SMART GREEN PORTS

D4.3 Digital Representation of Assets and Systems in Ports

Funded by
the European Union

This project has received funding from the European Union's Horizon 2020 (MFF 2014-2020) research and innovation programme under Grant Agreement 101036594

D4.3 DIGITAL REPRESENTATION OF ASSETS AND SYSTEMS IN PORTS

GRANT AGREEMENT NO.	101036594
START DATE OF PROJECT	1 October 2021
DURATION OF THE PROJECT	60 months
DELIVERABLE NUMBER	D4.3
DELIVERABLE LEADER	TNO
DISSEMINATION LEVEL	PU
STATUS	V1.0
SUBMISSION DATE	12-10-2023
AUTHOR	Cornelis Bouter, TNO (cornelis.bouter@tno.nl) Bas Gerrits, TNO (bas.gerrits@tno.nl) Wouter Korteling, TNO (wouter.korteling@tno.nl) Theodor Chirvasuta, TNO (theodor.chirvasuta@tno.nl) Sthefan Berwanger, INESC TEC (sthefan.berwanger@inesctec.pt) Annelore Luning, Port of Rotterdam (at.luning@portofrotterdam.com)

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 101036594.

PU=Public, CO=Confidential, only the members of the consortium (including the Commission Services), CI=Classified, as referred to in Commission Decision 2001/844/EC.

Modification Control

VERSION #	DATE	AUTHOR	ORGANISATION
V0.1	07-04-2023	C.A. Bouter	TNO
V0.2	15-09-2023	C.A. Bouter	TNO
V1.0	04-10-2023	C.A. Bouter	TNO

Release Approval

NAME	ROLE	DATE
Stephan Raaijmakers	TNO reviewer	29-06-2023
Barry Nouwt	TNO reviewer	29-06-2023
Peter Mollema	Peer reviewer	25-08-2023
Yves-Marie Bourien	Peer reviewer	31-08-2023
Zenaida Mourao	WP Leader	20-09-2023
Maarten Flikkema	Scientific Coordinator	25-09-2023
Arne-Jan Polman	Project Coordinator	23-10-2023

History of Changes

SECTION, PAGE NUMBER	CHANGE MADE	DATE

Table of Contents

Executive Summary

The MAGPIE project is an international collaboration working on demonstrating technical, operational, and procedural energy supply and digital solutions in a living lab environment to stimulate green, smart and integrated multimodal transport and ensure roll-out through the European Green Port of the Future Master Plan and dissemination and exploitation activities. The consortium, coordinated by the Port of Rotterdam, consists of 3 other ports (DeltaPort, Sines and HAROPA), 9 research institutes and universities, 32 private companies, and 4 other organisations. The project is divided in 10 main work packages which include energy supply chains, digital tools, 10 demonstrators for maritime, inland water, road, and rail transport, non-technological innovations and the development of a Masterplan for European Green ports.

This document, Deliverable 4.3, describes the various aspects of the language specification for the Port Digital Twin (DT) and serves as a guide for extensions of the language specification in the demos and tools that are currently being developed. In the previous deliverable, D4.2, a DT was defined as containing three components: the data sharing infrastructure, the language specification, and the tools. Details of the data sharing infrastructure were simultaneously covered in D4.2, the tool details will be specified in the respective deliverables across MAGPIE, and the language specification details are covered in this document.

First, we argue that semantic models, and specifically "ontologies" implemented using Resource Description Framework (RDF), provide a suitable framework to implement the language specification for the Port DT. Ontologies are a type of semantic models that provide a common vocabulary for defining, categorizing, and organizing information, making it easier to understand and interpret shared data. The reason that ontologies implemented using RDF provide a suitable framework for the Port DT is that they support, amongst others, future extensibility in new use cases, reuse of existing models, global identification, and federated data storage. The use of RDF can additionally support the European FAIR data principles. Second, we describe a methodology for extending the language specification. This methodology is necessary because the Port DT is intended to support tools that have not been finalized or conceived yet. The methodology facilitates the extension of the language specification to support these future applications. Third, to facilitate usability of the methodology, we demonstrate how this methodology can be applied in several use cases to extend and update the language specification. Fourth, we describe a core semantic model that reuses and integrates several existing models, such as SAREF originating from ETSI and the ERA vocabulary originating from the EU Agency for Railways. This core model works as a basis of quality models that can be reused and extended by the various tools.

We could not fully reflect all data requirements of the tools in T4.4 and T4.5, as well as the various MAGPIE demos, in the language specification. The reason is that the development of these tools and demos have only started near the end of T4.3 and their data requirements that have to be expressed in the language specification are therefore not specified at the time of writing this deliverable. The development methodology we have outlined and validated in this deliverable can, however, serve as a practical guideline to facilitate extensions of the core model to support new use cases once their data requirements are known.

Future work in MAGPIE should therefore focus on extending the core model using the proposed methodology to cover these tools and use cases once they are in a further stage of development. The developers of the tools, however, still need to invest in obtaining external expertise on RDF and semantic engineering. The current document guides a proper implementation of semantic models, but cannot completely replace the role of a semantic expert. In addition, future efforts in MAGPIE should focus on the technical integration of the

work presented in this deliverable with the work on data spaces and vocabulary hubs as identified in D4.2. This integration of D4.2 can D4.3 can then form a basis for the tools and demo's than want to connect to the Port Digital Twin.

1. Introduction

This document is the deliverable of task T4.3 in work package WP4. In this task, the methodology for defining and modelling the language specification that is needed to implement the Port Digital Twin (DT) (see deliverables D4.1[1] and D4.2[2]) is developed. A language specification contains a formal and generic specification of the language used for data sharing between different stakeholders and systems, such as those in MAGPIE. In addition to describing a methodology for developing an adequate language specification, the current document describes a first language specification for applications and systems in the port implemented as a semantic core model (or core ontology).

As explained in deliverable D4.2 of task T4.2, the operationalisation of a DT of the port requires three components: (1) a data sharing infrastructure, (2) a language specification of the shared and available port data and (3) (intelligent) tools and systems that produce and consume data. The data sharing infrastructure arranges communication between the various tools, systems, data users and data holders. The envisioned data sharing architecture to support the DT of the port is described in deliverable D4.2. The tools and systems that are integrated in the DT enable simulation and automation or facilitation of (strategic) decision processes. Several of such tools that need to be integrated to the DT of the port are developed in WP4 and described in deliverables D4.4 and D4.5. Finally, the language specification ensures interpretability and interoperability of the different systems and applications that are integrated in the DT. This is the topic of the current deliverable.

The objective of the current deliverable is four-fold:

1. Explain the rationale for using semantic models as language specification for the Port DT.
2. Describe a methodology of how to develop the semantic models such that they adhere to the functional requirements of the systems and application that want to integrate with the port DT and ensure semantic interoperability between these systems.
3. Show how the language specification can be maintained, extended and updated. Afterall, the language specification is a "living" specification that will continue to be extended based on additional requirements that emerge when new tools need to be integrated in the DT of the port.
4. Describe a first version of a semantic core model that can be used for further extensions in coming use cases.

This means that the work packages in MAGPIE that implement the demo's and/or tools and wish or require to connect to the port DT can use the development methodology described in this deliverable to describe their (data) requirements (data input/output, format, etc.) in an optimal way to elicit the requirements on the semantic model. Based on these data requirements, a data model or DT of the tool/system can be developed and added to the Port DT. Note, that quite some technical/semantic detail is needed from the tool or system to determine their data requirements and build an appropriate extension to the semantic core model. So, the development of the tools should already be in a phase where the required functionality, including the data input and data output, are clear before the semantic engineer can extend the core model.

The methodology is currently in the process of being applied on the various tools of T4.4 and T4.5. The methodology is also intended to connect the MAGPIE tools developed in other WPs to the DT infrastructure. This makes the current deliverable also valuable in the design process of these other tools. Additionally, the MAGPIE proposal states that the DT should

[1] MAGPIE Deliverable 4.1: Digital Platforms & Services for Port Operations (2022).
[2] MAGPIE Deliverable 4.2: Definition of Modular Architecture for Port Digital Twin (2023).

accommodate the implementation of tools not yet identified in the main project. The methodology is also intended to facilitate that process.

Two physical assets from the port domain are modelled by following the methodology and extensively described in this deliverable. These parts of the port we modelled as an example are the terminal and a berth with shore power. Both are essential assets in the proto port use case (described in D4.2) and MAGPIE objective (e.g., see D4.1 and D4.2).

The document is structured as follows:

- Chapter 2 describes a vision on the capabilities and requirements of a Port DT of the future according to the Port of Rotterdam (PoR). As will be explained in this chapter, the vision and approach overlap with the approach taken within the current deliverable of using semantic models to enable interoperability.
- Chapter 3 provides a more detailed explanation on what semantic models (and ontologies) are and why they are relevant. We additionally argue for the usage of the Resource Description Framework (RDF) as serialisation of our choice.
- Chapter 4 presents the ontology development methodology and shows how it can be applied for a specific use case. The methodology incorporates a "modular" approach and is focused on reusing existing specialised models.
- Chapter 5 describes the core ontology that can be used for extension to support different (future) use cases.
- Chapter 6 contains the conclusions and provides next steps that can be taken within MAGPIE.

