Data Platform for the Connection of Cognitive Ports

Title:	Document Version:
D2.2 Scalability, Interoperability and Definition Standards	1.2

Project Number:	Project Acronym:	Project Title:
H2020-871493	DataPorts	A Data Platform for the Cognitive Ports of the Future

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type*-Security*:
M24 (December 2021)	M24 (December 2021)	R-PU

*Type: P: Prototype; R: Report; D: Demonstrator; O: Other; ORDP: Open Research Data Pilot; E: Ethics.

**Security Class: PU: Public; PP: Restricted to other programme participants (including the Commission); RE: Restricted to a group defined by the consortium (including the Commission); CO: Confidential, only for members of the consortium (including the Commission).

Responsible:	Organisation:	Contributing WP:
Lucas Moulin	TRX	WP2

Authors (organisation):	
Andreu Belsa Pellicer (UPV)	Lucas Moulin (TRX)
Matilde Julián Seguí (UPV)	Reda Khouani (TRX)
Achilleas Marinakis (ICCS)	Hanane Becha (TRX)
Amir Shayan Ahmadian (UKL)	Daniel Garcia Latorre (EVR)
Alexandros Zerzelidis (CERTH)	

Abstract:

This deliverable is a report describing the need for scalability and interoperability in the seaport context. It emphasizes the need for standards to reach the required level of performance and availability and details the best practices adopted for DataPorts.

Keywords:

Scalability, Interoperability, Standardization, Industry 4.0, Data for AI, Port Authorities

Revision History

Revision	Date	Description	Author (Organisation)
V0.1	16.07.2020	First version of the document.	Réda Khouani (TRX)
V0.2	25.10.2021	Compiling all inputs from partners and the modified structure in the updated document.	Lucas Moulin (TRX)
V0.3	29.10.2021	Modified template of the document.	Lucas Moulin (TRX)
V0.4	17.11.2021	Inputs from ICCS (scalability) and UPV (semantic interoperability).	Achilleas Marinakis (ICCS) Matilde Julián Seguí (UPV) Andreu Belsa Pellicer (UPV)
V0.5	23.11.2021	Addition of data models description and cleaning of references.	Lucas Moulin (TRX)
V1.0	03.12.2021	Addition of Blockchain scalability content, improving some paragraphs and completing acronyms.	Lucas Moulin (TRX)
V1.1	17.12.2021	Review from EVR and CERTH.	Daniel Garcia Latorre (EVR) Alexandros Zerzelidis (CERTH)
V1.2	20.12.2021	Review from project coordinator.	Santiago Cáceres (ITI) Lucas Moulin (TRX) Achilleas Marinakis (ICCS)

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement № 871493.

More information available at https://DataPorts-project.eu

Copyright Statement

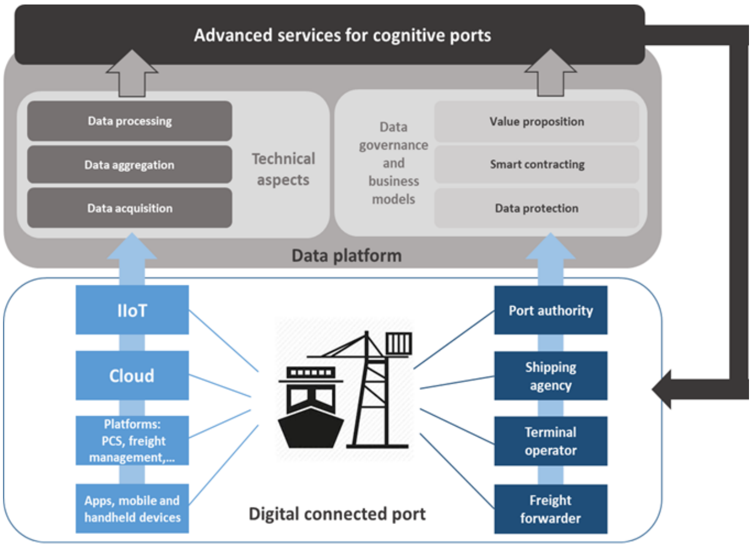
INDEX

1 INTRODUCTION

1.1 DATAPORTS PROJECT OVERVIEW

DataPorts is a project funded by the European Commission as part of the H2020 Big Data Value PPP programme, and coordinated by the ITI - Technological Institute of Informatics. DataPorts relies on the participation of 13 partners from five different nationalities. The project involves the design and implementation of a data platform, its deployment in two relevant European seaports connecting to their existing digital infrastructures and addressing specific local constraints. Furthermore, a global use case involving these two ports and other actors and targeting inter-port objectives, and all the actions to foster the adoption of the platform at European level.

Hundreds of different European seaports collaborate with each other, exchanging different digital data from several data sources. However, to achieve efficient collaboration and benefit from AI-based technology, a new integrating environment is needed. To this end, DataPorts project is designing and implementing an Industrial Data Platform.



The DataPorts Platform aim is to connect to the different digital infrastructures currently existing in digital seaports, enabling the interconnection of a wide variety of systems into a tightly integrated ecosystem. In addition, to set the policies for a trusted and reliable data sharing and trading based on data owners' rules and offering a clear value proposition. Finally, to leverage on the data collected to provide advanced Data Analytic services based on which the different actors in the port value chain could develop novel AI and cognitive applications.

DataPorts will allow establish a future Data Space unique for all maritime ports of Europe and contribute to the EC global objective of creating a Common European Data Space.

1.2 DELIVERABLE PURPOSE AND SCOPE

Specifically, the DOA states the following regarding this Deliverable:

This document will include: i) requirements for the realisation of a scalable data platform for cognitive ports and its related architecture, ii) the data flow exchange formats that serves as interfaces between the different services to simplify their orchestration to create a federated platform as a single entry point, and iii) the updated UN/CEFACT Core Components Library that covers all the definitions of the terms used in support of the cognitive ports platform.

In order to be consistent with the other deliverables in the project, the present document neither expresses detailed architecture requirements nor details the design of the components that allow the exchange of data flows. However, it breaks down the challenges for the realisation of a scalable data platform with the associated concepts allowing interoperability through the use of standards.

1.3 DELIVERABLE CONTEXT

Its relationship to other documents is as follows:

Primary Preceding documents:

- Deliverable 2.1 “Industrial Data Platforms and seaport community requirements and challenges” March 2021, which details the requirements of the seaport community regarding the main axes of Deliverable 2.2
- Deliverable 3.1 “Data access interfaces” June 2021, which details The Data Access Component enabling access to the different data sources in a secure way and thus key to interoperability
- Deliverable 3.2 “Data processing services” June 2021, which details The Semantic Interoperability Component providing a unified API to access the data from the different data sources connected to the DataPorts platform

Primary Dependant documents:

- Deliverable 2.4 “Platform architecture and specifications” December 2021, which details the components implementing the concepts introduced in Deliverable 2.2

1.4 DOCUMENT STRUCTURE

This deliverable is broken down into the following sections:

- **Section 2** links the challenges of seaports to the requirements for scalability, interoperability and standardization
- **Section 3** presents the scalability advantages of decentralized and serverless architectures
- **Section 4** details the semantic interoperability concepts and the interfaces exposed within the components adopted by DataPorts that will enable full scalability
- **Section 5** presents the standard data models that will allow DataPorts to realize the required interoperability
- **Section 6** concludes with the business challenges that DataPorts and other systems face regarding scalability, interoperability and standardization.

2 CHALLENGES FOR PORT-ORIENTED COGNITIVE SERVICES

The digital transformation of the ports towards the Fourth Industrial Revolution is revealing opportunities related to the add-on services that can be provided. Port platforms are now integrating available data sources, capturing the potential needs that arise from the increased demand for more accurate and complete information [16]. The value of that new trend is also boosted by the new wave of start-ups that implement over-the-top services. However, most port authorities are not technologically ready to host these services and frameworks. The necessary technology infrastructure has diametrically opposite requirements to the one that is deployed. The same drawback applies to the architecture where in most cases must be shifted to serverless oriented solutions.

