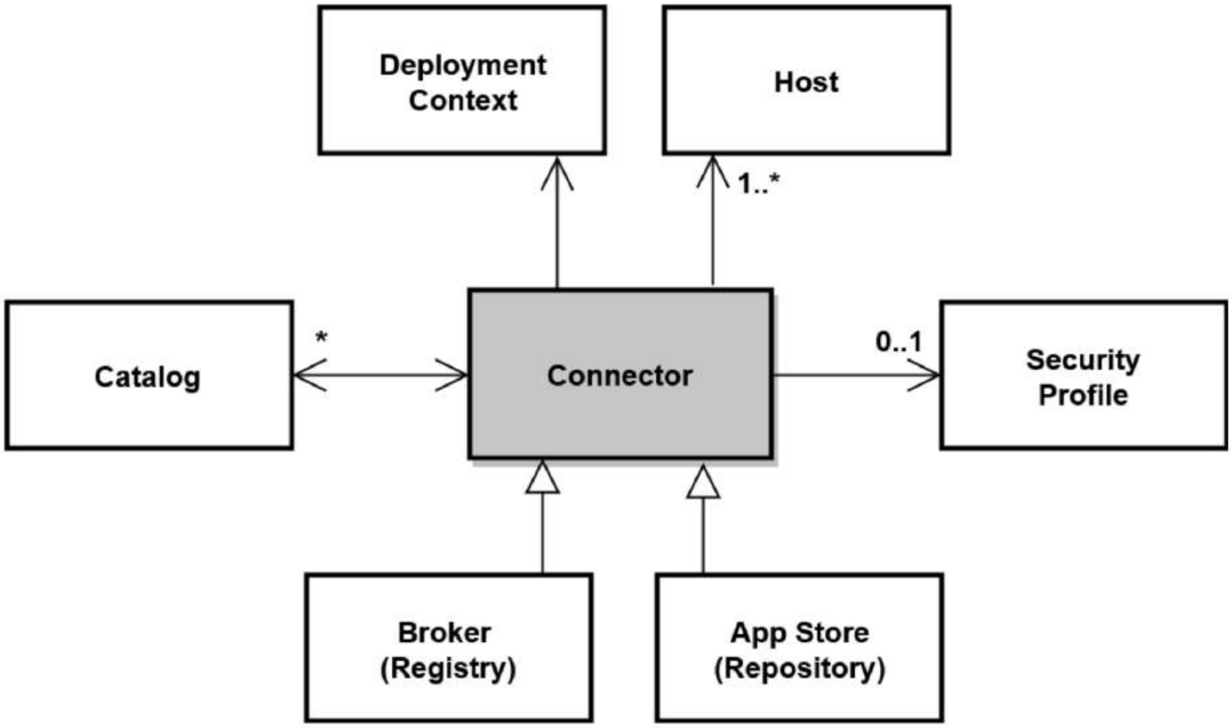

Figure 7: The Connector as the interface between IDS constituents [7, p.56].

The purpose of the Connector is the exchange of data (e.g. metadata) between all constituents of the IDS. Thus, it provides a standardized interface each constituent can rely upon and has to implement. It avoids security issues by providing an isolated environment. As a general guideline, data processing should be as close to the data source as possible to prevent security issues. The exact implementation of the Connector depends on the use case at hand and can result in variations of Connector implementations. Figure 8 illustrates data exchange in the IDS via the Connector. One connector transmits data and metadata from a data source to another connector linked to a data sink. Bidirectional data transfer is foreseen and possible. A broker exchanges metadata with both the data source and the data sink Connector. Furthermore, an app store transfers its apps to the Connector at the side of the data source.

Data exchange in the IDS is accomplished exclusively using the Connector concept. A potential scenario is that a data consumer requests a dataset or data service from the broker. The broker looks up the addresses and locations of data providers possessing the relevant dataset or service and shares this information with the data consumers. Subsequently, the data consumer establishes a dedicated connection between itself and the data provider. Once the connection has been established, transfer of the respective dataset or service starts. As a final step, data provider and data consumer accomplish accounting via a connection to the clearinghouse and settle an invoice. The clearinghouse furthermore keeps track of accomplished transactions and logs them.

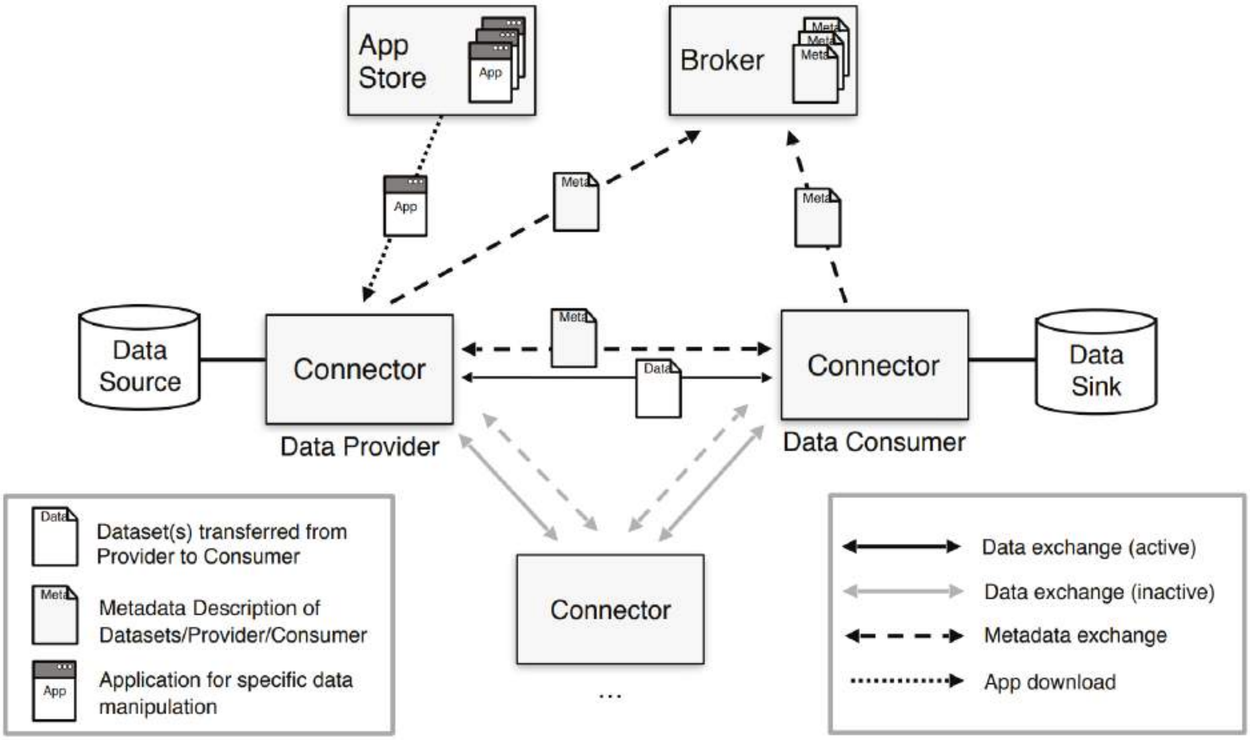

Figure 8: Exchange of data assets via the Connector [7, p. 60].