2. Port of Rotterdam Digital Twin Vision

In a white paper[3] titled "Digital Twins for the Port of the Future", PoR describes in collaboration with IBM Research, Cisco and ESRI a vision on the Port DT of the future. Its insights are relevant, as the purpose and requirements of a Port DT are presented from, but not limited to, the perspective of a port authority (i.e., PoR), a primary stakeholder in the domain. In this chapter, we briefly describe some key take-aways from the white paper document and relate these to the MAGPIE DT vision and the operationalisation of the language specification for the Port DT as presented in the current deliverable.

The white paper takes the perspective that a DT can be regarded as: "*a digital representation of a cyber-physical system that represents that system across its lifecycle from design through operations.*" (page 15). The digital representation is therefore not static but is continuously updated based on incoming information. In addition, the document argues that DTs are composable, which means that any DT may be composed of DTs of smaller components or objects (e.g., a port may be composed of multiple factories and assets, each having its own DT). Further, according to the document, artificial intelligence (AI) and automatic reasoning techniques are required for the continuous updating of the DT representation of objects. Finally, the document argues that the ultimate purpose of DTs is to help users to augment their capabilities for decision-making from design and manufacturing all the way through operation. This may, again, require AI and automatic reasoning techniques.

In MAGPIE, a DT is considered: "*a virtual and computerized counterpart of a physical system that can be used to simulate the system behavior considering different circumstances and environments, exploring the real-time synchronisation of the sensed data and is able to decide between a set of hypotheses which represent a better option.*". The focus on intelligent tooling where machine learning (ML) and computational models are utilized that augment human decision making and the real-time synchronisation of sensed data indicate that the MAGPIE vision largely overlaps with the vision presented in the PoR white paper.

The PoR white paper further envisions that the Port DT is composed of a federation of small building blocks that are referred to as "Smart Objects". These are the representations of the objects and assets in and around the port that may be connected to each other to exchange information directly: "*Events are no longer triggered by humans filling in forms, but by Smart Objects interacting among themselves, creating more lightweight, responsive, and autonomous processes*". (page 16). The white paper argues that data policies can then best be formulated on the level of these objects, as opposed to separately defining these per process or use case, which is currently the case. To operationalize this, the document proposes that a data representation needs to be formulated for these objects that extends across the multitude of processes and use cases where these objects are included. This is the suggestion presented in Deliverable D4.2 as well, where it is argued that the operationalisation of a DT of the port requires a language specification that specifies the meaning of the terms used throughout the systems, processes and applications involved.

To implement this language model, the white paper suggests to use semantic models. As further explained in Chapter 3, the current deliverable incorporates the same approach to initiating a language specification. Namely, the current deliverable describes a core ontology to be used and extended across applications in and around the port, such as the MAGPIE tools and demos. The MAGPIE core ontology can function as the semantic model suggested by the PoR white paper.

[3] S. McKenna, E. Rademaker (2020). Digital Twins for the Port of the Future.

3. Language Specification Approach

As explained in Deliverable D4.2, the operationalisation of a DT requires a language specification that provides a format for stakeholders to share and integrate data. Instead of aligning each data holder to each data user, applications then only have to be aligned once to the shared language specification when modelling, sharing and integrating information across systems. This greatly enhances interoperability between applications and eases the integration of new systems. To implement a language specification for the DT of the port, we follow the approach by PoR (see Chapter 0) and use a type of **semantic model**, namely **ontologies**.

In this Chapter, we first describe ontologies and why they are in our view suitable for implementing the language specification for the Port DT. We then briefly describe an existing ontology that can be used as starting point for further development of a core ontology. We end with identifying some of the requirements for a development methodology that can be used to further develop and extend a core ontology. This provides input for the next chapter (Chapter 0), where such a methodology is described.

3.1 Using ontologies

Deliverable D4.2 specifies the functional requirements for a language specification of the Port DT. The language specification should be: (1) globally available and follow the FAIR[4] principles, (2) globally unique, (3) reusable and extendible, and (4) descriptive rather than prescriptive.

As mentioned, we follow the approach by PoR and use a type of semantic model, namely ontologies. Ontologies are formal representations of knowledge within a specific domain as a set of concepts and the relationships that hold between these concepts. They provide a common vocabulary for defining, categorizing, and organizing information, making it easier to understand and interpret data that is defined in terms of the ontology. By using a shared ontology, data can be consistently represented, reducing ambiguity and allowing for more accurate comparison and integration of data. Ontologies are therefore well suited to help in the process of modelling, exchanging and integrating data from various sources.

In our view, the principles underlying ontologies implemented using RDF[5] (Resource Description Framework) closely follow the functional requirements of the language specification for the Port DT. We therefore propose using ontologies adhering to RDF (RDF ontologies):

- RDF ontologies follow the principles of **FAIR** and **open data**. The elements are identified and accessed through their IRI (Internationalized Resource Identifier, an extension of the common URLs). In addition, the use of IRIs makes RDF compliant with the principles of linked-data[6], as data elements and models can be interlinked though their IRIs.
- IRIs are **globally unique**. This means that every element is necessarily identical to elements having the same IRI, which prevents semantic confusion. An added benefit of IRIs is that their name can also indicate ownership of the data. This stems from their origins in semantic web research, where we want to give a computer readable meaning to every data element on the web. So, data elements of a particular transportation company can all be identified by its starting IRI, such as: <https://www.transportation-company.org/data/truck#adbd-badb-adbd-daba>.

[4] the FAIR data principles (force11.org)
[5] Resource Description Framework (RDF) (w3.org)
[6] Linked Data - Design Issues (w3.org)

- RDF ontologies can be easily **extended** to include new concepts and relationships whenever additional domain knowledge needs to be represented and shared among stakeholders. For example, if a new term or concept emerges in a particular domain, it can be added to the existing ontology and thereby integrated into the semantics for data sharing. This extensibility is facilitated by the modular architecture of ontologies. Namely, ontologies are typically composed of a set of smaller, more specialized modules, each of which is developed to represent specific aspects of a domain (e.g., see Figure 1). These modules can be added, removed, or modified as needed, allowing the ontology to be tailored to the specific needs of the user, such as the stakeholders within a given data space. The assets, tools and techniques situated and used within any Green Port of the (near) future, such as new energy consumption, storage, and supply modes, will continuously be modified and improved in the future. The extensibility of ontologies is therefore well suited to keep up with such innovations in the port.

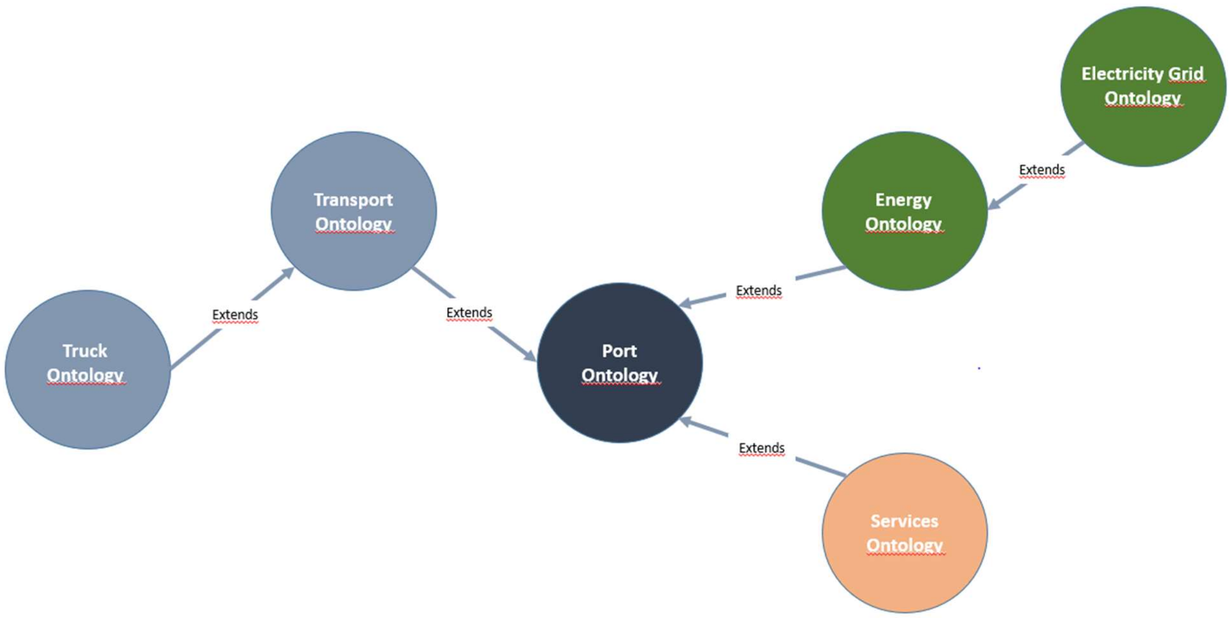


Figure 1. An example showing how several semantic models can extend each other. Each of these models can be developed by different parties and in different projects.