For example, port authorities have the chance to enrich their services with the use of smart container infrastructure. The addition of this type of IoT infrastructure enables the information about the entire journey and conditions of the cargo to be exchanged on-line, directly to the rest of the supply chain without human intervention. This provides greater visibility to the stakeholders within the transaction, as well as to regulatory agencies who need detailed information on the consignments before they arrive at the border. This technology can be combined with other innovations such as blockchain, big data or data pipelines to provide even more facilitation to the trading community. In all these cases, though, we see that creating clear, unambiguous message exchange standards will allow to capitalize the full potential of the enhanced data. That shared data will enable the creation of new value-added services that the port authorities can benefit from.

A cognitive service is a software component that uses AI and Big Data capabilities to solve a business task. Cognitive services are ready-to-use solutions to be integrated in the context of software products for improving the decision-making process related with data. Some cloud providers offer general cognitive services such as image classification or natural text recognition / translation. However, this novel paradigm has not been applied in the ports domain. Ports share common business tasks in which cognitive services could provide answers. Some examples are the prediction of the Expected Time of Arrival (ETA) of a container or a vessel, the truck turnaround time to deliver a container to a terminal, or the definition of a booking system to reduce environmental impact. The main goal is to build such services to be generic enough to be applied in different ports and use cases. To enable the cognitive services approach envisioned in the context of the project, the DataPorts platform must address the following technical challenges:

- One issue to address is to enable the data sharing between an undefined number of port stakeholders, such as terminal operators, port authorities, logistic carriers and so on. The accuracy of cognitive services is directly related with the amount of available data. For building such a data ecosystem, the defined architecture must include scalability as its first design principle. To address this challenge, we foresee the introduction of the International Data Spaces Association (IDSA) Reference Architecture Model [40], a standard solution with the required building blocks for achieving a seamless integration between organizations.
- Scalability for AI Training: Training models for cognitive services is a time-consuming task even with a powerful computing infrastructure for supporting it. As the state of the art evolves, new frameworks and techniques must be tested in order to find the optimal one, that which provides the most accurate results for the problem at hand. A cognitive service vision implies that no specific know-how from the data science domain is required: the end-user is the one that defines the training process, with little manual intervention. This fact leads to the definition of several training alternatives, which must run simultaneously over a distributed infrastructure. This challenge will be addressed by the DataPorts platform, introducing the most suitable technological approaches from the Machine Learning (ML) devops area.
- Heterogeneous data processing: The seaports domain is one in which several IT infrastructures, information systems like TOS or IoT sensing devices, are potential candidates to become a valuable data source. In this scenario, two main challenges arise: how to deal with the heterogeneity of data

sources (formats, schema, etc.) and with an undefined volume of data. The DataPorts platform will support the heterogeneity of schemas, applying techniques from the semantic interoperability domain and considering vocabularies or taxonomies from standardization bodies. Following the good practices for Big Data processing, such as the use of containerized application and distributed databases, we expect to provide the required tools for enabling a scalable data processing.

- A trusted data governance: Ownership of the data is a key issue in any discussion related with data sharing between different organizations. To enhance the data sharing needed to build cognitive services, the DataPorts platform must first provide a trusted framework for defining data-sharing rules to specific users, roles and organizations. This framework must also enforce that the data is used following the specifications the data owner has formally defined. Data management, when this data is outside the boundaries of the organization, is a challenge which requires a set of trusted software components and clear security procedures. We foresee the use of Smart Contracts, in the context of a blockchain network among organizations, as the technological foundation to address this challenge.

The next sections introduce in more detail how we expect to address these overall challenges in terms of scalability, interoperability and standardization.

3 SCALABILITY

In order to define a programming model and architecture where small code snippets are executed in the cloud without any control over the resources on which the code runs, the industry came up with the term "Serverless Computing" [3]. It is by no means an indication that no servers exist, simply that the developer should leave most operational issues to the cloud provider, such as resource provisioning, monitoring, maintenance, scalability, and fault tolerance. The platform must guarantee the scalability and elasticity of the functionalities of the platform (see D2.4 "Platform Architecture and Specifications"). This means proactively provisioning resources to cater both for actual load and in anticipation of potential load. This issue is one of the more difficult to tackle in a serverless computing environment, because these forecasts and provisioning decisions must be made with little to no application-level information. For instance, as an indicator of the load, the system can use request queue lengths, but is blind to the nature of these requests.

A serverless architecture is based on the premise that the deployment is a transparent process, where the developer is not aware of the cluster/server that the stateless functions are deployed on. Although the inability to specify where the functions should run seems to weaken the overall architecture, apparently producing a kind of lack of control, that is compensated by the benefit of performance of the application.

In a few words, serverless computing allows application developers to decompose large applications into small functions, allowing application components to scale individually [25]. There exist many implementations regarding scaling and auto-scaling on serverless functions. However, most of those systems use Kubernetes' built-in Horizontal Pod Autoscaling (HPA) for auto-scaling, which implements compute-resource-dependent auto-scaling of function instances [4]. Kubernetes is a system for deploying applications and utilizing more efficiently the containerized infrastructure that powers the apps. In fact, running containers is not enough, the system also needs to be able to:

- Integrate and orchestrate these modular parts
- Scale up and scale down on demand
- Ensure fault tolerance
- Provide communication across a cluster

More specifically, the adoption of Kubernetes aims at enhancing the performance of the platform towards four dimensions:

1. Scaling:
    - Horizontal infrastructure scaling: new servers can be added or removed easily
    - Auto-scaling: automatically change the number of running containers, based on CPU utilization or other application provided metrics
    - Manual scaling: manually scale the number of running containers through a command or the interface
2. Availability:
    - The system is designed from the ground up to be fault tolerant and handle if parts fail
    - If one worker node goes down, app instances on available worker nodes continue to run
    - Rolling updates to change the software without downtime
3. Automation:
    - Health checks and self-healing: offers self-healing and auto replacement if a container or pod fails
    - Traffic routing and load balancing: responds to outages or periods of high traffic
    - Automated rollouts and rollbacks: in case the rollout is not successful, the system automatically rolls back

4. Portability:
    - It can run on a public or a private cloud
    - It can run on premises or in a hybrid environment
    - It is possible to move a Kubernetes cluster from one hosting vendor to another, without changing (almost) any of the deployment and management processes

Scalability is the ability of a system to handle a growing amount of work by adding resources to itself. This means that scalability stands as a direct solution to any workload issue that might emerge in a system. Therefore, today's frameworks should take a full advantage of scalability's benefits by implementing tools that achieve exactly that. Solutions vary on how a system can be scalable. However, the selection field narrows down a lot when it comes to serverless computing systems. That is because of the nature of serverless applications, something we touched upon earlier, due to which they are created to have the scalability issue solved in-advance. What remains to be answered is how we can further improve scalability in the serverless world.

A solution to the scalability issue, distinct from most of the available ones, is through the prism of a Data-as-a-Service Marketplace [37]. When combined with a fully working Function as a service (FaaS) platform, this approach can lead to optimum scaling results. The core idea is about creating a serverless platform as part of a Data as a Service (DaaS) marketplace repository framework, that enables dynamic scaling in order to ensure business continuity, such as real time accommodation of rapidly evolving user numbers, and fault tolerance. That framework contains a multitude of readily accessible APIs to serve the needs of a growing and changing DaaS platform marketplace, while it provides great flexibility in selecting topologies and architectures for the storage pool. In essence, any node, either located in the cloud, or at the physical location of the marketplace, or even at the edge of the network, may be used to store data as part of the repository cluster. That DaaS strategy uses the cloud to deliver data storage, analytics services and processing orchestration tools in order to offer data in a manner that new-age applications can use. Moreover, for such kind of marketplace repository frameworks to operate efficiently, there is a need to be integrated with an evaluation and monitoring system. The goal of the latter would be to provide real-time metrics and evaluation ratings, concerning both the quality of the provided services as well as the resources utilization of the machines. To this end, indicative measurements like response time, throughput, latency, CPU and/or Random Access Memory usage could assist the decision making system of the framework, regarding when to scale up or down, in order to satisfy specific requirements and constraints, which are expressed by the applications designers or the end-users.