The Connector is managed using container technology and differentiates between an execution and a configuration phase ([7]). Figure 9 illustrates these two phases and the constituents participating in each phase. The execution and configuration phase are isolated from each other, which facilitates implementation and deployment of components without influencing the others. In the execution phase, self-written services ("Customer Container") or services downloaded from an app store ("App Store Container") exchange data with the core container. The data router of the core container establishes communication with the services, while the data bus accomplishes the actual data exchange. Neither the data router nor the data bus have mandatory technology stacks, i.e. they might be exchanged with different technology or implementations if the use case requires it.

As the name implies, the configuration phase covers the configuration of the Connector. This involves the development of a configuration model and its deployment. The configuration manager orchestrates these operations and validates the configuration model before triggering its deployment by the execution configurator. A dedicated validator checks the configuration model for compliance with the rules imposed by the IDS. In cases of violation, this might incur the raising of warnings or failure of deployment.

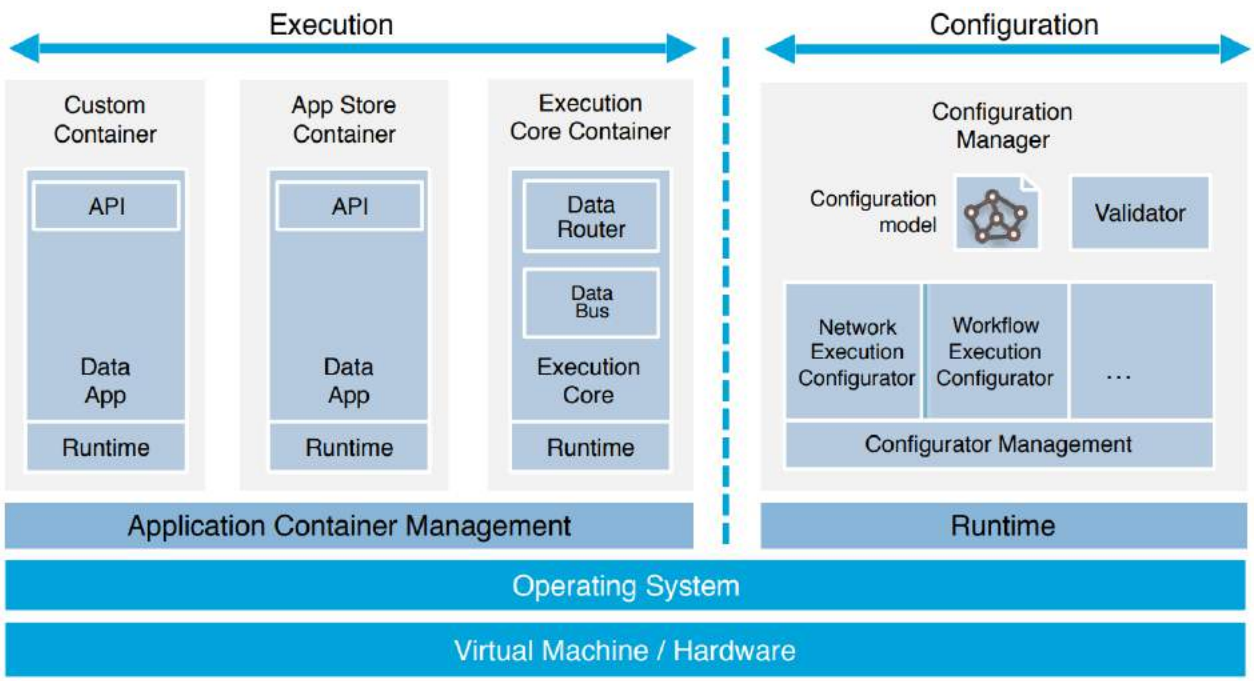

Figure 9: The technology stack and architecture of the Connector [7, p. 62].

There is not a single, one-serves-it-all, multi-purpose Connector. Instead, actual implementations of Connectors depend upon the requirements of the given use case and can differ greatly. The reference architecture model lists three examples for Connectors [7]:

- The developer Connector: In a development scenario, the usage of container management might be omitted for reasons of simplicity and speed of development.
- The mobile Connector: In mobile applications, where resources are often sparse, container management might also be omitted to avoid the additional computational overhead and save resources for actual applications.
- The embedded Connector: This type of Connector also omits container management, similar to the developer and mobile Connector. In addition to this, such a Connector might also exclude a configuration manager, which in turn requires remote configuration of the Connector.

A special type of Connector is the so-called "Trusted Connector". It puts a special focus on secure and trustworthy data exchange and provides data protection, security, and trust. It emphasizes the concept of data sovereignty, i.e. the ability of a data provider to define rules and requirements for usage, exchange, management, processing, and analysis of its data. The following section describes the Trusted Connector in more detail.

3.2.2 The Trusted Connector

The Trusted Connector is an execution platform to supply internal data within the IDS infrastructure and to run applications for data processing and analysis. It puts special emphasis on secure data exchange and a trustworthy interaction of the participants in a data ecosystem and as such is a crucial component to implement and guarantee data sovereignty. It is built on top of the Connector specification by the IDS reference architecture model [7]. The architecture of the Trusted Connector implements security by encapsulating service applications and restricting their access to the network interface (see Figure 10). In addition, it leverages a remote attestation procedure, which certifies the trustworthiness and validity of applications and services.