- RDF Ontologies are highly **reusable**. The reason is that since ontologies represent a standard vocabulary and set of relationships for a particular domain, ontologies can be (re-)used by any application or system that (partially) operates within that domain. This facilitates interoperability between applications and reduces the costs and effort of developing and integrating new applications.
- We view the use of RDF ontologies as **descriptive** rather than prescriptive. This means that stakeholders may agree upon recommending a particular semantic model, but parties are always free to model their data according to the model of their preference. Deviations between data models used by different parties ultimately interferes with the added benefit of interoperability provided by semantic models. If stakeholders choose to deviate from the recommended data model despite the interoperability advantages, it is their responsibility to govern the alignment between the different models.

3.2 Starting point for a core ontology

In order to model and share linked data in the port using RDF ontologies, we develop a generic overarching core model that describes the principal high-level concepts related to transport and logistics activities in the port. In line with Figure 1, we extend this core model in a modular way by reusing a wide range of more domain-specific models that describe knowledge in sub-domains within the transport and logistics field, such as information on particular modalities of transport. The core model and its modular extensions can be reused and possibly further extended by parties possessing expertise on the domain of their specific use case or tool.

As a starting point for building an overarching core ontology for the port, we modify and extend the FEDeRATED ontology[7] (see Figure 2). The main function of the FEDeRATED ontology is to support inter-operability between a broad range of domain-specific ontologies in the transport and logistics sector. For this purpose, the ontology contains many of the primary concepts relevant to transport and logistics operations and explicitly excludes domain specific information such as details that only apply to a particular modality of transport. The scope of the FEDeRATED ontology is therefore well-suited as starting point for developing a core ontology for the logistics and transport domain in the port. Further information and specification of the FEDeRATED ontology is provided in Appendix 1.

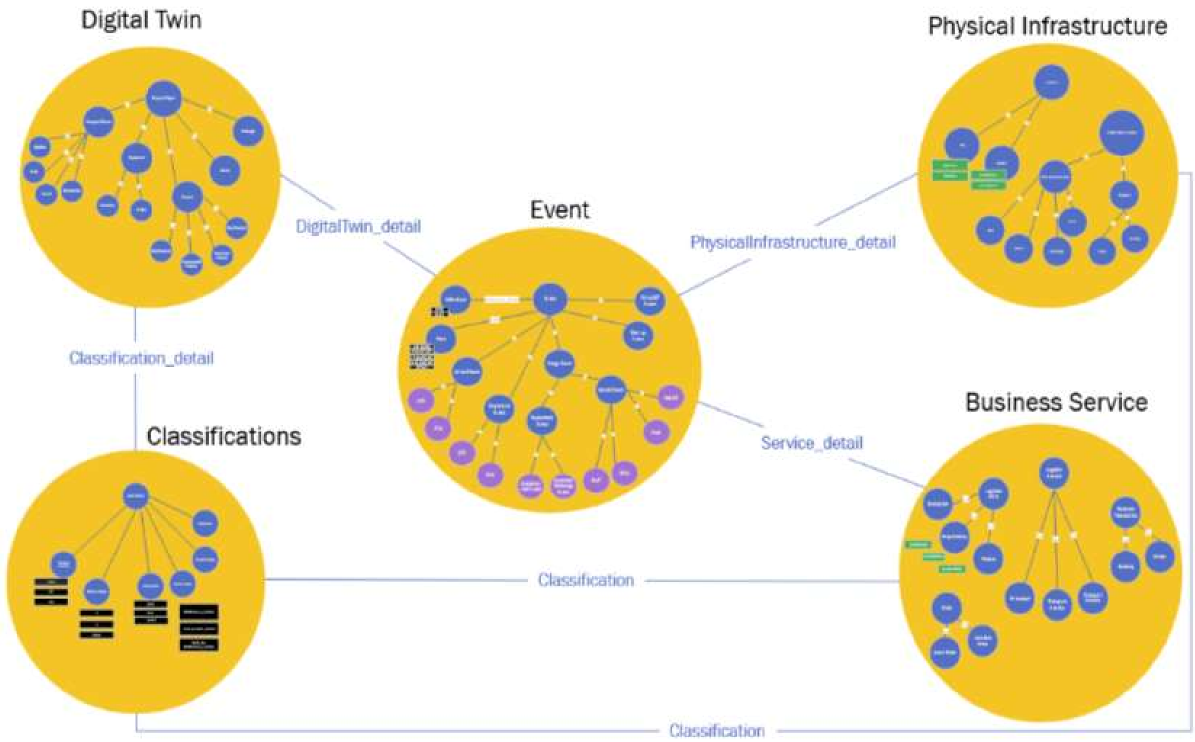


Figure 2 Some of the main modules and concepts in the FEDeRATED ontology.

3.3 Requirements for an ontology development methodology

Development of RDF ontologies and other semantic technologies for the port requires proper guidance. First, because they are innovative technologies stemming from various academic disciplines such as information science, philosophy, and AI. Second, the requirements of

[7] Documentation can be found here: Developer Portal (federatedplatforms.eu)

future tools and use cases that have to be integrated and supported by the semantic model are not fully known in advance. To guide future efforts to implement, improve or extend the expressivity of the core RDF ontology, the current document presents a concise methodology that partners and (future) stakeholders can use to integrate their use cases and tools to the semantic architecture of the Port DT.

Various ontology development methodologies have been proposed in the academic literature throughout the years. The work by Gruninger on Ontology Development 101[8] and the work on Competency Questions is usually indicated as the starting point. Other major development methodologies are SAMOD[9], SABiO[10], the Linked Open Terms methodology[11]. These methodologies document the complete lifecycle of identifying a domain and consequently developing a semantic model for that domain. The field of ontologies is constantly changing, so the important aspects of the available methodologies are similarly changing.

The MAGPIE project is characterized by two aspects that indicate which kind of methodology is needed:

- The scope of MAGPIE is substantial. It extends beyond just port logistics to energy and environmental concerns as well. It will be hard to find a domain expert whose expertise encompasses all those areas. We therefore do not expect to deliver *one* ontology to model all concepts relevant across the MAGPIE tools. Instead, we expect to deliver an ontology consisting of links between numerous sub-ontologies that model the various domains and subdomains involved. For example, diverse concepts such as charging stations, trains, greenhouse gas emissions, and port terminals all may require a different collaborative process with the domain expert.

  Additionally, some of these models, standards, and protocols may already have been developed independently of the MAGPIE project. In Appendix 1 we provide a list of initiatives where ontologies are developed for related domains, such as energy charging, the railways, and trucks. Reuse of existing models is a design philosophy of ontologies, following the trend of increasingly smaller models aligning or extending existing ones. We call the construction of an ontology out of a multitude of small potentially independently reusable parts ontology *modularisation*. The development methodology explicitly needs to support this.

- Furthermore, the MAGPIE DT is focused on supporting the needs of the various tools, demo's, and use cases. The primary goal of the ontology is to facilitate data sharing instead of providing a philosophically grounded conceptual model of the port. Being able to answer the data requests by the various tools ultimately is the requirement for the ontology. It should be competent to answer all those questions.

  The ontology development technique highly relevant in this aspect is the Competency Question. That is the idea of having the tool designer formulating all requirements as natural language. These can be used to guide the ontology development process and to validate at the end of the process that the ontology can indeed answer those questions. The MAGPIE methodology should therefore additionally support the usage of competency questions.

[8] Ontology-Development-101 (protege.stanford.edu)
[9] SAMOD (essepuntato.it)
[10] SABiO (nemo.inf.ufes.br)
[11] Linked Open Terms (lot.linkeddata.es)

Competency Questions have long been recognized as important for describing the functional requirements in collaboration with domain experts and end users. Modularisation is similarly regarded as benefit for the (re-)usability and flexibility of the ontology. In existing ontology development methodologies, however, both aspects feature primarily in the initial stages of development, with competency questions sometimes making a reappearance in the evaluation phase. The methodology we propose positions the added benefits of explicitly including competency questions and modularisation throughout the ontology workflow. We additionally make it better usable for non-experts.

4. Ontology Development Methodology

In this chapter we describe the methodology that supports the tool and demo developers to extend the ontology to their requirements. In the appendix of Deliverable 4.1 the preliminary form was shared. In the current chapter we elaborate on the procedure.