3.1 SCALABILITY ENSURED BY INTERNATIONAL DATA SPACES REFERENCE ARCHITECTURE MODEL

The adopted proposal in the context of the DataPorts project is the International Data Spaces (IDS) Reference Architecture Model [40]. According to the official documentation, the IDS is a virtual data space, leveraging existing standards and technologies, as well as governance models well-accepted in the data economy, in order to facilitate secure and standardized data exchange/linkage in a trusted business ecosystem. Therefore, it provides a basis for creating smart-service scenarios and facilitating innovative cross-company business processes. At the same time, it guarantees data sovereignty for data owners. In short, the IDS reference architecture model's high scalability is attributable to the fact that this model is a decentralized architecture ("peer-to-peer" data exchange with redundant replicated connectors and brokers) without a central bottleneck. This proposal then sets the foundation for the implementation of a scalable platform.

3.2 SPECIFIC BLOCKCHAIN SCALABILITY CONSIDERATIONS

As the blockchain technology is a large part of DataPorts architecture, it is important to consider specifically the scalability of this component.

The performance of a blockchain component can be affected by many variables, such as transaction size, block size, network size, as well as hardware limits. We could divide the elements that can impact the performance of the blockchain network into four layers:

- Infrastructure: the set of machines and physical resources on which the Hyperledger Fabric network is deployed.
- Hyperledger Fabric network architecture: the configuration of the number and type of nodes, and the rest of the elements that make up the Fabric network.
- Fabric protocol: the core of how a fabric network works.
- Chaincodes: the smart contracts in which distributed business logic is implemented.

Each of these layers affects the performance of a Hyperledger Fabric-based system in different ways.

The fundamental element of the infrastructure that affects the performance of Fabric is the allocation of Central Processing Units (CPU), therefore, for the network to work at its maximum performance, it must have enough CPUs.

Unfortunately, indefinitely increasing the CPUs does not increase the number of transactions per second, so a correct architecture design allows to optimize performance. There are two possible approaches: i) scaling the number of "endorser peers", and ii) optimally increasing the number of channels, both, of course, taken in conjunction with an optimal CPU allocation.

Another factor that impacts at the protocol level is the ideal block size and time settings based on the expected transaction load. It is important to correctly specify the size of these parameters.

Finally, a correct design and implementation of the chaincodes, avoiding Multi-Version Concurrency Control (MVCC) conflicts. These conflicts can occur when the network is under heavy load and multiple transactions try to modify the same information in the ledger at the same time.

4 INTEROPERABILITY

4.1 INTRODUCTION

Interoperability among different computer systems is the ability to share services and data between them. Since each software solution provides its own infrastructure, devices, APIs and data formats, achieving interoperability entails having effective communication and coordination on multiple levels between those components and systems that together compose a uniform platform at a larger scale. In order to be interoperable, two or more systems must be able to exchange, interpret, and present shared data in a way that is understood by all.

There are two types of data interoperability: syntactic interoperability and semantic interoperability. Syntactic interoperability is based on the adoption of a common data format and communication protocol and is a prerequisite to semantic interoperability. Semantic interoperability refers to the ability of computer systems to exchange data with a shared and unambiguous meaning [18].

This section focuses on interoperability from the point of view of Semantic Interoperability and Application Programming Interfaces (APIs). On the one hand, Semantic Interoperability is based on the use of an ontology model that ensures that the requester and provider have a common understanding of the services and data being exchanged. On the other hand, APIs document all the available services that are exposed by a software system, as well as the information about the respective communication protocols. Therefore, APIs are considered a significant interoperability tool that allows the standardization of the components' communication. The present section describes the evolution of the models based on the ever-changing needs of communication, as well as the most powerful state-of-the-art tool for API standardization.

4.2 SEMANTIC INTEROPERABILITY

For every system where interoperability is a requirement, it is essential to consider the production, collection, transmission, and processing of vast amounts of data. An application consuming those data needs to understand their structure and meaning. These aspects are represented in the metadata, which provides a semantic description of the data and can be utilized in many ways, such as resource discovery, management, and access control [17]. The metadata provides a semantic description by linking each data element to a controlled, shared vocabulary in a way that is machine readable, thus providing a shared meaning for the data. Within this shared vocabulary are links to an ontology that represents a set of concepts within a domain and the relationships among those concepts. The more expressive the language used for representing the metadata is, the more accurate the description might become.

An ontology can be defined as a structure that provides a vocabulary for a domain of interest, together with the meaning of entities present in that vocabulary. Typically, the entities in an ontology may be grouped, put into a hierarchy, related with each other, and subdivided according to different notions of similarity. Ontologies provide the ability to share a common understanding of the domain, to make its assumptions explicit, and to analyse and reuse the domain knowledge. In order to achieve a shared meaning of data, the platforms or systems must use a common ontology either explicitly, or implicitly, for example, via a semantic mediator [14].

The development of an ontology for logistics is not a trivial task. Typically, organizations in transportation and logistics, with particular focus on port logistics, have their own local standards, which may have a poor formalization of semantics, or not have explicit semantics at all [15]. Thus, it is necessary to define and use guidelines and best practices in this domain. To implement precise and consistent solutions, a proper theoretical and methodological support for ontology engineering is required, in addition to providing solutions to practical issues, to be close to the actual needs of the market [9].

The DataPorts project is developing a semantic framework for describing port data, together with mappings to standard vocabularies, in order to simplify the reuse of data applications for analytics and forecasting. This semantic framework will represent the domain knowledge in a systematic and standardized way, thus enabling the reuse and exploitation of the data by the experts directly, thereby empowering the construction of cognitive port applications.

The relevance of the use of data platforms and the exploitation of data sharing is boosted by the high volume of different companies and public bodies that need to collaborate among them with different degrees of digital capacities. In this aspect, a semantic interoperability framework with a global ontology designed for the application domain of interest will improve such collaboration and data representation. Regarding the target group of users from the logistics domain, the DataPorts semantic framework makes the discovery and interpretation of data and metadata more manageable for data users.

From a technical point of view, the first step for the definition of the semantic framework has been the identification of the different data sources to be integrated in the DataPorts platform, including the data storage and the mechanisms to facilitate data management. The results of this step have been reported in deliverable D5.1. The next steps involve the definition of ontologies, mechanisms and enablers to provide semantic interoperability with the digital infrastructures of the port, including IoT devices and mobile apps, as well as legacy databases and systems. Finally, the last step is the development of the semantic-based components and tools needed to interact and manage the information of these data sources through the DataPorts platform. The data platform will guarantee semantic interoperability, to provide a unified way to access and understand the data, so that it can be used by the different data consumers and the data analytic services. Figure 2 shows the architecture designed to achieve this purpose. The architecture of the platform is explained in detail in deliverable D2.4 “Platform architecture and specifications”.