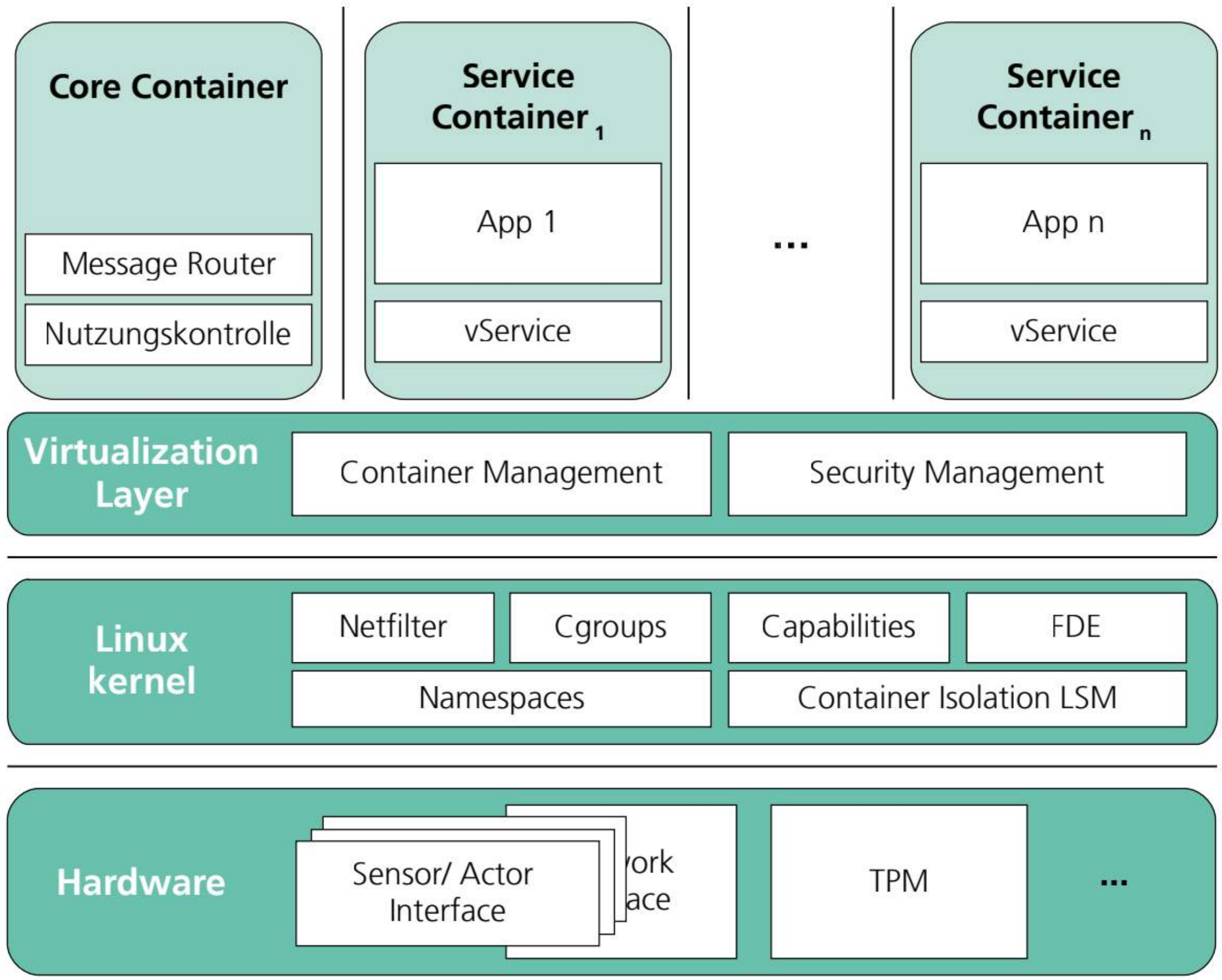

Figure 10: The technology stack of the Trusted Connector [8, p. 7].

The core platform of the Trusted Connector handles message routing between containerized apps. Message routing is accomplished using Apache Camel[15].

During start-up of the Trusted Connector, a secure boot process based on the specifications of the Trusted Computing Group[16] checks the validity of the used software components. This remote attestation procedure knows three levels of security:

- Level 0: This is the lowest security level. The participating connectors merely exchange information that no TPM or secure boot process is available.
- Level 1: This level verifies core components of the Trusted Connector such as firmware, bootloader, kernel, and the core platform

[15]Apache Camel: camel.apache.org

[16]Trusted Computing Group: trustedcomputinggroup.org

- Level 2: In addition to the verifications of level 1, this level encompasses the verification of installed apps.

3.2.3 Broker Format Mapper

This component takes as input the data or metadata as received by a third party DM and transforms it into a format compliant with the TRUSTS broker format. The Broker Format Mapper will be developed based on a selected subset of suitable DMs (e.g. two relevant DMs). The components will be integrated into the TRUSTS platform.

The purpose of the Broker Format Mapper strongly resembles the Mapping Builder developed in TRUSTS T3.4 “Data Governance: Metadata, Lineage, and Semantic Layer”. In the next period, we will clarify if it is advisable to merge the two components into one and report the chosen approach in the next version of the deliverable.

3.2.4 DM Metadata Crawler

The metadata crawler links to external DMs and harvests available metadata from them. The data is passed over to the Broker Format Mapper (see previous section). The DM Metadata Crawler shares significant functionality with the component “Metadata Harvester” from the DMA project. Consequently, the existing code of this component will be thoroughly examined to assess reusability and avoid the necessity for implementation from scratch.

Harvesting metadata requires a structured crawling scheme, i.e. a schedule determining the time intervals of harvesting to avoid outdated information about a DM’s data in the TRUSTS broker. Refinement of this schedule will be exploratory, i.e. a preliminary schedule will be determined using an educated guess. Subsequently, content changes will be compared and the crawling schedule adapted accordingly.

3.2.5 DM Operation Converter

Third-party DMs provide a different set of functionality. If external DMs are supposed to be operable from within TRUSTS, their functionality needs to be mapped to an equivalent set of functions within TRUSTS. This requires both an availability of those functions from the backend, but also an appropriate representation from a user interface. Ideally, the user interface should follow a design principle where it closely follows the native design of TRUSTS, but also the look-and-feel of the external DM. This ensures ease of use by providing interfaces that are most familiar to DM users, both from the TRUSTS perspective but also from the external DM.

3.2.6 Clearing House

The clearinghouse is a component envisioned by the IDS Reference Architecture Model. Its purpose is to log transactions and provide the means for accounting. In the IDS Reference Architecture Model, the clearinghouse provides accounting facilities for purchases between data consumer and data providers. In

Task 3.3, this functionality will have to be widened to the needs of interaction with external DMs. Several questions need to be addressed here:

- How to provide enough flexibility within the clearinghouse to cater for the needs of a diverse set of payment mechanisms, as is expected from different DMs?
- Should payment based on cryptocurrencies also be implemented? If yes, what does an appropriate solution look like?

3.2.7 MQTT

MQTT[17] (Message Queuing Telemetry Transport) is a messaging protocol for the Internet of Things following an OASIS[18] standard. It uses a brokerage concept to broadcast messages. Clients in the MQTT network can publish their messages to the broker under self-defined topic names. Clients can also subscribe to receive updates on existing topics. The MQTT broker broadcasts updates and all clients subscribed to the given topic receive the update. In case there are no subscribers for a topic, new update messages are discarded. Application scenarios are for example temperature sensors publishing updates about the temperature at specific time intervals. They publish their temperature data to the MQTT broker, who in turn broadcasts it to subscribed clients. The clients could use cloud technology to serialize the temperature values in a database.

MQTT was specifically designed to cater for the needs of the IoT. It is lightweight in terms of resource-intensiveness and makes it possible to connect millions of devices to the cloud. It is used in the areas of automotive, manufacturing, smart homes, logistics, and many more. Reference implementations exist for a plethora of programming languages. The interoperability prototype will use the Eclipse Paho MQTT Python Client[19] as a client library and the Eclipse Mosquitto MQTT broker[20].

3.2.8 Metadata Mapping

The metadata of datasets and services from external DMs will be mapped to the TRUSTS metadata schema and subsequently serialized in the metadata store. The mapping is a transformation of the present metadata schema of the respective DM to a schema following established standards, such as Dublin Core. The Dublin Core is a widely recognized standard used to describe resources, e.g. videos, audio files, books, etc. It consists of 15 core elements, which are part of a bigger set of elements, the DCMI Metadata Terms[21].

Table 7 shows the mapping of Namara metadata to the Dublin Core core elements.

Table 7: Mapping of general Namara data to the Dublin Core core elements.