As described in Chapter 3, the methodology intends to facilitate the reuse of semantic technology. Following the approach from the literature the method is separated into a requirements phase, an implementation phase and a validation phase. The requirements phase is a use case analysis that requires the least amount of semantic knowledge. Instead, it is a structured analysis to specify the requirements the ontology engineers should model. Only in the implementation phase we move towards the implementation of the model; a task for the semantic specialist. The validation phase is aimed at testing whether the developed ontology meets the requirements of the use case.

The results of applying the requirements phase steps should be a further specification of the use case beyond what is described in Deliverable 4.1. The preliminary definition of the data needs and functional requirements has been specified there. However, the level of detail is often insufficient for semantic modelers to implement an ontology extension based on it.

To explain our ontology development methodology, we first describe the different roles that are involved when carrying out this methodology. We then describe the steps that are to be carried out. Finally, we end with a brief example where the application of the methodology results in a conceptual model that satisfies the functional requirements of a use case.

4.1 Roles

The division of work already implies the various roles involved in developing a semantic representation. For each of the steps in the methodology we specify the roles expected to be involved.

- The **User** has a clear idea about the problems to be solved, the goals of the ontology and the tools that will eventually use the model. The user communicates this via a set of both technical and functional requirements, drawing from their expertise of the domain. This role will in the MAGPIE project typically be fulfilled by people from domain companies or researchers whose role is close to the domain. In general we can expect the person fulfilling this role to play a major role at the port.
    - A specialised user is the **application developer**. They develop the tool that processes the ontology-based data drawn from the DT. This role can be fulfilled by employees of the involved companies that develop the application. The role requires expertise in software development.
- The **ontology specialist** is the one who is specialised in leveraging semantic technologies, such as ontologies, for data interoperability. Three specialised roles can be identified, that can potentially be fulfilled by the same person. In the MAGPIE project all ontology specialists are employees of the RTOs involved.
    - The **ontology designer** creates a conceptual model based on the requirements specified by the user. This role requires in-depth knowledge of knowledge representation.
    - The **ontology engineer** is a technical role that requires in-depth knowledge of the serialisation language, which is RDF in our case. The ontology engineer transforms the conceptual model of the ontology designer into the formal ontology language.
    - The **ontology tester** role verifies in collaboration with the ontology user whether the requirements have been properly implemented. This role requires

a mix of domain knowledge, knowledge representation expertise, and formal ontology expertise.

- The **domain expert** provides detailed knowledge about all kinds of entities and properties in the domain of interest. In the MAGPIE project these domains would be, for example, the port, logistic activities, energy usage, and environmental features. Employees at the port (e.g., PoR or Port of Sines) or energy companies would fulfil this role.

The steps described in the next subsection that are to be taken when carrying out the methodology primarily specify the way in which the user, application developer, and domain expert should provide their requirements to the ontology expert.

4.2 Steps

4.2.1 Requirements phase

We identify five steps to clarify the ontology requirements: use case description, specification of data sources available, competency questions, architecture and flow of control, and return data.

Use case description

The user should provide a description of the tool or demo whose interoperability goals the ontology should support. In D4.1 several tools already provided such an overview. For example, the Smart and Green Logistics Tool is described as follows:

"The Smart and Green Logistics tool being developed under MAGPIE 4.5.3 will essentially be a decisions support system (DSS) that will provide recommendations to the end-user of the tool on the choice of transportation mode(s) (i.e., a choice between rail, trucks, barge, or their combinations) as well as carrier(s) along with an estimate of charging requirements per mode. The objective of delivering such recommendations as the tool's output is to balance the cost and emissions generated over the entire 'container' transport chain from the Port to the destination in the hinterland. While the scope of the tool is currently limited to the operational level decisions that would be made 24-48 hours in advance, the tool can also be extended to inform and evaluate tactical and strategic level decisions in the future by simulating a range of 'What if?' scenarios. The tool makes a few assumptions with respect to the scope: a) we only consider shipment of containers across the network and b) operations undertaken within the port (i.e., in-port) operations are not considered in detail. The tool developed within 4.5.3 uses (as input) the output of the GHG tool (developed in 4.5.1) i.e., the allocation of emissions to the different transport modes. This information provided by the GHG tool helps drive the decisions of selecting the greenest and cheapest transport chain from the port to the final container delivery destination. The output of the tool is fed back into the GHG tool for a detailed emission calculation. In a similar way, the tool uses inputs from 4.5.2 regarding the availability of energy, while the results are used by 4.5.2 to perform the energy matching in detail." [12]

In collaboration with the PoR, we identified a use case that contains the necessary interaction between a future e-barge and a Shore-Side Electricity (SSE) module in order to determine whether they meet each other's requirements for connection. In this use case, the e-barge and SSE module function as two "smart objects" (as envisioned in Chapter 0) that directly communicate to each other. We refer to this use case as the **Shore-Side Electricity use case** and it can be described as follows:

[12] MAGPIE Deliverable 4.1, section 5.2.3

In the Shore-Side Electricity use case, an e-barge and Shore-Side Electricity (SSE) module communicate to each other to determine whether they meet each other's requirements. The e-barge contains containers and needs to recharge its batteries at or near a terminal in a port. It has identified several SSE modules located at different berths at the terminal in the port that, depending on the characteristics, requirements and planning of the SSE modules, it may potentially use to charge its batteries. In order to know whether one of the SSE modules is available and suitable for the e-barge, the e-barge and SSE module exchange information.

We use the Shore-Side Electricity use case in the remainder of this chapter to explain and showcase our ontology development methodology.

Data sources available

Next, an overview of the various data sources that are relevant and available for the use case should be provided by the user, potentially in collaboration with the application developer and the domain expert. The overview of data sources should give an indication of the interoperability problem at hand. The table below provides an imaginary example of a specification of data sources that would be relevant in the Shore-Side Electricity use case. Additional elements such as access rights, availability, and ownership may also be relevant depending on the specific use case.

Table 1. Overview of data sources.

Identifica-tion	Name	Schema available	API available	Example data	Live data	Security	Located at
DS01	Shore-side e-lectricity (SSE) module	Json	Yes	Yes	Yes	Password protected	Terminal in PoR
DS02	e-Barge	Json	Yes	Yes	Yes	Password protected	e-Barge carrier

Legend:

1. Identification: a simple code for unique identification of this data source.
2. Name: a human readable name
3. Schema available: is there a data schema available, such as XML schema or json schema?
4. API available: Do we have access to an API? This may both be real data as well as fictional data.
5. Example data: is there an example data set available?
6. Live data: do we have access to live data?
7. Security: is there some additional security that inhibits access?
8. Located at: where should we access the (example) data set or API?

Competency questions

Formulating the functional requirements of an ontology as a set of competency questions is the recommended way of working following the scientific literature, as discussed in Chapter 3. Each question is a natural language question identifying a single functional requirement of the ontology model. If we can transform the natural language question into a formal query using exclusively ontology concepts, the requirement is considered satisfied.

It is important to distinguish between requirements for the ontology and requirements for an application, as these may sometimes be different. This distinction may be difficult for a user to identify, but should be the responsibility of the ontology specialist. The ontology is supposed to contain the necessary data concepts and properties needed for data communication between parties and applications. The application that consumes data that is described in terms of the ontology may perform additional computations or modifications on the data. When describing competency questions, it is therefore critical to distinguish between the questions that should be answered by the data described in terms of the ontology and the questions that should be answered by an application that consumes the ontology-based data. The output of an application may, of course, be incorporated in the semantics of the ontology such that it can easily be shared with other parties using the ontology.

For each competency question we expect the natural language expression, an example answer, an answer schema, and optionally a complexity level. The list of functional requirements should be provided by the user in collaboration with the application developer. They have a proper view on which questions the tool expects the ontology model to be able to answer. In the next few steps we will see a close connection between the competency question and the architecture of the tool.

The domain expert and the ontology expert are in a supportive role for this task. The example answer or answer schema may be provided by the domain expert if it's too technical for the user. The ontology expert is expected to review the list of competency questions for completeness and usability. Below is an example list of competency questions provided by PoR for the Shore-Side Electricity use case.

Table 2. List of competency questions.