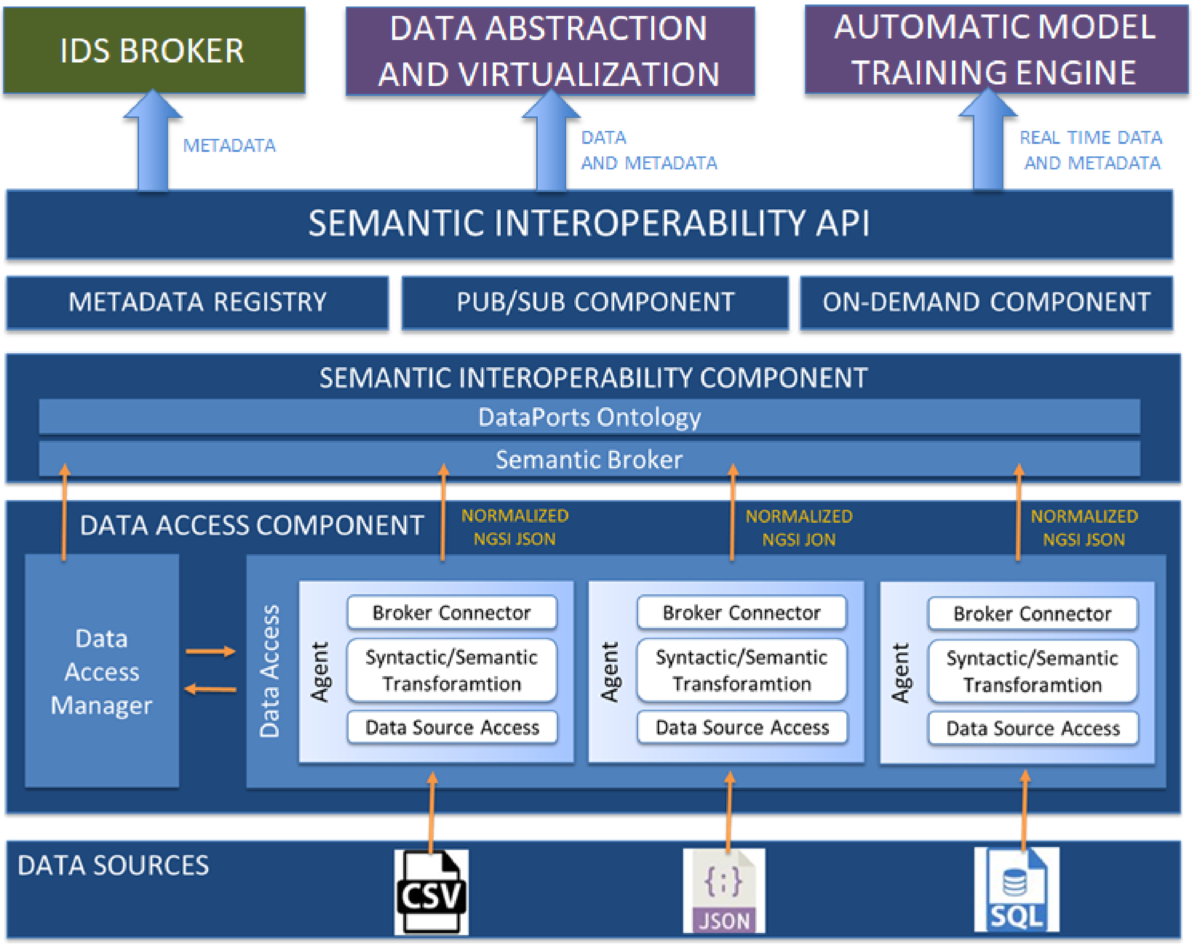


Figure 2 Data Access and Semantic Interoperability Layer components of DataPorts Platform

The Data Access Component (which is described in detail in deliverable D3.1) enables access to the different data sources in a secure way. Given the variety of data sources that could be connected to the platform, it is necessary to analyse the interfaces, data formats and data models of each data source, in order to access and understand their data. To deal with these heterogeneous data sources, a data access agent is necessary for each data source integrated in the DataPorts platform. An agent is the piece of software in that acquires data from a source under certain conditions. Then, the agents transform this data to the DataPorts common data model. Finally, this data is sent to the other platform components. In addition, the Data Access Manager manages the metadata description of the data processed by the agents and the interaction of the agents with the other platform components.

The Semantic Interoperability Component (which is described in detail in deliverable D3.2) provides a unified API to access the data from the different data sources connected to the DataPorts platform, providing both real-time and batch data to the data consumers. In collaboration with the Data Access component, it will provide a solution for the interoperability of diverse data sources using existing ontologies. In order to ensure proper data protection in the exchanges with the data consumers, the API of this solution will interact with the Data Governance component to enable authentication and confidentiality, as well as to enforce the data access policies. Regarding the metadata, the Semantic Interoperability Component obtains information about the different data sources from the Data Access Manager and stores it in the Metadata Registry to make it available to the other local subcomponents. In addition, the metadata is sent to the IDS broker to provide information to data consumers about the available data sources. Other components of the DataPorts platform, like Data Abstraction and Virtualization and Automatic Prediction Engine, can retrieve the metadata that describe the data sources from the semantic interoperability API. The output data and metadata will follow the common DataPorts ontology and the common NGSI format.

The Semantic Interoperability Component provides a repository with the DataPorts Data Model and DataPorts Ontology. It will offer an ontology definition using OWL[29] and a Data Model description using JSON schema and JSON-LD context documents, which describe the representation of each entity type and the rules to interpret the data according to the ontology, respectively. The aim is to integrate the following ontologies and data models: Fiware Smart Data Models[30] , IDSA Information Model[31] , United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT)[32] model, Blockchain in Transport Alliance (BiTAS)[44], DCSA Interface for track & trace (DCSA)[45], IPSO Smart Objects (OMA SpecWorks)[46] and Smart Applications REFerence (SAREF) ontology[33]. Most of these ontologies and data models are described in Section 5.
The open source Fiware platform [11] has been selected to act as a core element of the Data Access and Semantic Interoperability components of the architecture presented and it will be adapted, customised and extended to fit the DataPorts project expectations.

The idea behind Fiware is that technology should be accessible for everyone, with a focus on interoperability, modularity, and customizability. Data from different relevant sources should be integrated seamlessly. For that reason, Fiware components provide open standardized data formats and API specifications to simplify this integration and facilitate the development of smart solutions. The platform is open source and can be easily embedded by ecosystem partners in the design of their solutions to reduce vendor lock-in risks. The standardized API means that services can operate on different vendor platforms. Fiware NGSI [12] is the API exposed by the Orion Context Broker and is used for the integration of components within a “Powered by Fiware” platform and by applications to update or consume context information. The Fiware NGSI (Next Generation Service Interface) API defines a data model for context information, an interface for exchanging context information and a context availability interface for queries on how to obtain context information. The agents are the connectors that guarantee the transmission of raw data to Orion Context Broker using their own native protocols.

Some implementation decisions have been made in order to collaborate with the Fiware ecosystem. Firstly, all data and metadata formats are going to be designed to follow Fiware NGSI data models specifications [13]. More concretely, the Data Model has been designed to be compatible with NGSI-LD, which is a standard set by the European Telecommunications Standards Institute (ETSI)[47]. NGSI-LD defines an interface and information model for sharing context information and represents an evolution of Fiware NGSI v2 to support Linked Data, which enables the association of the data being exchanged with an ontology in a way that can be processed by a machine. Secondly, regarding the data access agents, the use the pyngsi Python framework [38] has been proposed as a standardized SDK (Software Development Kit) to develop agents, although the use of this specific technology for their implementation is not mandatory. Finally, the use of the Orion Context Broker is being adopted as the semantic broker component [12]. In addition, another Fiware component, named Cygnus[48], has been included in the platform in order to ease the management of historical data. Cygnus is based on Apache Flume[49] and provides connectors to persist data in a variety of storage systems such as MongoDB, MySQL or HDFS, among others. The use of Orion Context Broker as a core element of the Semantic Interoperability Component is also relevant because the Orion Context Broker and Cygnus constitute one of the building blocks of the Connecting Europe Facility (CEF) Digital catalogue[50], named the CEF Context Broker.

The CEF building blocks are based on interoperability agreements in the European Union and provide basic capabilities that can be reused to facilitate the communication between IT systems and enable the implementation of services across borders and sectors in Europe. Each building block includes technical specifications and standards, a reusable implementation that complies with those specifications and a set of complementary services (such as onboarding services, helpdesk, tests, etc). The FIWARE/CEF Context Broker can be considered as an open de-facto standard for context information management and has been adopted by several relevant bodies like GSMA, TMForum, as well as initiatives like OASC (Open and Agile Smart Cities). Moreover, it has facilitated the development of smart solutions in multiple application domains, such as Smart Cities, Smart Agrifood and Smart Industry. Regarding the dataspaces, the Fiware Foundation is involved in the development of IDS implementations and is actively cooperating with the Industrial Data Space Association (IDSA). Fiware and IDSA participate in the Data Spaces Business Alliance (DSBA)[51], which is an initiative that aims to bring together the necessary industry players to build dataspaces and promote their adoption by organizations. In addition, an open-source implementation of the IDS Reference Architecture using Fiware components has been proposed [2].

The work done in previous European projects, where Fiware technology is a key element, is taken as a reference in the DataPorts implementation. For example, it is interesting to highlight projects like SynchroniCity[34] and Boost 4.0[35] . SynchroniCity is aimed at establishing a reference architecture for the envisioned IoT-enabled city marketplace, with identified interoperability points and interfaces and data models for different verticals. The baseline of the SynchroniCity[34] data models is the FIWARE Data Models initially created by the FIWARE Community and have been expressed using the ETSI standard NGSI-LD. Regarding Boost 4.0, an aim of the project is implementing the European Data Space with FIWARE technologies.