DCMI Element	Namara Equivalent	Note

[17] MQTT: mqtt.org

[18] OASIS (Organization for the Advancement of Structured Information Standards): www.oasis-open.org

[19] Eclipse Paho MQTT client: github.com/eclipse/paho.mqtt.python

[20] Eclipse Mosquitto: mosquitto.org

[21] DCMI Metadata Terms: dublincore.org/specifications/dublin-core/dcmi-terms/#

Contributor	---	Not available
Coverage	---	Not available
Creator	Namara	Not available
Date	Last updated	Not available from API, only from web interface
Description	Description	Not available from API, only from web interface
Format	Result format	Default is JSON, CSV and GEOJSON are also available.
Identifier	Data set ID	
Language	---	Encoded in the version id, e.g. “en-0”; not available via API
Publisher	Namara	
Relation	---	
Rights	---	Subject to the general license terms of Namara
Source	Source	Not available from API, only from web interface
Subject	---	
Title	Title	Not available from API, only from web interface
Type	---	Not available from API, only from web interface

Most of the metadata elements are only available via the Web interface, and not via Namara’s REST API. This requires manually assembly of the metadata for each requested dataset. Namara only exchanges the data itself but not the metadata. A software solution to facilitate manual metadata assembly will be beneficial to assist operators in the conversion process. The metadata mapper from the DMA could serve as the right tool for this.

Figure 11 shows an exemplary XML output of an Amazon customer review dataset. The elements have been defined based on the information of the Namara Web interface.


```
<dc:contributor>---</dc:contributor>
<dc:coverage>---</dc:coverage>
<dc:creator>Namara</dc:creator>
<dc:date>Mar 22, 2019</dc:date>
<dc:description>Amazon Customer Reviews (a.k.a. Product Reviews) is one of Amazon’s iconic products. In a period of over two decades since the first review in 1995, millions of Amazon customers have contributed over a hundred million reviews to express opinions and describe their experiences regarding products on the Amazon.com website. Over 130+ million customer reviews are available to researchers as part of this dataset.</dc:description>
<dc:format>json</dc:format>
<dc:identifier>0a81e83f-b260-49a9-8cfb-13a1ce6e03b5</dc:identifier>
<dc:language>en</dc:language>
<dc:publisher>Namara</dc:publisher>
<dc:relation>---</dc:relation>
<dc:rights>?</dc:rights>
<dc:source>AWS</dc:source>
<dc:title>Amazon Customer Reviews Dataset</dc:title>
<dc:type>---</dc:type>
```


Figure 11: XML representation of the Dublin Core core elements for the Namara dataset “Amazon Customer Reviews Dataset”.

4 Conceptualization of an Interoperability Prototype

The interoperability solution, which is being developed in this task enables the TRUSTS platform to exchange data with external, existing DMs and thus turns TRUSTS into a DM federator instead of a mere DM. TRUSTS will use the Trusted Connector (see subsection 3.2.2). A metadata crawler harvests metadata and data assets from third party DMs and serializes them in the TRUSTS broker. The TRUSTS broker, whose development is outside of the scope of Task 3.3, is foreseen as a combination of a data catalog as well as a listing of addresses and locations of data suppliers available on TRUSTS. Potential data buyers get an overview of the data available in TRUSTS. In addition to mere metadata lookup, the prototype is also envisioned to provide functionality to interact with DMs on a higher level, e.g. to forward requests, purchase assets and forward data assets, and to make TRUSTS metadata visible to external DMs. Figure 12 gives an overview of the high-level architecture of the interoperability component.

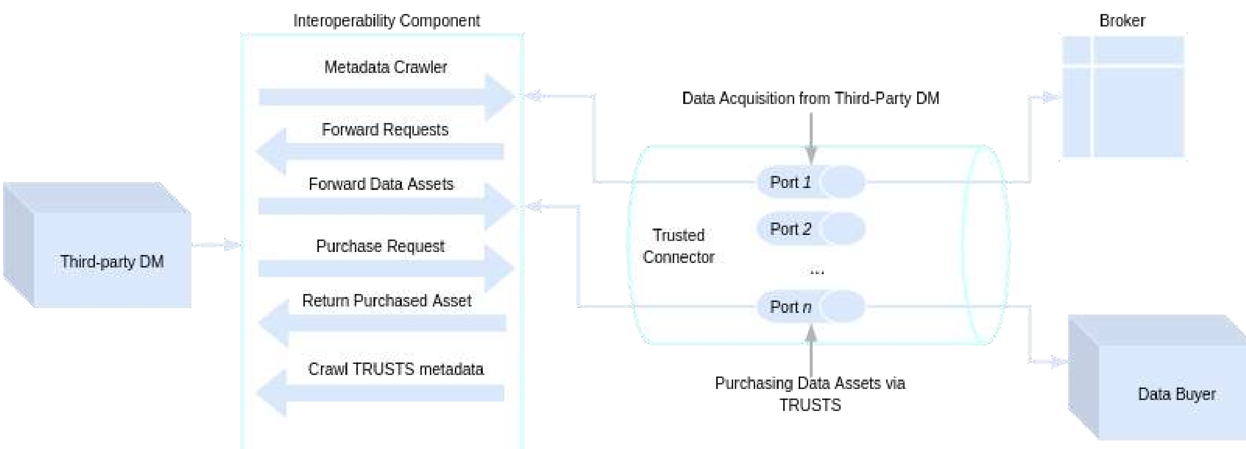

Figure 12: The architecture of the interoperability component.

The architecture of the interoperability prototype is based on the exemplary scenario for communicating with a REST API via the Trusted Connector[22], see Figure 13. Two Trusted Connectors are used in this example. The first connector receives data from an MQTT broker, which itself receives data from a mimicked temperature data from an MQTT sensor. This connector transmits the data via its core platform to the core platform of the second connector. At this location, the data is routed to an application providing a REST interface and made available using a web server. It can be accessed with a web browser.

[22]Documentation of the Trusted Connector: industrial-data-space.github.io/trusted-connector-documentation /docs/rest

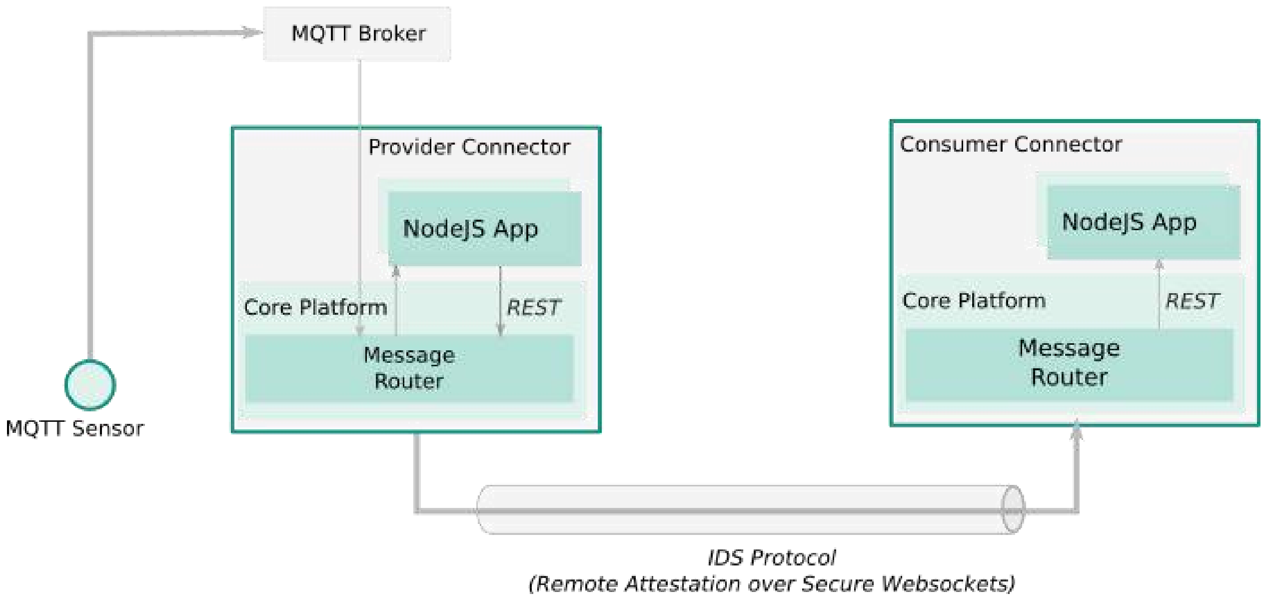

Figure 13: Message routing from an MQTT broker to a NodeJS app via the IDS Trusted Connector [4].