Identification	Natural language	Example answer	Answer schema	CQ level (optional)
CQvessel.01	What type of vessel is this?	e-Barge	String with vessel type.	Low
CQvessel.02	What is the battery level of this vessel?	75%	Percentage.	Low
CQvessel.03	What electricity capacity will this vessel require?		Voltage (kV), expected minimum (MW), maximum (MW), peak (MW), average (MW), frequency (Hz) and stay length.	Low
CQvessel.04	What are the names of the contact persons for incident management procedures?	First Name, Second Name.	List with contact person names as strings.	Low
CQvessel.05	What are the required		Voltage (kV), frequency (Hz)	Low

	specifications of an SSE module for this vessel?		and connector type.	
CQvessel.06	Who is responsible for connecting the vessel to the SSE module?	Name	Name of the person as string.	Low
CQvessel.07	Where are the connection points on this vessel?	Location	Location of connection points.	Low
CQsse.01	What are the SSE modules at this location?	ex:SSEModule_1, ex:SSEModule_2.	List of SSE module objects.	Medium
CQsse.02	What is the earliest available SSE module for this vessel at this location?	ex:SSEModule_1	SSE module object	High
CQsse.03	At what berth should I be?	ex:Berth_1	GLN number of berth, location of berth.	High
CQsse.04	What are the required specifications of a vessel to connect to this SSE module?		Voltage (kV), frequency (Hz) and connector type.	Low
CQsse.05	What are the names of the contact persons for incident management procedures?	First Name, Second Name.	List with contact person names.	Low
CQsse.06	Where are the connection points of this module?		Location of connection points.	Low
CQsse.07	Who is the energy supplier?	Company Name	Name of the company.	Low
CQsse.08	How do I pay for the charged electricity?	Terminal fee	One of the following: {subscription, pay per use, port charges, terminal fees}.	Low
CQsse.09	What is the cable management system of this SSE module?		Type of connection points.	Low

Legend :

1. Identification: please provide a identification for each CQ. A useful standard way would be to have all follow: CQ<highLevelConcept>.<counter>. In this example we have used two high-level concepts: vessels and Shore-Side Electricity (SSE) modules.
2. Natural language: please provide the CQ as a normal natural language question, describing the question that one of the data sources should provide an answer to.
3. Example answer: please provide a valid example answer
4. Answer schema: if available, please provide the schema or constraints that the answer should follow. For example, boolean, string, integer between 0 and 100, or list of address objects.
5. (Optional) CQ level: if you're familiar with competency questions levels, please provide an indication whether the CQ is on the detailed atomic level, on the middle connectivity level, or on the high use case level.

Architecture and flow of control

In this section we expect a flow of all exchange between the various tools and the data sources where it draws its information from. This should be a combination of the two previous items. The data source should feature as the sources of information. The competency questions should function as the queries that are being sent to the various data sources to draw information from them.

Return data

A table that indicates in terms of the ontology which kind of data the tool feeds back into the DT. This section is only applicable if the tool produces data in addition to consuming data.

4.2.2 Implementation phase

In the implementation phase the roles of the ontology designer and ontology engineer transform the requirements into a conceptual model that follows ontological principles. The conceptual model is then serialized in a semantic web-compliant language, such as RDF/XML, TTL, or JSON-LD. This phase is extensively described in various handbooks, tutorials and methodologies. We recommend as introductory handbooks Semantic Web for the Working Ontologist[13] by Dean Allemang, the Introduction to Ontology Engineering[14] by Maria Keet, The Knowledge Graph Cookbook[15] by Andreas Blumauer, and The Web of Data[16] by Aiden Hogan. These can serve as a way for the ontology engineer to increase their expertise.

The implementation and publication that occurs in this phase, are extensively described in the Linked Open Terms methodology[17], in the Ontology Implementation and the Ontology Publication phases. We recommend users to follow these steps.

4.2.3 Validation phase

In the validation phase the goal is to check whether the ontology extensions indeed model the requirements that were identified. This consists of a formal verification part where the natural language questions are translated to a formal query language. If a formal query using ontology constructs can be designed that replicates the competency question, we say

[13] Semantic Web for the Working Ontologist (sciencedirect.com)
[14] An Introduction to Ontology Engineering (open.umn.edu)
[15] The Knowledge Graph Cookbook - Recipes That Work (poolparty.biz)
[16] The Web of Data (Book) (aidanhogan.com)
[17] LOT - Linked Open Terms (linkeddata.es)

the ontology is "competent" to answer that question. This check verifies whether the ontology was built correctly following the requirements.

The proper validation should happen subsequently via an implementation of the tool that was intended to use the ontology as part of its interoperability solution. This step validates whether the right ontology was identified in the requirements. This step also includes the integration of the tool with the DT environment as specified in deliverable D4.2.

The existing Linked Open Terms methodology has been expanded with a module on facilitating application development.[18] The work specifies how the existing set of competency questions can be analyzed to form the basis of a set of APIs that allow a tool to communicate using ontology-based data formulation.

SPARQL[19] is the standardized query language that can be applied to data formatted in RDF. It can be used to translate the original competency questions into a language that is able to query RDF data. Using these SPARQL queries and applying it to RDF data formatted according to the ontology, it can be validated whether the ontology meets the functional requirements posed by the competency questions. Below, we translated some of the competency questions presented before into a SPARQL query template that can be instantiated to answer the particular query.

Table 3. List of SPARQL queries.

Identification	Natural language	SPARQL query template
CQvessel.01	What type of vessel is this?	$this rdf:type ?vesselType . ?vesselType rdf:subclassOf dt:Vessel .
CQvessel.02	What is the battery level of this vessel?	$this magpie:batteryLevel .
CQsse.01	What are the Shore-Side Electricity (SSE) modules at this location?	?asset rdf:type magpie:ShoreSideElectricityModule ; magpie:locatedAt $thisLocation .
CQsse.02	What is the earliest available SSE module for this vessel at this location?	This question involves a computational component, so the answer cannot be derived solely through a SPARQL query. Using custom functions within the query that have to be implemented via the tools, the following query could solve this: ?sse rdf:type magpie:ShoreSideElectricityModule ; magpie:locatedAt $thisLocation ; magpie:slot ?slot . ?slot magpie:isAvailable "true"^^xsd:boolean .

[18] Extending Ontology Engineering Practices to Facilitate Application Development (link.springer.com)
[19] SPARQL Query Language for RDF (w3.org)

		FILTER (appropriateChargingConfigurations(?sse, $thisShip) .

Whenever it turns out not to be possible to construct a SPARQL query that returns the desired output of the corresponding competency when applied on ontology-formatted RDF data, the ontology might have to be adapted or extended. This means that the ontology designer and engineer return to the implementation phase and make appropriate adjustments. It might, however, also be possible to review the competency question at hand and reconsider, together with the domain expert, whether the competency question is really a requirement to the ontology as opposed to a requirement of an application that consumes the ontology-based RDF data and performs additional computations to derive the desired information.

4.3 Example application

As an example of how to apply the methodology and its competency questions for modelling, we provide three conceptual models as a set of diagrams (see Figure Figure 3, Figure 4 andFigure 5) that, when combined, are able to answer the competency questions provided by PoR for the Shore-Side Electricity use case. The conceptual models can be designed by the **ontology designer** in collaboration with the **user**. When translating the conceptual models into a recommended RDF serialisation, the **ontology engineer** should take the additional principles of model reuse into account, which are abundantly documented in the various handbooks (see section 4.1.2).

Many concepts can be reused from the various related ontologies mentioned in Appendix 1. For example, the Legal Person mentioned in both models can be taken from the FEDeRATED model and from the FOAF ontology, the various codelists can be implemented using SKOS, and the location can be identified using most of the geography vocabularies. In addition, the applicability of SAREF4ENER for the power profile of the charging of a vessel seems interesting to investigate.

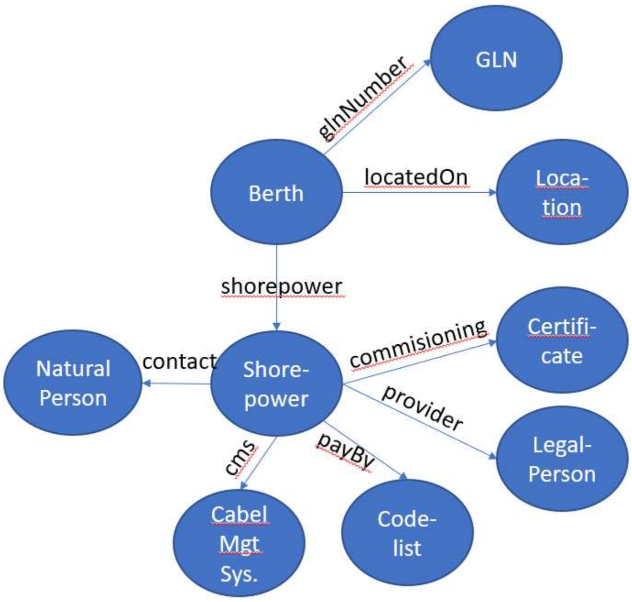


Figure 3 Shore power model answering CQs.

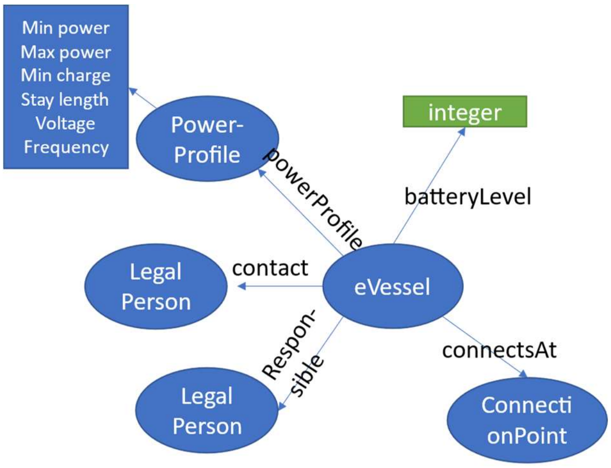


Figure 4 Chargeable vessel model following CQs.