4.3 APPLICATION PROGRAMMING INTERFACES FOR SERVERLESS PLATFORMS

For the envisioned software components of the DataPorts architecture to be manifested, their interconnection through a standardized API is crucial in order to avoid a monolithic service approach that does not face the challenges presented above and causes a major drawback in the process of building a federated ecosystem. The concept of a service within a computational infrastructure has been fundamental through the evolution of different architectural designs and implementations. Application Programming Interfaces, however, constitute a powerful interoperability tool that enables the communication within a heterogenous infrastructure, resulting in loosely coupled components. For that reason, their usage has almost dominated the landscape of services, especially within serverless infrastructures.

Considering the WWW as an ecosystem of heterogenous services that require simplicity, Web APIs, or also known as RESTful services, have been increasingly dominating over the Web services that are based upon WSDL and SOAP. RESTful services conform to the REST architectural principles, which include constraints regarding the client-server communication, the statelessness of the request, and the use of a uniform interface. In addition, these services are characterized by resource representation decoupling, such that the resource content can manifest in different formats (i.e. JSON, HTML, XML etc.). Furthermore, the majority of Web APIs reliance on URIs for resource identification and interaction, and the HTTP protocol for message transmission, result in a simple technology stack that provides access to third parties, in order for them to consume and reuse data that originate from diverse services in data-oriented service compositions named mashups [24].

The evolution of the monolithic service-oriented architectures (SOA) has progressed in scope, from the management of the complexity of distributed systems to the integration of different software applications and has evolved through microservices into serverless architectures. In service-oriented architectures a service provides functionalities to other services mainly via message passing. With the modularization of these architectures into microservice ecosystems, different services are developed and scaled independently from each other according to their specific requirements and actual request stimuli, leading to the localization of decisions per service regarding programming languages, libraries, frameworks etc. However, the rise of cloud computing has led to serverless architectures that support the dynamic resource allocation and the corresponding infrastructure management, in order to enable auto-scaling based on event stimulus and to minimize operational costs [23]. In that context, Web APIs should be considered the cornerstone of the exploding evolution of service value creation, where enterprise systems are embracing the XaaS (anything as a Service) paradigm, according to which all business capabilities, products and processes are considered an interoperable collection of services that can be accessed and leveraged across organizational boundaries [5]. Nevertheless, since the beginning of the APIs prevalence over the traditional Web service technologies, the former has evolved in an autonomous way, lacking an established interface definition language [24]. Hence, in terms of serverless infrastructures, the subsequent need for homogeneity in application design and development has risen.

The common denominator in the development of serverless functions is their ability to support different functionalities in a scalable and stateless manner. For instance, there might be a serverless application that is integrated with an already existing ecosystem of functions that support API calls to cloud-based storage. While the former is scalable, the underlying storage system's on-demand scalability is bound to reliability and QoS guarantees. As far as these serverless implementations are concerned, two major use cases are addressed hereunder. The first one involves the composition of several APIs, while filtering and transforming the consumed data. A serverless function that implements this functionality mitigates the danger of network overload between the client and the invoked systems and offloads the filtering and aggregation logic to the backend. The second serverless application involves API aggregation, not only as a composition mechanism, but also to reduce API calls in terms of authorization, for example. This composition mechanism simplifies the client-side code that interacts with the aggregated call, by disguising multiple API calls into one with optional authorization from an external authorization service, e.g., an API gateway [3].

Despite the commonalities amongst serverless platforms in terms of pricing, deployment, and programming models, the most significant difference between them is the cloud ecosystem [3]. Differences in cloud platforms lead to discrepancies in developing tools and frameworks that are available to developers for creating services native to each platform. The ever-evolving serverless APIs, in combination with the corresponding frameworks and libraries, represent a significant obstacle for software lifecycle management, and for service discovery, and brokering. The plethora of incompatible APIs in terms of serverless technology has created the need for multicloud API standardization, interoperability, and portability in order to achieve seamlessness. In this direction, informal standardization has been formed, following community efforts towards addressing the lack of a common programming model that enables platform-agnostic development and interoperability of functions [42].

Following the problem identification depicted above, the solution to the lack of a standardized and programming language-agnostic interface description language is fulfilled by the OpenAPI Specification (OAS). The OpenAPI initiative was founded in November 2015 by the collaboration of SmartBear, 3Scale, Apigee, Capital One, Google, IBM, Intuit, Microsoft, PayPal, and Restlet. This initiative was formed as an open-source project under the Linux Foundation and was designed to enable both humans and computers to explore and understand the functionalities of a RESTful service without requiring access to source code, additional documentation, or inspection of network traffic. OAS enables the understanding and interaction with the remote service with a minimal amount of implementation logic according to a vendor neutral description format. The OpenAPI Specification was based on the rebranded Swagger 2.0 specification, donated by SmartBear Software in 2015 [28].

The most important advantages of the OpenAPI Specification are two-fold. On one hand, the business benefit that emerges is the recognition of this standardization as a useful means for a lot of developers to develop open-source repositories of tools that leverage this enablement. Furthermore, OAS is supported by a group of industry leaders that contribute with their strong awareness and mindshare, while indicating stability across a diverse code base. On the other hand, OAS is registered as a powerful technical tool that is most importantly language-agnostic and provides understanding of an API without the involvement of server implementation. Its documentation is regularly updated by a broad community that provides additionally example implementations, code snippets and responses to inquiries [28].

According to the above, the significance of the standardization that OAS offers, rationalizes its adoption as the proposed solution interface description language for the proposed port platform. The pluralism of different data sources that need to be integrated within this platform, in combination with the existence of different frameworks and technologies of the existing APIs that are already utilized by ports, introduce the need for a uniform description of the exposed interfaces. For instance, the IoT infrastructure that includes APIs that are exposed by smart containers, enable the aggregation of information that implement the life cycle assessment (LCA) applied for port logistics operations, and need to be integrated seamlessly with the different components of the platform. Moreover, crucial role in the message exchange between the different infrastructures within the port ecosystem play the APIs that facilitate the communication between components, and therefore, the standardization of their interface is of absolute importance for the scalability and interoperability of the platform. The DataPorts architecture can be constructed based on the OpenAPI specification, enabling the development of add-on services and the creation of added value of the available data. Furthermore, the implementation of the platform within a serverless architecture framework underlines the significance of the OpenAPI specification as a powerful standard for the interface description of all services within and exposed by the DataPorts Platform.

4.4 INTEROPERABILITY ENSURED BY INTERNATIONAL DATA SPACES REFERENCE ARCHITECTURE MODEL

The IDS Reference Architecture Model benefits from an information model, which is an essential agreement shared by the participants and components of the IDS, facilitating interoperability and compatibility. The main aim of this formal model is to enable (semi-)automated exchange of digital resources in a trusted ecosystem of distributed various parties, while the sovereignty of Data owners is preserved.

Data sovereignty is defined as an entity’s capability of being entirely self-determined with respect to its data [40]. To this end, all organizations attempting to access the IDS ecosystem must be certified, as do the core software components (for instance IDS connector) used for trusted data exchange and data sharing. Such a certification not only ensures security and trust, but the existence of certified components guarantees the compliance with technical requirements, ensuring interoperability.

5 STANDARDISATION

IDSA [40] aims at open, federated data ecosystems and marketplaces, by ensuring data sovereignty for the creator of the data through the establishment of a virtual data space for the standardised, secure exchange and trade of data. Standards, be they national, regional or global, are the fruit of collective efforts and guarantee interoperability. They can be revised to meet industry needs and remain relevant over time. Standards organisations, where participants from different segments of the industry gather, are among the few places where competitors work side-by-side. Standards organisations offer a safe place to do so from an anti-trust perspective. Standards development participants are industry experts, tech companies and customers representing all fields of the industry.

Adoption of global multimodal data exchange standards guarantees interoperability. In fact, the smart container standardisation effort [7] [8] is one of many standardisation initiatives [26] supporting global trade. Standards enable stakeholders in the logistics chain to reap the maximum benefits from smart container solutions, while enabling them to share data and associated costs. Standards-based data exchange usage increases the ability to collaborate, which in turn increases efficiency. Additionally, such standards reduce development and deployment costs and cut time-to-market for Internet of Things (IoT) solution providers.

Data exchange standards developed in an open process offer a useful aid to all parties interested in the technical applications and implementation of smart container solutions. Additionally, if solution providers find there are new data elements required to accommodate changing business requirements, it is possible to create a backwards compatible revision of the standard to accommodate their needs.