The interoperability solution planned in Task 3.3 will follow this principle. Certain components will be replaced with applications relevant for the interoperability task. For example, the NodeJS exposing REST endpoints will be replaced by a respective Python implementation with enhanced functionality. Additionally, the MQTT broker might eventually be replaced with a more flexible solution, especially in light that MQTT brokerage is mainly designed for usage in the area of IoT devices, which not necessarily reflects the needs of the interoperability component.

A potential use case of the interoperability component is the purchase of data assets from a third party DM via TRUSTS. Figure 14 shows a sequence diagram of this procedure. The process starts with a purchase intent of a data buyer. The buyer browses through available data assets in the TRUSTS broker and identifies an asset relevant for their data project. The relevant asset is not directly from within TRUSTS; instead, the third party DM hosts and sells the asset. The broker returns the asset id. The buyer registers the purchase in the clearinghouse, where accounting and receipts are handled. For that purpose, the clearinghouse acquires the prices applicable to the asset and stored in the TRUSTS broker. Having all the required information, the clearinghouse issues an invoice to the buyer. Once the invoice is settled, the clearinghouse triggers the purchase in the interoperability component. The interoperability component subsequently requests the purchase from the third party DM, which returns the requirements needed to settle the payment. Once the payment has been accomplished using the accounting information transmitted by the clearinghouse, the DM initiates data transfer. The interoperability component sends payment data to the clearinghouse, where the transaction is logged for future reference.

The TRUSTS platform is envisioned as having a federated architecture, where nodes and components connect in a decentralized way. In this spirit, the DM transfers the asset directly to the data buyer, without a detour over any other TRUSTS components. The direct transfer is accomplished via the Trusted Connector, which establishes a secure connection between the third-party DM and the data buyer.

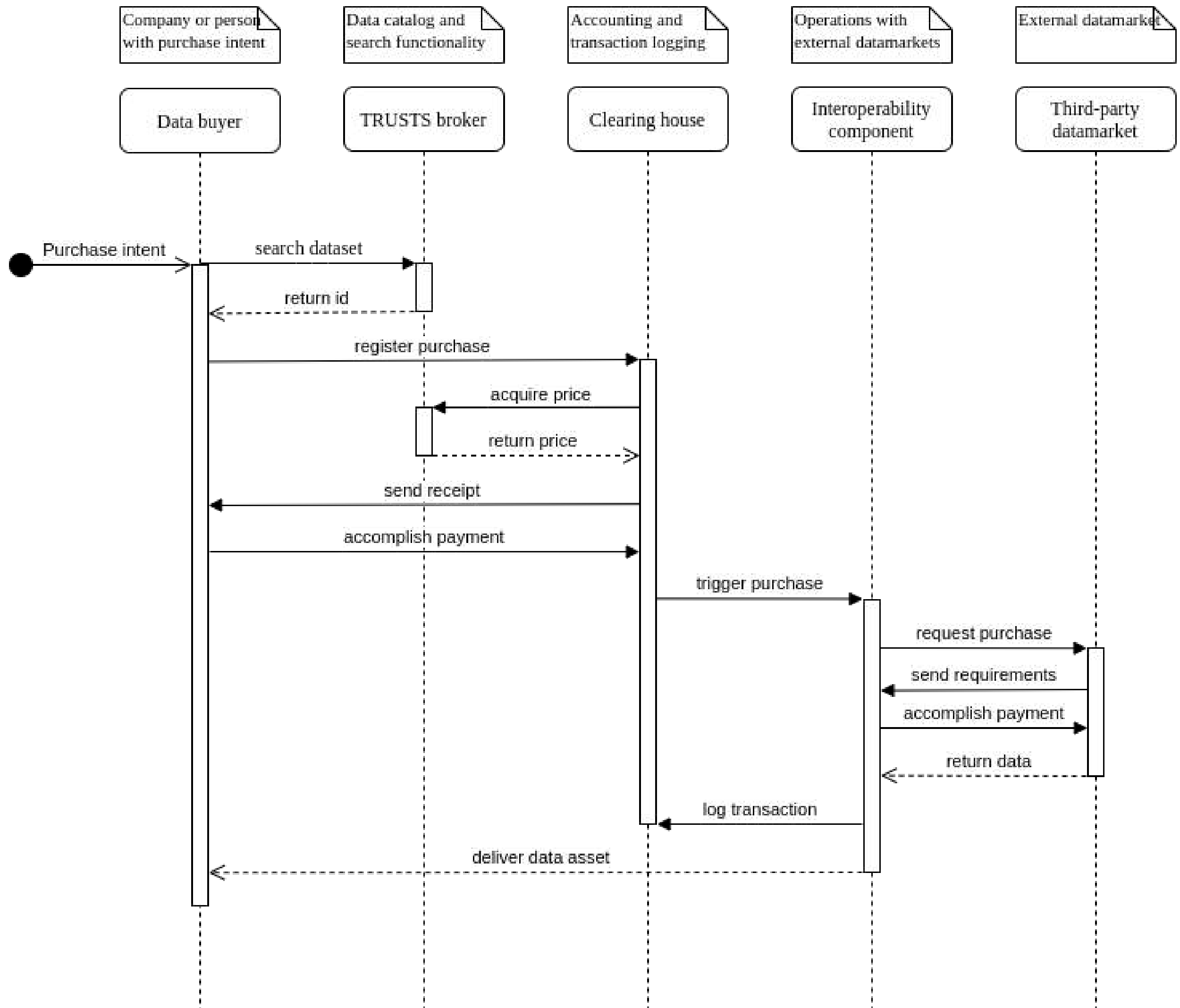

Figure 14: The sequence diagram of an exemplary purchase.

The presented workflow requires the availability of a respective API in the third-party DM. At this moment it is unclear how many DMs expose such a flexible API, or how many of them plan to do so in the future. For example, DAWEX does not operate an API. Instead, it lets data providers register their own APIs within DAWEX. Interested data buyers can request authentication keys to subsequently get direct access to the API of the provider. The respective interoperability scenario is described in the subsequent section.

5 Involvement of DM Operators

This section describes the activities within Task 3.3 aimed at involving DM operators. It describes the efforts to gather information about the technical background of DMs using a strategy consisting of an online survey and focus groups. Furthermore, we present the "Registry of Data Markets", a platform envision as the central point for information, questions, and business exchange around DMs for interested stakeholders.