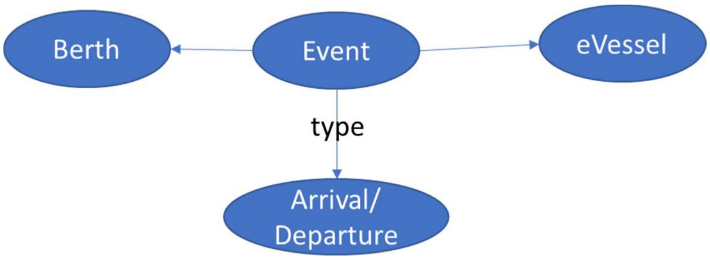


Figure 5 Linking the vessel and berth via arrival or departure events.

During the course of the project we have applied the procedure on several other use cases as well, ranging from scenario's made-up on the spot during tutorial sessions as well as several deep-dive sessions into applying the procedure for the use cases, tools, and demo's indicated in D4.1. Appendix 2 provides the results by INESC TEC that applied the methodology for a loading and charging use case in the port. It shows the language specification requirements of their use case as well as an extensive specification of a proposed extension of the MAGPIE core ontology (see next Chapter) to meet these requirements.

5. Magpie Core Ontology Specification

In this chapter we describe the MAGPIE core ontology that can be used to connect to the DT architecture. The core ontology can be regarded as a "multimodal logistics modular core ontology". Let us unpack these terms:

- **Multimodal logistics** means that the ontology is intended for data sharing across logistic modalities: air, rail, road, sea, inland waterways, and pipelines. It usually is a challenge to share data across modality boundaries, because of differences in format, meaning, and architecture. Aligning existing models from the various modalities via an ontology attempts to solve the differences in meaning and format. Sharing data via the DT attempts to solve the differences in architectures.
- **Modular** means that the ontology consists of many independent parts. The ontology therefore mostly consists of alignments among existing models instead of developing additional large models.
- **Core ontology** means that this ontology is intended to be further specialised based on requirements from various sources, which are the tools and demo's in the case of MAGPIE. For example, the ontology is intended to be extended for specialised uses of electrical vessels and electrical trucks, or whichever requirements may come up in a future tool.

The MAGPIE core ontology is assembled from various building blocks that all descend from different ontology or standardisation initiatives, listed in Appendix 1. The approach of providing alignments between the modules follows the ontology principles of extensibility and reusability. The MAGPIE scope transcends the logistics domain to include also concepts of sustainability and energy, so we reuse existing models from those domains. We first show the modularisation overview of the core ontology and then describe each of the module components.

5.1 Modularisation

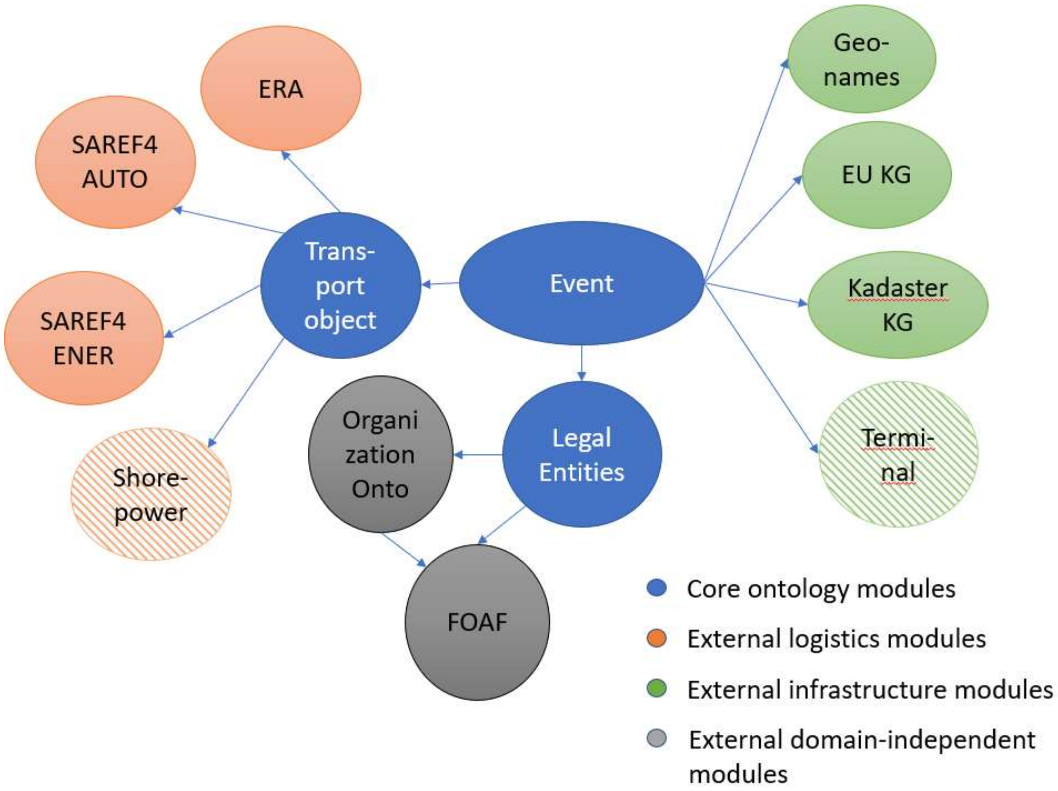


Figure 6 Overview of modules of the core ontology (dark blue) and alignment with external modules: logistics domain ontologies (orange), geography and infrastructure ontologies (green), and domain-independent ontologies (gray). Two examples of how MAGPIE may extend this core are highlighted with diagonal stripes.

Figure 6 shows the modularisation overview of the core ontology and its alignments with external modules. The modules coloured in dark blue are the modules described in section 3.2 as stemming from previous work on a DTLF ontology: Event, TransportObject, and LegalEntity. These are further described in the following subsections.

Additionally, in orange, green, and grey we indicate the various ontologies we reuse to both increase the scope of the core model and to reuse proper ontology work of peers. Some ontologies provide a detailed view on the domain: SAREF4AUTO provides a detailed model of trucks, and the ERA vocabulary provides a detailed model for trains. The various geography and infrastructure vocabularies facilitate the reference to real-world locations. These vocabularies have pros and cons, which we elaborate in the respective section. The scope of the ontology is broadened by the inclusion of standardised energy models, such as SAREF4ENER. This module can indicate how to specify the energy usage constraints.

Finally, the modularisation contains some modules that the MAGPIE project expects to deliver based on the goals of the various demos and use cases. The development of these modules is ongoing, and is supported by the methodology described in the previous section. For example, it is expected that following the methodology for the e-Truck demo will result in an extension of the Truck model for electrical trucks. It is also expected that various use cases will want to design a shore power model, since various use cases seem to require that. Optimally, these models are aligned and merged with each other, which is also supported by the methodology of the previous section. These extensions are expected to be documented in the deliverables of the respective tools, demos, and use cases.

5.2 Modules description

In this section we give an overview of the existing modules including the reasons for the alignments that are indicated in Figure 6.

Event

An event indicates something that happens during the logistic process. Simple examples are the arrival/departure of a transport means or the load/discharge of a container, but also the merging of goods or the sealing of a container may be considered an event. The module may also be arbitrarily extended to cover administrative events, such as orders and bookings.

The centrality of the Event module in the modularisation is in part a leftover from the FEDeRATED ontology we used as inspiration. The digital architecture in which the FEDeRATED ontology was used forced partners to share data via events. In the digital architecture we develop for the Port DT, we loosen this constraint. Any data object can be shared via the DT. Still, the event remains such a central concept that it is also a major part of the conceptualisation of the core ontology.

Some of the main event classes that we include are shown in Figure 7 and described in Table 4.