With the ramp-up of new and emerging technologies, these standards are more necessary than ever. Standards reduce the risk of developing proprietary technologies with significant deployment limitations and the lack of interoperability among systems and devices. Standards enable the parties to avoid costly and time-consuming integration and limit the risk of vendor lock-in. In this context, IDS provides a generic framework that can be leveraged by domain-specific instantiations.

In the following sections, industry standards relevant to DataPorts are introduced. These standards are either brought as an answer to the general requirements and linked to compliant technical components or brought as the basis of semantic interoperability by partners' technologies or products. The data model resulting from the integration of all the use cases in accordance with these standards is planned to be detailed in Deliverable 3.5 and used in the pilots.

5.1 UN/CEFACT SMART CONTAINER

The United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT) Smart Container Business Requirements Specification (BRS) ensures that the various ecosystem actors share a common understanding of smart container benefits by presenting various use cases. It also details the smart container data elements [41]. Defining the data elements that smart containers can generate accelerates integration and the use of smart container data on different platforms for the enhancement of operations. In addition, utilising standard smart container data enables open communications channels between supply chain actors.

Standard data models and standard APIs would help stakeholders to make the necessary transformation to achieve supply chain excellence [6]. Indeed, APIs are key to ensuring simplification and acceleration of the integration of digital services from various sources.

The focus of the UN/CEFACT Smart Container project is to define the data elements via varied use cases applicable to smart container usage. Currently, the data model is being developed, which will provide the basis for the smart container standard messaging and Application Programming Interfaces. The Smart Container API catalogue will be taken into account in the interface's specification, enabling software components to communicate with each other. It is crucial to first determine and align the required data elements and their semantics.

Smart containers will revolutionise the capture and timely reporting of data throughout the supply chains.

Such containers are an essential building block to meeting the emerging requirements for end-to-end supply chains. As leading carriers adopt smart container solutions, they gain valuable data that can be shared with shippers and other supply chain stakeholders.

However, generating and collecting data is not enough to make smart container solutions or supply chains "smart." Stakeholders already manage huge amounts of data and struggle with multiple technologies that take time away from their core businesses. A smart container solution must deliver data that matters, in a standard format for easy integration into different systems. It must enable unambiguous data interpretation and empower all involved stakeholders with actionable information. Clear semantic standards are essential for effective smart container data exchange, ensuring that all stakeholders understand the same information in the same way. Then and only then, can smart containers truly become part of digital data streams [36].

The UN/CEFACT Smart Container project aims to create multimodal communications standards that can facilitate a state-of-the-art solution in providing and exposing services. Any intermodal ecosystem stakeholder may then orchestrate and enrich these services to meet their business process needs. The availability and exposition of these services can boost the digital transformation of the transportation and logistics industry, fuelling innovation in new applications and services. Physical supply chains that move goods need a parallel digital supply chain that moves data describing the goods and their progress through the supply chain. The smart container data flows ensure that the physical flow is well synchronised with the required documents flow. Data are the raw material of Maritime Informatics. Without data streams emanating from operations there can be no data analytics. As we digitalise, we improve operational productivity and lay the foundation, through Maritime Informatics, for another round of strategic and operational productivity based on big data analytics and machine learning.

UN/CEFACT Multimodal Transport data model has been relevant for DataPorts as a context-setting standard going beyond only the interactions at the seaport, encompassing the full shipping process. However, at this point of the project, no update to the Core Components Library has yet been identified to contribute to the standard.

5.2 SMART DATA MODELS

The DataPorts data model will follow the guidelines of Smart Data Models [52] and it will also integrate some of their concepts. The aim is to contribute to the Smart Data Models initiative with new definitions of entity types based on the use cases and domains of DataPorts.

The Smart Data Models initiative is a collaborative Fiware-related initiative for the publication of models describing data that participants of data spaces can exchange. Those models provide harmonised formats and semantics to represent the data, are based on open standards and real use cases, and are published under open and royalty-free licenses in order to facilitate interoperability in different application domains. The initiative is led by the FIWARE Foundation[53], IUDX[54], TM Forum[55] and OASC[56] and accepts contributions from other organizations.

A smart data model includes mainly the following elements:

- the schema, or technical representation of the model, defining the technical data types and structure
- the specification of a written document for human readers
- the examples of the payloads for NGSIv2 and NGSI-LD versions

The principles of the Smart Data Models initiative are based on a community site with detailed data models for multiple business sectors available for open use. The elaboration is done together with other relevant organizations, curating data models in different domains and subjects. The aim is providing coherence and consistency between data models across different domains. They are focused on creating a method for AGILE standardization and evolution of these data models. In addition, the data models are based on widely adopted standards (e.g., ontologies and international schemas, schema.org). They are focused on providing extended usefulness to FIWARE platform users in terms of extended interoperability and reduced time dedicated to data model coding, capitalising on the accumulated experience from other projects. Finally, they use the consensus as their main decision method.

This standard is especially relevant for DataPorts as it is fully integrated in Fiware components chosen for the platform.

5.3 BITA (BLOCKCHAIN IN TRANSPORT ALLIANCE)

BiTA members are primarily from the freight, transportation, logistics and affiliated industries. Alliance members share a common mission of driving the adoption of emerging technology forward and try to accomplish this by developing industry standards, educating members and others on blockchain applications/solutions and distributed ledger technology (DLT), and encouraging the use and adoption of new solutions.

The concepts of BiTA are especially relevant for DataPorts as VPF has adopted multiple ideas from them as the basis for their cognitive seaport platform requirements.

5.4 DCSA INTERFACE FOR TRACK & TRACE

The Digital Container Shipping Association and its member carriers have published track and trace (T&T) standards for the global container shipping industry. These standards comprise a common set of processes as well as data and interface standards that can be implemented by carriers, shippers and third parties to enable cross-carrier shipment tracking. Once implemented, the standards enable customers and supply chain participants to digitally communicate with all carriers in a unified way.

The data model ensures track and trace data definitions are consistent for all users, leveraging any system. These definitions are based on the Industry Blueprint published by DCSA and its carrier members in 2019, which established a consistent vocabulary and proposed current state standards for industry processes.

The T&T standards are also aligned with the UN/CEFACT standards to ensure existing investments are preserved while streamlining communication among all supply chain participants. They are especially relevant for DataPorts as they were adopted by VPF and TRX for their own data model initiatives.

5.5 IPSO SMART OBJECTS (OMA SPECWORKS)

Launched in 2018, OMA SpecWorks joins together the Open Mobile Alliance (OMA) and the IPSO Alliance with a new mission to build technical documents including specifications, smart objects, and white papers for a connected world. Their goal is enabling interoperability across networks and growth in fixed and mobile wireless markets and the Internet of Things (IoT).

IPSO Smart Object Guidelines provide a common design pattern, an object model, that can effectively use the IETF CoAP protocol to provide high-level interoperability between Smart Object devices and connected software applications on other devices and services.

The common object model is based on the Lightweight M2M specification from the Open Mobile Alliance which is a device management and service architecture specification based on IETF CoAP that provides a simple and flexible object template (object model) for constrained device management. It is specifically adapted to large fleet of IoT devices that require large scalability and security. This is the basis for the definition of Smart Containers sensors data model from TRX which makes this standard relevant for DataPorts.

6 BUSINESS OUTCOMES AND CHALLENGES

The fast-growing complexity at seaports makes data management essential, hence the optimal goal is to achieve greater efficiency. The use of large volumes of data (Big Data) is indisputably a major aid to this goal [27]. AI based services available in a smart seaports is a new revenue source for many stakeholders. When such data and services are offered through a standard mechanism, as is a data-driven platform, this offering is of great importance to third parties, as it acts as a leverage to improving and increasing various port operations, especially those that are associated with traffic, vessel and cargo movement. Imagine the case where cargo transfer data are accessible to the shipping lines and, at the same time, all seaport operations can be available to Port Authority associates. Passenger mobility patterns may be available not only to the Port Authority but also to the city's decision and policy makers. Commercial or cultural associations may also be interested in accessing such services, especially when coming from seaports with high passenger activity. Such services will make seaports "smart" and "cognitive", and eventually increase their ecosystem boundaries, as data exchange will create a demand for additional services. Towards this direction, the research community and the shipping related SMEs, or even the start-up community, may benefit from analysing large volumes of data offered by data providers and propose additional offerings. Moreover, Analytics-as-a-Service, using data collected from shipping and freight companies, warehouses, customs brokers and other port operations, may be key for data monetization. This will dissolve any fears of losing a competitive advantage due to data sharing.