5.1 DM Survey

Desktop research alone for the identification of requirements for interoperability with DMs was insufficient because of the already aforementioned lack of documented APIs and technical specifications. Thus, we decided to conduct a survey to learn more about DM's technologies. The survey consisted of an introductory part with eight general questions about the DM, a technical part with 14 questions to understand more about the technology stack used by DMs, and a business-related part consisting of six questions. The technical questions were elaborated in Task 3.3, while the business-related part was defined in WP 5. The survey questions are in Annex I.

The survey was implemented using LimeSurvey. A first attempt using Google Forms failed because of the incomplete functionality of Google Forms. The survey was launched on Nov. 9, 2020 and sent to a set of 25 DMs, listed in Table 8. Relevant DMs were selected by manual judgment of two annotators, who accomplished desktop research about the respective DM in parallel and assessed their relevance from the information available on the website. DMs with two negative assessments were excluded from the list.

Table 8: The list of DMs targeted by the survey.

Agrimetrics	IOTA	Namara
QueXopa	Mobility Data Marketplace	ThinkDataWorks
Data Intelligence Hub	oneTRANSPORT	Quandl
Advaneo	Sobloo	HealthVerity
Databroker	Caruso	Fysical
DAWEX	Farmobile	BIGToken
Streamr	BattleFin Ensemble	Datapace
Spaziodati	Kraken	

By the time of writing this deliverable, responses to the survey are still pending. However, response rate seems to be low. As a backup strategy, we decided to use an alternative approach to get feedback by DMs and TRUSTS stakeholders. Together with Task 2.1 we will organize focus groups, where we systematically interview domain experts using the Delphi method. The focus groups are planned for beginning- to mid-2021 and are currently in the conceptualization phase.

5.2 Working Group for Data Markets Interoperability

5.2.1 Introduction

For the TRUSTS project to remain successful and active beyond project lifetime, it will be necessary to build a vibrant ecosystem of participants around it. Task T3.3 will lay the foundation of this effort and provide a platform to reach out to other DMs and potential stakeholders. As part of this effort, the development of a "Registry of Data Markets" (RoD) has been initiated. The purpose of this registry is to become the central entry point for DM related questions and issues. In the first version, the registry will act as a search engine for DMs. This will also be the place to host interoperability guidelines for DMs interested in entering a relationship with TRUSTS.

As soon as the RoD is established and mature enough to be released to the public, we will actively seek contributors from industry and academia. These contributors will be incentivized to maintain and further develop the registry and to continuously fill it with content. New and emerging DMs will be entered, and the facilities for mutual exchange will be developed. The idea and vision of the RoD closely follows the concept of "OpenDOAR"[23], which is a central point of contact for open access repositories (see Figure 15).

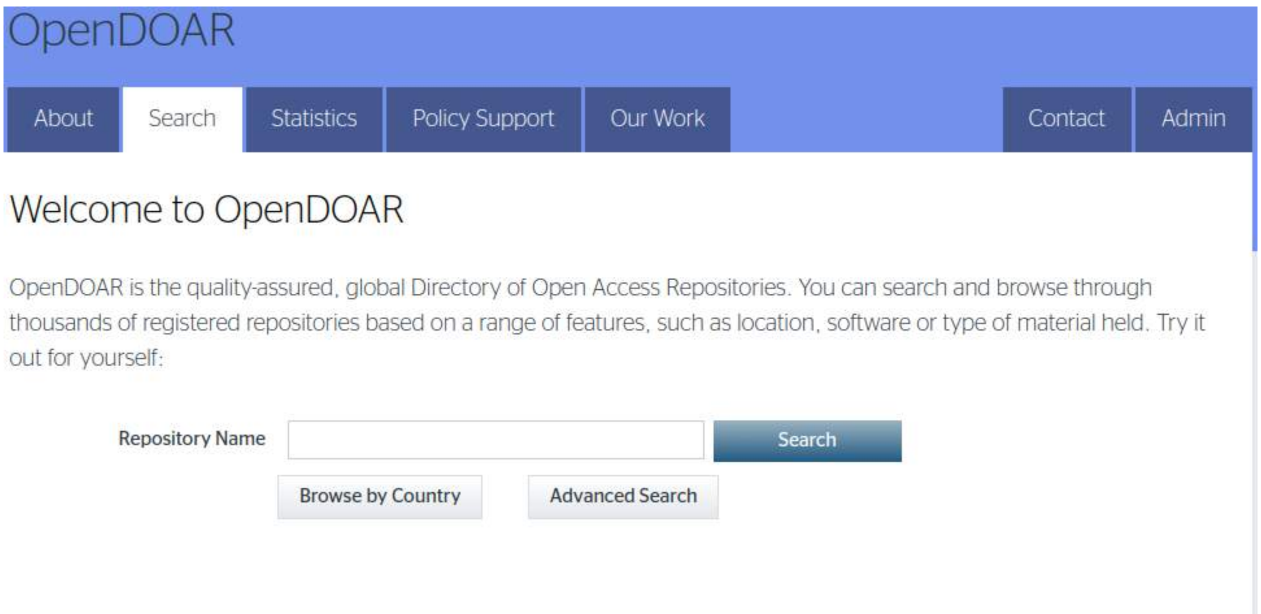

Figure 15: Visitors can search for open access repositories using OpenDOAR.

The ideal promoters for the RoD are DIO and IDSA. They have both the partner and multiplier network required to become ambassadors of the RoD and furthermore possess the marketing techniques and methods required. The current activities of IDSA have made the IDS a well-known and widely referred to concept. Similarly, DIO has undertaken strong efforts to raise awareness about data-driven technologies among Austrian companies and has built a strong network of members and multipliers to promote future innovations. While the RoD will not be implemented by DIO/IDSA, it is possible that they, together with DMs themselves, take over further technical and organizational maintenance and push forward technological innovations in the RoD.

5.2.2 Technical Specification and Current Status

The RoD is a Python Django application, i.e. a dynamic website hosting the RoD functionality. In its current state, this encompasses a prototypical set of features. There is a landing page, supposed to be

[23]OpenDOAR: v2.sherpa.ac.uk/opendoar

filled with general information about DMs and related topics. Furthermore, there is a detail page including search functionality for keyword-based search (see Figure 16). Lastly, there is also a contact page, where interested DM operators can request being included in the registry. The RoD's backend is running on a Linux server and features a PostgreSQL database.