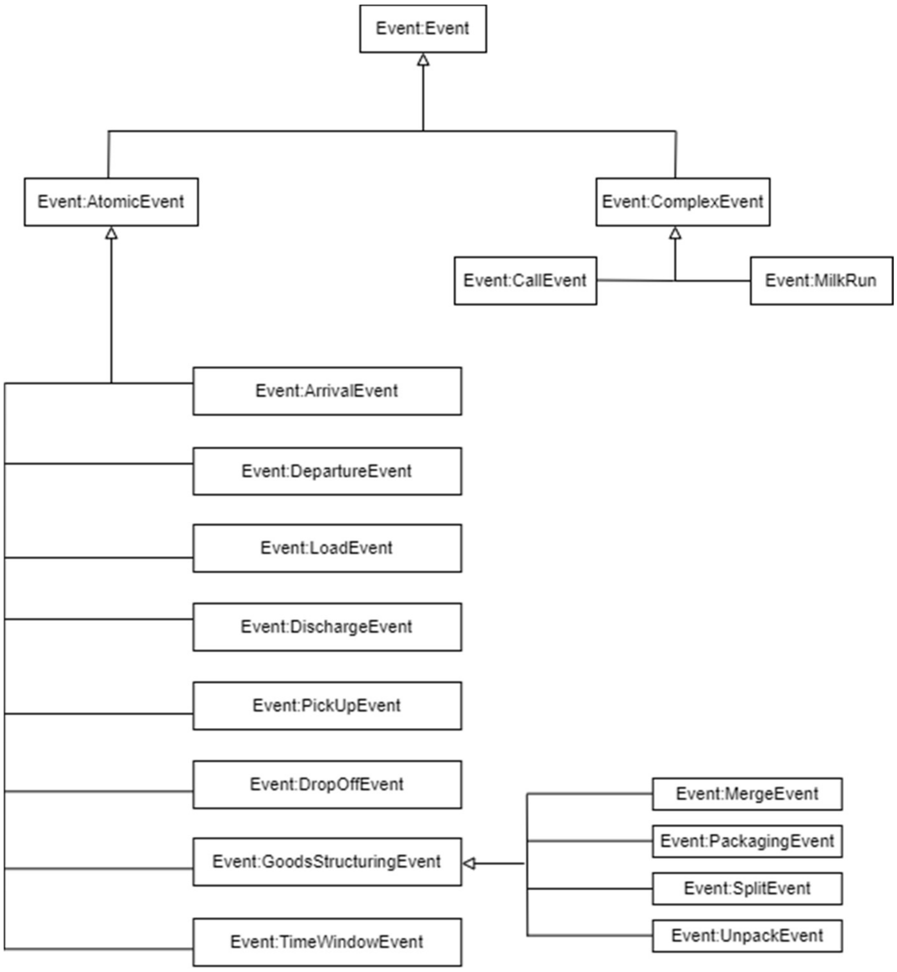


Figure 7. Some of the main classes in the Event module. An empty arrowhead indicates a sub- superclass relationship.

Table 4. Description of some of the main classes in the Event module.

Class	Description
ArrivalEvent	The arrival of a transport means at a location.
DepartureEvent	The departure of a transport means at a location.
DischargeEvent	The discharge of cargo from an equipment device.
DropOffEvent	The drop off of an equipment device by a transport means.
Event	An event reflects any activity or change in the real world, creating, updating, or 'deleting' an association between objects or entities.
GoodsStructuringEvent	An event that changes the way in which multiple transportable objects are combined or structured.
LoadEvent	The loading of cargo on an equipment device.
MergeEvent	Merge two sets of goods into a single combined set of goods.
PackagingEvent	An event association between transportable objects (Goods or Products) to show that one has been packaged into the other.
SplitEvent	A split of a transportable object into a set of smaller transportable objects.

UnpackEvent	The unpacking of goods/packaged products into another packaging type of these goods/packaged products.

The specific meaning of an event is defined by its properties. The core properties we provide for events in the Event module are listed in Table 5. One of the main properties is the DataTimeType, which indicates the type of timestamp that is linked to the event. For example, a DateTimeType "actual", indicates that the event has actually happened at the specified timestamp, whereas a DateTimeType "estimated" indicates that the event is estimated to happen at that time. Another main property is the Milestone of an event, which is either "start" or "end".

The meaning of an event is further indicated by the associations it has with other entities. For example, an arrival event in which a transportation object *vessel_x* and a physical infrastructure *terminal_y* are involved means that *vessel_x* has arrived at *terminal_y*. As shown in Table 5, these associations are also modelled as properties of the event. The entities and objects that may be associated with events are modelled in other modules, which we describe in the next subsections.

Table 5. Core properties of events in the Event module.

Property	Description	Datatype
hasDateTimeType	Indicates the type of the timestamp that is associated with an event, namely: "actual", "estimated", "expected", "planned", or "requested".	
hasSubmissionTimestamp	Indicates the timestamp when the event was submitted.	xsd:dateTime
hasTimestamp	Indicates when the event has occurred or will occur.	xsd:dateTime
involvesLegalEntity	Indicates which legal entity is associated with an event.	
involvesPhysicalInfrastructure	Indicates which physical infrastructure is associated with an event.	
involvesTransportationObject	Indicates which transportation object is associated with an event.	

Legal Entities alignments

The Legal Entities module contains various classes to model the legal actors involved in a use case (people or companies), location roles, and other business-related submodules. In the core ontology the submodules present limited functionality, which motivated us to align standardized, industry specific modules that better model business service concepts.

First, we have aligned the 3 ERA modules (Manufacturer, VehicleKeeper, InfrastructureManager) by considering them subclasses of the core ontology module LegalEntities. These alignments enable a representation of a railway-oriented organisation that includes both railway associations, based on the aligned ERA modules specification, as well as the commercial details based on the foaf[16] module, part of the core ontology.

Transportation modalities alignments

The Transportation Object module contains the concepts to describe objects that are "moving around" during logistical operations, such as vehicles and packages. The concept of Transportation Object closely relates to the idea of Smart Objects that are used in the conceptualisation as envisioned by PoR (see Chapter 0). Transportation Objects, however, would link to the Smart Objects that are moving around, as opposed to objects that are part of a physical infrastructure (which are modeled in another module, see next subsection).

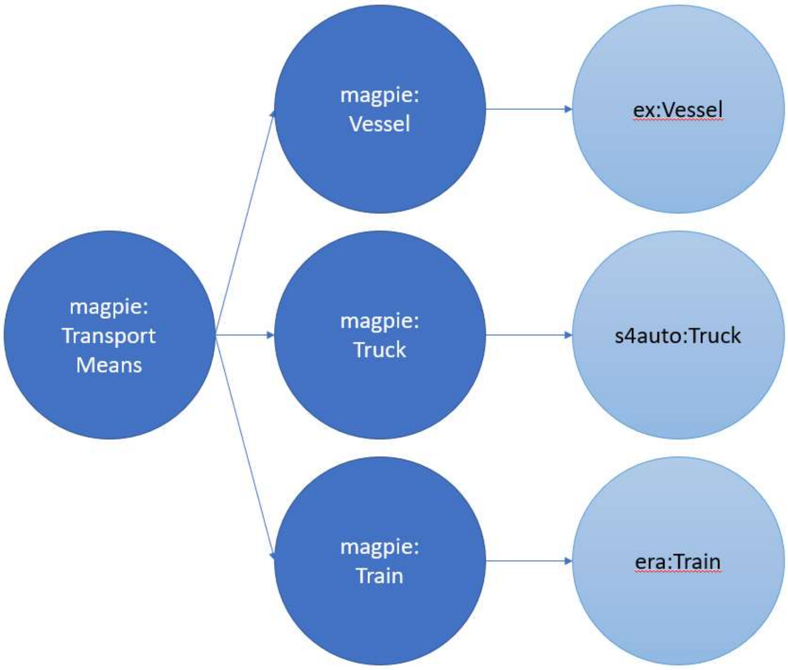


Figure 8 Overview of transport means alignments.

An important type of Transportation Objects is Transport Means, which are the vehicles used for transport. **Fout! Verwijzingsbron niet gevonden.**For two separate transportation modalities, namely road- and railway-based transport, we have created alignments to existing models relevant to MAGPIE (see Figure 8). The external models that we have aligned are SAREF4AUTO, a model on automotive concepts and ERA, which models the railway domain. Additional types of transport means could be extended in another dedicated extension. For example, we envision a third alignment to express water-based transport, a fourth alignment on transport via planes, and potentially a fifth on transport via pipelines. These alignments enable the association between events, the involved transportation objects, and the geographical locations involved. Each of the external modules we align with is described in Appendix 1.

Infrastructure and geography

There are several related initiatives that model relevant infrastructural and geographical information in ontologies. We align and reuse some of these models.

First, the alignment of the SAREF4AUTO infrastructural modules to the core ontology module Physical Object enables the association of logistic events and industry specialised digital representations of physical elements part of a route. This, in turn, supports automotive automation and route scheduling. Moreover, the alignment of the ERA infrastructural modules enables specification of logistic events including railway infrastructural objects described using low-level, locomotive engineering information. The ability to express logistic events that include low-level, domain-specific information about the infrastructural objects involved leads to a more comprehensive digitalization of all physical factors influencing the actual execution of the event.

Second, Geonames provides a Linked Data source covering all infrastructural and geographical objects around the world. The EU Knowledge Graph has a similar goal of providing a Linked Data source containing information about geographical and infrastructural objects across Europe. The Kadaster KG is yet another Linked Data source that provides infrastructural and geographical data of The Netherlands. This wealth instead of scarcity of semantic data can be problematic. Geonames, Kadaster and the EU KG use different models, so a tool expecting a geographical location following Geonames cannot directly be applied on data drawn from Kadaster.

A list of pros and cons for all four initiatives is provided in Table 6.