From a business perspective, for data and service sharing to be effective and useful for as many beneficiaries as possible, certain Quality of Service (QoS) characteristics should be followed. Some of the key characteristics considered, especially for Big Data, are Volume, Velocity, Variety, Veracity, and Value. Moreover, offering data and services should match certain needs and be easily accessible to the users. Therefore, in terms of time, data should be up to date, real and able to be authenticated. Additionally, guarantying a QoS may be difficult for heterogeneous data, especially when the competitiveness is increasing proportionally to the demand for new data and services. Hence, a monitoring mechanism is needed to ensure the above-mentioned characteristics as well as the validity of the transferred data. Regulatory compliance is considered a main business-related concern with data QoS, which currently is vague. Also of concern are customer satisfaction, which is the main goal, data validity and accuracy for purposes of decision making, data relevance, data completeness (e.g., no missing values), and data consistency in the format expected by the users.

From a technical perspective, all types of data science applications deal with large amounts of data stored in various storage devices or systems. Distributed storage systems are often chosen for storing these data, as depicted also in Fig.1, thus shaping requirements for IDS. Some of the requirements posed to those storage systems may concern Quality of Service (QoS) aspects formally expressed in a Service Level Agreement, as was the traditional approach in the past. The role of QoS is to provide the necessary technical specifications that specify the system quality of features such as performance, availability, scalability, and serviceability. Within the IDS ecosystem, system qualities are closely interrelated. Requirements for one system quality might affect the requirements and design for other system qualities. For example, having a higher level of security policies might affect performance, which in turn might affect availability. Adding extra servers to address availability issues affects serviceability (maintenance costs). Understanding how system qualities are interrelated and the trade-offs that must be made is the key to designing a system that successfully satisfies both business requirements and business constraints. Having these QoS attributes in mind, it's evident that the QoS management in a distributed and heterogeneous environment is a challenging task, given the possible storage device heterogeneity, the dynamically changing data access patterns, the client's concurrency and storage resource sharing. The problem becomes even more complicated when distributed computing environments with virtualized and shared resources like Clouds and Blockchains are considered. Furthermore, various heterogeneous devices or objects should be integrated for transparent and seamless communication under the umbrella of Internet of things (IoT). This would facilitate the open access of data for the growth of various digital services. Building a general framework or selecting an approach for handling

QoS becomes a complex task due to the heterogeneity in devices, technologies, platforms and services operating in the same system. Additionally, Data Analytics and Governance should follow an all-encompassing approach to consumer privacy and data security, in contrast to Compliance Regulations, that will become a benchmark for how personal data are treated in the future. In the area of cognitive ports, regulations introduce major restrictions and complexities for QoS from a technical perspective, especially those parts that address how contact data are handled, or those that specify both tools and processes as part of the compliance effort to achieve data quality. In such heterogeneous environment, technical aspects of QoS relate to technical translation and treatment of compliance as concerning entities of rights to: “access”, “be informed”, “data portability”, ”be forgotten”, ”object”, ”restrict processing”, ”be notified”, ”rectification” and so on in addition to the aforementioned technical specifications.

Most technical challenges mentioned above were solved using the serverless architecture approach, as described in Section 3. The use of APIs and Semantic Interoperability described in Section 4.2 provides a small introduction on this approach. Therefore, using the Serverless as well as the common microservices paradigms, cognitive ports are in a PaaS and Data-as-a-Service environment, where the majority of traditional QoS aspects are dynamical adapted to needs.

Additionally, the aspects of Compliance Regulations are approached by the creation of workflows into the Blockchain and Broker infrastructure of the Cognitive Port. For example, upon the request for every provider to comply with existing regulations concerning data and identity attributes, any stakeholder that provides data is responsible for the integrity and compliance of their provided data. Therefore, the Cognitive Ports are candidates for becoming a dynamic data sharing ecosystem.

7 REFERENCES AND ACRONYMS

7.1 REFERENCES

A large part of this deliverable has also been accepted by BDVA as a chapter of the book entitled "Technologies and Applications for Big Data Value" to ensure dissemination of the project's results. Credited authors are: Santiago Cáceres, Francisco Valverde, Carlos E. Palau, Andreu Belsa Pellicer, Christos A. Gizelis, Dimosthenes Krassas, Hanane Becha, Réda Khouani, Andreas Metzger, Nikos Tzagkarakis, Anthousa Karkoglou, Anastasios Nikolakopoulos, Achilleas Marinakis, Vrettos Moulos, Antonios Litke, Amir Shayan Ahmadian and Jan Jürjens.

1. Amir Shayan Ahmadian, Jan Jürjens, and Daniel Strüber. Extending model-based privacy analysis for the industrial data space by exploiting privacy level agreements. In Proceedings of the 33rd Annual ACM Symposium on Applied Computing - SAC '18. ACM Press, 2018.

2. Álvaro Alonso, Alejandro Pozo, José Cantera, Francisco de la Vega, and Juan Hierro. Industrial data space architecture implementation using FIWARE. Sensors, 18(7):2226, jul 2018.

3. Ioana Baldini, Paul Castro, Kerry Chang, Perry Cheng, Stephen Fink, Vatche Ishakian, Nick Mitchell, Vinod Muthusamy, Rodric Rabbah, Aleksander Slominski, and Philippe Suter. Serverless computing: Current trends and open problems. In Research Advances in Cloud Computing, pages 1–20. Springer Singapore, 2017.

4. David Balla, Markosz Maliosz, and Csaba Simon. Open source FaaS performance aspects. In 2020 43rd International Conference on Telecommunications and Signal Processing (TSP). IEEE, jul 2020.

5. Rahul C. Basole. On the evolution of service ecosystems: A study of the emerging API economy. In Handbook of Service Science, Volume II, pages 479–495. Springer International Publishing, oct 2018.

6. Hanane Becha. How Standard APIs Open the Door to Powerful Digital Services. https://hananebecha.home.blog/2019/11/28/the-un-cefact-smart-container-project/, November 2019. [Online; accessed 30-October-2020].

7. Hanane Becha. Standardization Supporting Global Trade. Port Technology International, 2019. Edition 91, Shipping 2020: A Vision for Tomorrow, https://www.porttechnology.org/editions/shipping-2020-a-vision-for-tomorrow/.

8. Hanane Becha. The UN/CEFACTSmart Container Project. TheReport: TheMagazine of the International Institute of Marine Surveying, March 2020. Issue 91, https://www.iims.org.uk/wpcontent/uploads/2020/02/The-Report-March-2020.pdf.

9. Laura Daniele and L Ferreira Pires. An ontological approach to logistics. Enterprise interoperability, research and applications in the service-oriented ecosystem, IWEI, 13:199–213, 2013.

10. DataPorts H2020 EU Project. http://dataports-project.eu. [Online; accessed 30-October-2020].

11. Fiware Open Source Platform. https://www.fiware.org/. [Online; accessed 30-October-2020].

12. Fiware Orion Context Broker. https://fiware-orion.readthedocs.io/en/master/. [Online; accessed 30-October-2020].

13. Fiware Data Models. https://fiware-datamodels.readthedocs.io/en/latest/index.html. [Online; accessed 30-October-2020].

14. Giancarlo Fortino, Claudio Savaglio, Carlos E. Palau, Jara Suarez de Puga, Maria Ganzha, Marcin Paprzycki, Miguel Montesinos, Antonio Liotta, and Miguel Llop. Towards multi-layer interoperability of heterogeneous IoT platforms: The INTER-IoT approach. In Internet of Things, pages 199–232. Springer International Publishing, jul 2017.

15. Maria Ganzha, Marcin Paprzycki, Wieslaw Pawlowski, Pawel Szmeja, and Katarzyna Wasielewska. Semantic technologies for the IoT - an inter-IoT perspective. In 2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI). IEEE, apr 2016.

16. Christos-Antonios Gizelis, Theodoros Mavroeidakos, Achilleas Marinakis, Antonis Litke, and Vrettos Moulos. Towards a smart port: The role of the telecom industry. In Artificial Intelligence Applications and Innovations. AIAI 2020 IFIP WG 12.5 International Workshops, pages 128–139. Springer International Publishing, 2020.

17. Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing? International Journal of Human-Computer Studies, 43(5-6):907–928, nov 1995.

18. Sandra Heiler. Semantic interoperability. ACM Computing Surveys, 27(2):271–273, jun 1995.

19. Scott Hendrickson, Stephen Sturdevant, Tyler Harter, Venkateshwaran Venkataramani, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. Serverless computation with openlambda. In 8th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 16), Denver, CO, June 2016. USENIX Association.

20. Apache OpenWhisk. https://openwhisk.apache.org/. [Online; accessed 05-January-2021].

21. OpenFaaS. https://docs.openfaas.com/. [Online; accessed 05-January-2021].

22. Kubeless. https://kubeless.io/. [Online; accessed 05-January-2021].

23. Nane Kratzke. A brief history of cloud application architectures. Applied Sciences, 8(8):1368, aug 2018.

24. Maria Maleshkova, Carlos Pedrinaci, and John Domingue. Investigating web APIs on the world wide web. In 2010 Eighth IEEE European Conference on Web Services. IEEE, dec 2010.

25. Garrett McGrath and Paul R. Brenner. Serverless computing: Design, implementation, and performance. In 2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW). IEEE, jun 2017.

26. Mikael Lind, André Simha, and Hanane Becha. Creating value for the transport buyer with digital data streams. The Maritime Executive, 2020. https://maritimeexecutive. com/editorials/creating-value-for-the-transport-buyer-with-digital-data-streams.

27. Vrettos Moulos, George Chatzikyriakos, Vassilis Kassouras, Anastasios Doulamis, Nikolaos Doulamis, Georgios Leventakis, Thodoris Florakis, Theodora Varvarigou, Evangelos Mitsokapas, Georgios Kioumourtzis, Petros Klirodetis, Alexandros Psychas, Achilleas Marinakis, Thanasis Sfetsos, Alexios Koniaris, Dimitris Liapis, and Anna Gatzioura. A robust information life cycle management framework for securing and governing critical infrastructure systems. Inventions, 3(4):71, oct 2018.

28. OpenAPI Specification. https://www.openapis.org/faq. [Online; accessed 30-October-2020].

29. OWL. https://www.w3.org/TR/owl-guide/. [Online; accessed 05-January-2021].

30. FireWire Data Model. https://www.fiware.org/developers/data-models/. [Online; accessed 05-January-2021].

31. IDSA Information Model. https://github.com/International-Data-Spaces-Association/InformationModel. [Online; accessed 05-January-2021].

32. UN/CEFACT. https://umm-dev.org/about-umm/. [Online; accessed 05-January-2021].

33. SAREF Ontology. https://saref.etsi.org/. [Online; accessed 05-January-2021].

34. SynchroniCity. https://synchronicity-iot.eu/. [Online; accessed 05-January-2021].

35. Boost 4.0 – Big Data for Factories. https://boost40.eu/. [Online; accessed 05-January-2021].

36. Federico Pigni, Gabriele Piccoli, and Richard Watson. Digital data streams: Creating value from the real-time flow of big data. California Management Review, 58(3):5–25, may 2016.

37. Evangelos Psomakelis, Anastasios Nikolakopoulos, Achilleas Marinakis, Alexandros Psychas, Vrettos Moulos, Theodora Varvarigou, and Andreas Christou. A scalable and semantic data as a service marketplace for enhancing cloud-based applications. Future Internet, 12(5):77, apr 2020.

38. Pyngsi Python Framework. https://github.com/pixel-ports/pyngsi. [Online; accessed 30-October-2020].

39. Zillner S., Curry E., Metzger A., Auer S., and Seidl R. European big data value strategic research & innovation agenda. Big Data Value Association., 2017.

40. Sören Auer, Jan Jürjens, Boris Otto, Gerd Brost, Christoph Lange, Christoph Quix, Jan Cirullies, Steffen Lohmann, Andreas Eitel, Christian Mader, Daniel Schulz, Thilo Ernst, Nadja Menz, Julian Schütte, Christian Haas, Lars Nagel, Markus Spiekermann, Manuel Huber, Heinrich Pettenpohl, Sven Wenzel, Christian Jung, and Jaroslav Pullmann. Reference Architecture Model for the Industrial Data Space. https://www.internationaldataspaces.org/wpcontent/uploads/2019/03/IDS-Reference-Architecture-Model-3.0.pdf, April 2019. [Online;accessed 30-October-2020].

41. UN/CEFACT (2019) The UN/CEFACT Smart Container Business Specifications (BRS). https://www.unece.org/fileadmin/DAM/cefact/brs/BRS-SmartContainer_v1.0.pdf. [Online; accessed 30-October-2020].

42. Erwin van Eyk, Lucian Toader, Sacheendra Talluri, Laurens Versluis, Alexandru Uta, and Alexandru Iosup. Serverless is more: From PaaS to present cloud computing. IEEE Internet Computing, 22(5):8–17, sep 2018.

43. Milano M. Curry E. García Robles A. Hahn T. Irgens M. Lafrenz R. Liepert B. O'Sullivan B. Zillner S., Bisset D. and Smeulders A. Strategic research, innovation and deployment agenda - ai, data and robotics partnership. third release. Big Data Value Association., September 2020.

44. BITAs https://www.bita.studio/

45. DCSA Interface Standard for Track and Trace 1.2 https://dcsa.org/standards/track-trace/

46. IPSO Smart Objects (OMA SpecWorks) https://omaspecworks.org/develop-with-oma-specworks/ipso-smart-objects/

47. ETSI https://www.etsi.org/committee/cim

48. Cygnus https://fiware-cygnus.readthedocs.io/en/latest/

49. Apache Flume http://flume.apache.org/

50. Connecting Europe Facility (CEF) Digital catalogue https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/CEF+Digital+Home

51. Data Spaces Business Alliance https://internationaldataspaces.org/bdva-fiware-gaia-x-and-idsa-launch-alliance-to-accelerate-business-transformation-in-the-data-economy/

52. Smart Data Models https://smartdatamodels.org/53. FIWARE Foundation http://fiware.org/

54. IUDX http://iudx.org.in/

55. TM Forum http://tmforum.org/

56. OASC https://oascities.org/

7.2 ACRONYMS

Acronym List	
AI	Artificial Intelligence
API	Application Programming Interface
BDV(A)	Big Data Value (Association)
BITA	Blockchain In Transport Alliance
BRS	Business Requirements Specification
CEF	Connecting Europe Facility
COAP	COnstrained Application Protocol
CPU	Central Processing Unit
CSV	Comma Separated Values
DaaS	Data-as-a-Service
DCSA	Digital Container Shipping Association
DLT	Distributed Ledger Technology
DOA	Description Of Action
DSBA	Data Spaces Business Alliance
ETA	Estimated Time of Arrival
ETSI	European Telecommunications Standards Institute
FaaS	Function-as-a-Service
GSMA	Global System for Mobile communications Association
HDFS	Hadoop Distributed File System
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IDS(A)	International Data Spaces (Association)
IETF	Internet Engineering Task Force
IoT	Internet of Things
IPSO	Internet Protocol for Smart Objects
IT	Information Technology
IUDX	India Urban Data Exchange
JSON	JavaScript Object Notation
JSON-LD	JavaScript Object Notation for Linked Data
M2M	Machine to Machine
ML	Machine Learning
MVCC	MultiVersion Concurrency Control
NGSI	Next Generation Service Interface
NGSI-LD	Next Generation Service Interface for Linked Data
OAS	OpenAPI Specification
OASC	Open and Agile Smart Cities
OMA	Open Mobile Alliance
OWL	Web Ontology Language
PPP	Public Private Partnership
QoS	Quality of Service
RAM	Reference Architecture Model
REST	REpresentational State Transfer
SAREF	Smart Applications REFerence
SDK	Software Development Kit
SDO	Standard Development Organization
SME	Small and Medium Enterprises

SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
T&T	Track and Trace
TOS	Terminal Operating System
UN/CEFACT	United Nations Centre for Trade Facilitation and Electronic Business
URI	Uniform Resource Identifier
WDSL	Web Services Description Language
WWW	World Wide Web
XaaS	anything as a Service
XML	eXtensible Markup Language