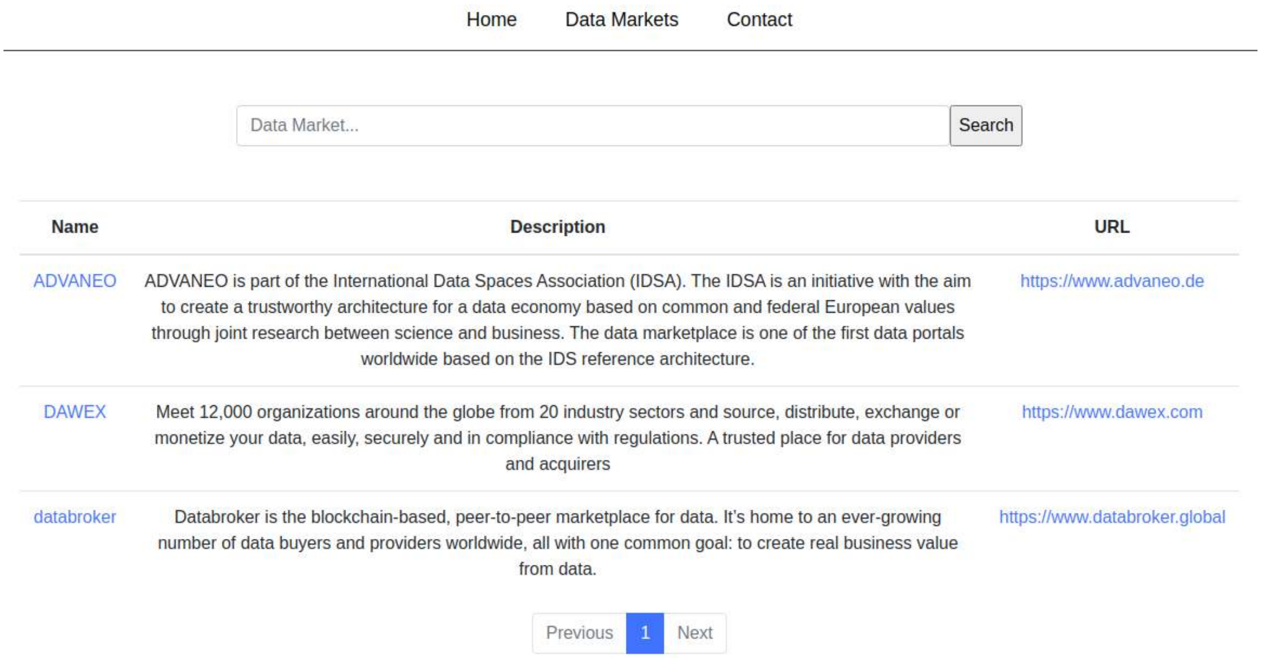

Figure 16: The search functionality of the Registry of Datamarkets.

Figure 17 shows the database schema of the RoD. In its current phase, the RoD works with six tables. The table "Data Market" is connected to the tables "Characteristic", "Country", and "Ping". "Characteristics" is used to hold information related to the taxonomy of DM business models (see [6] and Table 3). "Country" contains the name and country code of the DM's country of origin. "Ping" is in use by a sub-module of the RoD scanning for the availability of registered DMs. This sub-module sends requests to DMs in regular intervals ("pings" them). In case a DM does not respond, an alert is triggered upon which manual evaluation will verify if the DM actually ceased to exist or if it is just temporarily down or has moved to a new location.

The tables "Characteristics", "Dimension", and "Domain" map the taxonomy of business models (see [6] and Table 3) to the relational database schema. "Domain" and "Dimension" have a one-to-many relation, i.e. the same domain can have multiple dimensions, at least one. Similarly, "Dimension" and "Characteristics" have a one-to-many relation, i.e. the same dimension can have multiple characteristics, again at least one.

The RoD will undergo further development within the next months. The database schema might expand or change during this time. The updated and final schema of the database will be available in the next deliverables.

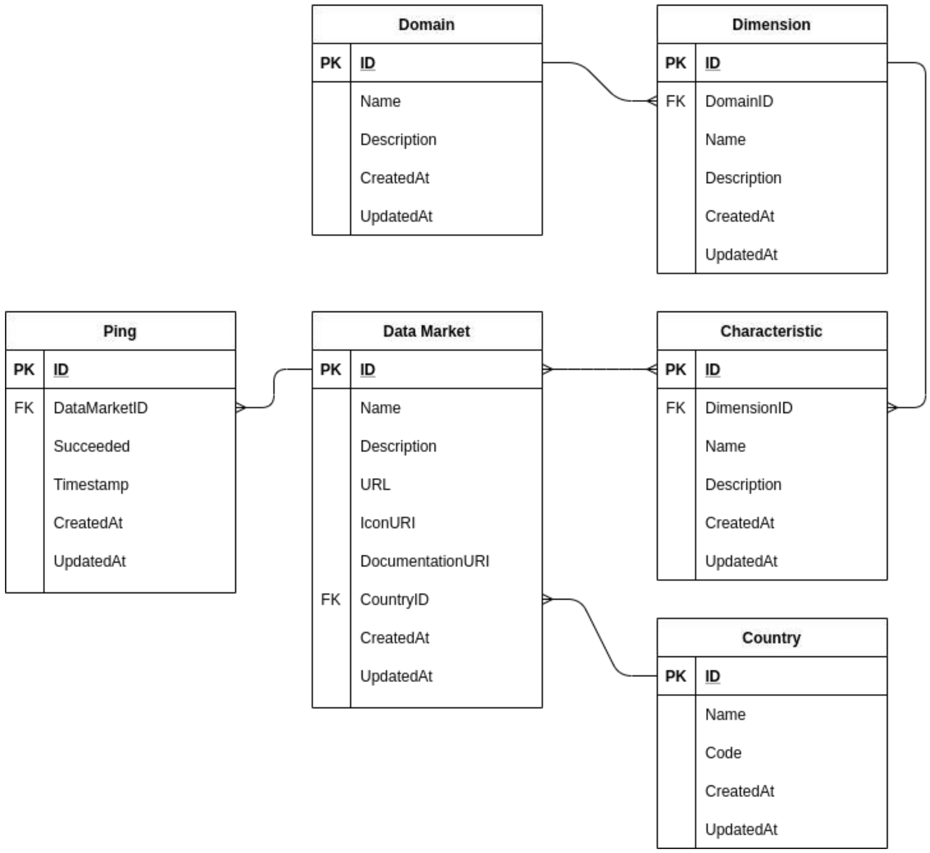

Figure 17: The RoD database schema.

5.2.3 Registering Data Markets in the RoD

In the initial phase, where general public attention about the RoD is presumably low, the RoD will get populated manually from the DMs listed in [6]. Later, when promotion of the RoD has increased awareness among DM operators, they themselves can request to join the RoD by sending a formal request via the platform. Upon manual verification, the DM operator will get access credentials and can subsequently upload their DM specific information.

5.2.4 Provision of Machine-readable Interfaces

The RoD will feature machine-readable interfaces in the form of a REST API. The purpose is to allow DMs to register and update their information in the RoD. Furthermore, it will allow them to extract information about other DMs from the RoD to learn more about their domains from the provided information. OAuth2 will serve as the authentication mechanism to operate with the REST API.

6 Conclusions and Next Actions

This deliverable summarizes the activities in Task 3.3. “Data marketplaces interoperability solutions” and is the first of a series of three reports. It documents the investigations of external data markets concerning interoperability with TRUSTS, software prototype development to examine the technical specifications of DM interfaces, and a potential architecture for an interoperability component. Furthermore, it describes the initiation of the registry of DMs, a repository of DMs to be used in academia and industry alike, serving as a central point for questions related to DMs and as an overview platform of existing DMs.

The deliverable summarizes the status of investigations of technical features of DMs with regard to interoperability with TRUSTS. One result identified through this investigation is that DMs very often do not foresee any means of machine-interoperability. Instead, they provide data from their web interface. This would require a manual download of datasets before they could get included in the TRUSTS broker and data catalog. Task 3.3 will approach this situation in two ways: on the one hand, it will develop interoperability components, i.e. software libraries, for two DMs where an API is available. These interoperability components will implement a set of functionality these DMs provide and make them available from within TRUSTS. The status of the architecture is described in this deliverable. On the other hand, Task 3.3 will elaborate on best practices and develop guidelines for DMs. These guidelines will help DMs interested in interoperation with TRUSTS on how to proceed when they develop their own interoperability solutions to participate in the TRUSTS DM federator. In other words, the guidelines will document how to develop software libraries that are capable of exposing and exchanging data via the TRUSTS platform. The deliverable also describes the components that are currently envisioned as part of the interoperability component. This includes the Trusted Connector, an MQTT broker, a metadata crawler to acquire metadata from third party DMs as well as a broker format converter, transforming the incoming data from a DM into a format compliant with the TRUSTS broker metadata schema.

A set of software prototypes in the form of Google Colaboratory notebooks has been developed to investigate the range of functionality of third party DMs. The deliverable describes these notebooks and summarizes the output generated by them. Development of these prototypes was necessary to understand the technical requirements of the APIs and the specifications of the data delivered by the platform, as documentations of platforms are often unclear about technical details. This knowledge will help to conceptualize, design, and implement the metadata crawler and the broker metadata convert.

Turning TRUSTS into a successful entity beyond project lifetime is a key aspect of the project. Task 3.3 will contribute to this effort by creating a registry of DMs, listing existing DMs, providing background information on the concept of DMs in general and the specific characteristics of existing DMs, as well as providing a manual for DMs to initiate interoperation with TRUSTS. The ultimate goal of the RoD is to establish the basis for the creation of a vibrant community for the data sharing economy. This deliverable describes the goal of the RoD, its technical foundation as well as the status of the prototype, which will be turned into a fully functional platform to register, identify, and look up DMs within the next months.

Next steps include the investigation of further DMs with software prototypes and the conduction of focus groups with DM operators and TRUSTS stakeholders. The knowledge gained through this process will help to refine the interoperability solution. The implementation of this technical component is the major goal of the next steps. It will give TRUSTS an interface for current and emerging DMs with an intention to participate in a data ecosystem. The interoperability solution will be based on the requirements identified in business-related and legal WPs. Interested DMs will be able to connect through this interface and exchange and trade their data assets with the community in TRUSTS.

7 References

[1] Stahl, F., Schomm, F., Vossen, G. (2016). A Classification Framework for Data Marketplaces. Vietnam J Comput Sci 3, 137–143. https://doi.org/10.1007/s40595-016-0064-2

[2] Meisel, L. & Spiekermann, M. (2019). Datenmarktplätze - Plattformen für Datenaustausch und Datenmonetarisierung in der Data Economy. ISST Report, Fraunhofer Institute for Software and Systems Engineering ISST.

[3] Otto, B., Lis, D., Jürjens, J., Cirullies, J., Opriel, S., Howar, F., Meister, S., Spiekermann, M., Pettenpohl, H. & Möller, F. (2019). Data Ecosystems - Conceptual Foundations, Constituents, and Recommendations. ISST Report, Fraunhofer Institute for Software and Systems Engineering ISST.

[4] Connecting REST Apps using the IDS. https://industrial-data-space.github.io/trusted-connector-documentation/docs/rest/. Last accessed: Oct 28, 2020.

[5] Amorim, R.C., Castro, J.A., Rocha da Silva, J., Riberio, C. (2016). A comparison of research data management platforms: architecture, flexible metadata and interoperability. Universal Access in the Information Society, 16, 851–862 (2017). https://doi.org/10.1007/s10209-016-0475-y

[6] van de Ven, M.R. (2020). Creating a Taxonomy of Business Models for Data Marketplaces. Master Thesis, TU Delft Technology, Policy and Management.

[7] Otto, B., Steinbuß, S, Teuscher, A. & Lohmann, S. (2019). IDS Reference Architecture Model, v3.0. International Data Spaces Association.

[8] Schütte, J., Brost, G. & Wessel, S. (2018). Datensouveränität im Internet der Dinge – Der Trusted Connector im Industrial Data Space. Whitepaper, Fraunhofer AISEC.

Annex I: The Technology-related Questions of the DM Survey

Part 3/3 - Technology

The goal of the technology part of our survey is to explore technologies used in existing data platforms. This will guide the design of an architecture that optimizes interoperability with a wide variety of data platforms and the creation of data ecosystem catering to the needs of a wide variety of data and services.

For the purpose of this survey, we will use the term data product to refer to datasets, data services or other data sources that can be exchanged or offered.

Building a data market is a complex problem. Which technology stack are you building upon?

Data management platforms

Programming languages

Databases

Search Engines

Other

Do you use REST APIs to allow automatic exchange of data or external services with customers?

Yes	No	No answer

What type of automatic exchange of data with external services or customers do you use REST APIs for?

Comment only when you choose an answer.

- [ ] For searching through existing datasets
- [ ] For accessing the data
- [ ] For publishing new datasets
- [ ] For buying and/or selling datasets

Other:

What data would you consider to share with 3rd parties via APIs?

	Anonymous Users	Authenticated Users	Not sure	No answer

	Anonymous Users	Authenticated Users	Not sure	No answer
Basic metadata of data products (name, description, etc.)	○	○	○	○
Data products	○	○	○	○
Pricing information of data products	○	○	○	○
Licencing information of data products	○	○	○	○
Size of data product	○	○	○	○
Limitations of the data product (e.g. missing values or other quality measures)	○	○	○	○
Reviews and / or reputation scores of data products	○	○	○	○

Do you use container technology in your technology stack?

✔ Yes	⊘ No	○ No answer

Does your data platform offer data processing services?

✔ Yes	⊘ No	○ No answer

Where do the offered data processing services reside?

❶ Comment only when you choose an answer.

☐ We offer / broker services running on our own infrastructure

- [ ] We offer / broker services running on the buyer's infrastructure
- [ ] We offer / broker services running on the seller's infrastructure
- [ ] We offer / broker services running on third-party infrastructure

Other:

Does your data market allow for the exchange of private or sensitive data?

Yes	No	No answer

Are you aware of the IDS Trusted Connector framework?

Yes	No	No answer

Do you think that it would be easy for you to adopt the IDS Trusted Connector in your technology stack?

Choose one of the following answers

- The IDS Trusted Connector is already part of our technology stack
- Deployment of the IDS Trusted Connector is under active consideration
- Deployment of the IDS Trusted Connector is not considered but it would be easy
- Deployment of the IDS Trusted Connector is not considered
- Not sure

No answer

Please enter your comment here:

Can your data platform easily adopt existing standards for exposing information about data products, e.g. OAI-PMH?

Yes	No	No answer

Which other established standards do you use in your data platform?

Distributed Ledger Technology (blockchain) is often mentioned in the context of data platforms. Do you use this technology in your data platform?

Yes	No	No answer

What do you use Distributed Ledger Technology for, and what is your experience?

Is your data platform solution open source?

Yes

No

No answer

Please could you provide a link to the source code?