Table 6 Advantages and disadvantages of different semantic model initiatives describing geographical information.

Initiative Name	Pros	Cons	Example implementation
Geonames	- SPARQL endpoints & interface - All possible interactions are documented - Up to date with latest changes	- Paid service for fault-less and advanced querying - Peer reviewed wiki-like source of information, no standardisation	Event01 Event:involvesPhysicalInfrastructure https://sws.geonames.org/2747891/
Kadaster	- Juridical information on locations - Detailed structural and geometrical information of all locations - Extensive querying and visualisation exploratory tooling	- All attributes' names are in Dutch - Limited to only locations in the Netherlands - Paid service for legally quotable information	Event01 Event:involvesPhysicalInfrastructure https://data.kkg.kadaster.nl/id/gemeente/0599
Schema.org	- Easy to explore attributes - Based on a basis schema which enables interoperability with other industry extensions	- No SPARQL / user interaction - No data hosted	Event01 Event:involvesPhysicalInfrastructure schema:Destination . Schema:Destination a schema:location ; Schema:Destination schema:address Rotterdam; Schema:PostalCode 3011.
EU KG	- Free querying service - European Union initiative - Extensive geographical and legislative details of locations	- Limited documentation on data model specification - Facilitates peer reviewed wiki data, no standardisation	Event01 Event:involvesPhysicalInfrastructure https://www.wikidata.org/wiki/Q34370

Comparing these options, we observe the approach from Schema.org allows more control over which data properties are specified, being able to omit irrelevant ones. In contrast, the EU KG, Geonames and Kadaster offer an easier way of describing a location associated with an event, as all locations are referenceable using only an IRI. Moreover, the complexity of data stored for each location by these three initiatives may lead to more powerful querying capabilities but at the same time to a harder to use interface when looking for a particular location. We therefore recommend the user to use reference URIs from the EU KG, Kadaster KG, or Geonames, in that order, if available. Otherwise, the user may manually represent the location in question.

It is important to note that these initiatives that model infrastructural and geographical information in ontologies may not be sufficient to fully support the use cases in the MAGPIE domain. It is likely that tools and use cases need additional information relating to specific assets in and around the port that is not contained in these semantic models, such as infrastructural components above or under the ground in a specific port area. This information is currently contained in silos by different ports, but may need to be incorporated into ontologies and linked to the core model.

Codelists and classifications

Various resources are unavailable in a semantic format, but only exist in pdf or in another structured format. These include codelists for dangerous goods, container types, and many more. We have provided a preliminary RDF serialisation of the codelists used in the eFTI data requirements[20]. All these codelists are drawn from the UNECE codelists.

The best solution would be that existing standards provide linked data URLs for the identification of their codelist elements. That enables a direct reference to the codelist element from any other RDF data element. Additionally, it allows anyone with internet access to look up the intended meaning of the concept. However, this is for now an unreachable goal. Semantic modellers can also take this task upon themselves by providing a SKOS[21] implementation of the codelist when necessary.

[20] DG MOVE - eFTI Data Requirements (svn.gefeg.com)
[21] SKOS Simple Knowledge Organization System Reference (w3.org)

6. Conclusions and Next Steps

In this deliverable we have:

1. Explained our vision on using ontologies for implementing a language specification for the Port DT.
2. Described an ontology development methodology for reusing and extending ontologies on a use case basis.
3. Applied the methodology in new use cases.
4. Described a MAGPIE core ontology that functions as an overarching ontology that can be extended according to the proposed ontology development methodology.

We conclude that RDF ontologies, part of the semantic web and linked data paradigm, provide a suitable framework to implement the language specification for the Port DT. The reason is that they support the FAIR principles of data sharing, as well as future extensibility in new use cases, reuse of existing models, global identification, and federated data storage.

The methodology for developing ontologies described in this document leverages these features as it incorporates a use-case perspective and is centered around modularisation. The objectives of MAGPIE extend across multiple broad and interrelated domains, such as transport, logistics, energy, and sustainability. As it is unrealistic to adequately describe these domains ad hoc in a single semantic model that covers all possible use cases at any moment in time, the methodology facilitates incremental development on a use case basis. Further, it stimulates flexible reuse and extension of already existing models by incorporating a modular approach. We expect that this methodology facilitates easy adoption and integration of the semantic technology approach described in this document by stakeholders in other ports and use cases.

The MAGPIE core ontology that is implemented functions as an overarching ontology that can be used to integrate information from various interrelated domains relevant to MAGPIE. The ontology is modular and mainly consists of alignments with specialised modules in the different domains, such as SAREF4ENER to describe energy usage, ERA Vocabulary to describe railways and trains, SAREF4AUTO to describe trucks, and various geography ontologies for geographical and infrastructural objects. The core ontology is intended to be extended with additional (existing) modules such that it meets the requirements of new tools. Based on a shore-side electricity use case by PoR and a loading and charging use case by INESC TEC, we conclude that the core model can be reused and extended to support new MAGPIE applications by following the proposed development methodology.

We could, unfortunately, not manage to fully reflect the data requirements of the tools in T4.4 and T4.5 as well as the various MAGPIE demos in the semantic model. The reason is that the development of these tools and demos have only started near the end of T4.3 and their data requirements that have to be expressed in the semantic model are therefore not specified at the time of writing this deliverable. The development methodology we have outlined and validated in this deliverable can, however, serve as a practical guideline to facilitate extensions of the core model to support new use cases once their data requirements are known.

Future work in MAGPIE should therefore focus on extending the core model using the proposed methodology to cover the tools in T4.4 and T4.5 and MAGPIE demos once they are in a further stage of development. Validation of the methodology in these MAGPIE tools also validates that the methodology sufficiently guides the design of the ontology for future tools not yet envisioned in the project proposal, which is a requirement of the Port DT. The developers of these tools and demos, however, still need to invest in obtaining (external) expertise on RDF and semantic engineering. The current document guides a proper implementation of semantic models, but cannot completely replace the role of a semantic expert.

To integrate the work described in this document with the data space architecture described in Deliverable D4.2, future work should additionally focus on the integration of an implanted data space with data serialized in semantic web languages. For example, the data spaces architecture in D4.2 identifies a vocabulary hub component where the ontology is hosted and governed, for which the Semantic Treehouse tool can be employed. This tool should support the translation of competency question to query. Additionally, data space connectors facilitate the translation from a proprietary language to the shared ontology, but their useability should be investigated. We expect the various tools to gather experiences that can be documented in their deliverables.

7. References

[1] MAGPIE Deliverable 4.1: Digital Platforms & Services for Port Operations (2022).

[2] MAGPIE Deliverable 4.2: Definition of Modular Architecture for Port Digital Twin (2023).

[3] McKenna, E. Rademaker (2020). Digital Twins for the Port of the Future.

[4] the FAIR data principles (force11.org)

[5] Resource Description Framework (RDF) (w3.org)

[6] Linked Data - Design Issues (w3.org)

[7] Developer Portal (federatedplatforms.eu)

[8] Ontology-Development-101 (protege.stanford.edu)

[9] SAMOD (essepuntato.it)

[10] SABiO (nemo.inf.ufes.br)

[11] Linked Open Terms (lot.linkeddata.es)

[12] MAGPIE Deliverable 4.1, section 5.2.3: Smart and Green Logistics Tool.

[13] Semantic Web for the Working Ontologist (sciencedirect.com)

[14] An Introduction to Ontology Engineering (open.umn.edu)

[15] The Knowledge Graph Cookbook - Recipes That Work (poolparty.biz)

[16] The Web of Data (Book) (aidanhogan.com)

[17] LOT - Linked Open Terms (linkeddata.es)

[18] Extending Ontology Engineering Practices to Facilitate Application Development (link.springer.com)

[19] SPARQL Query Language for RDF (w3.org)

[20] DG MOVE - eFTI Data Requirements (svn.gefeg.com)

[21] SKOS Simple Knowledge Organization System Reference (w3.org)

[22] Digital Transport and Logistics Forum (DTLF) (europa.eu)

[23] SAREF: the Smart Applications REFerence ontology (etsi.org)

Appendix 1: Related Initiatives

Reusing the expressivity of already existing ontologies provides a major feature of our modularisation approach to developing ontologies. In the current section, we describe several initiatives where ontologies are developed that are in some way relevant to transport and logistics within and around the port area. The first ontology is the FEDeRATED ontology which we use as inspiration for developing a core ontology that can align various domain-specific models. We then describe several more domain-specific ontologies that may be considered for alignment with the core model.

FEDeRATED/DTLF

The FEDeRATED ontology is developed by the Digital Transport and Logistics Forum (DTLF)[22], an expert group raised and chaired by the DG Move that aims to support the European Commission in promoting digital transformation of the transport and logistics sector. The main goals of this transformation have been security, openness, and neutrality in data sharing.

The FEDeRATED ontology contains several modules covering conceptually separate sub-domains of information, namely a Digital Twin, Physical Infrastructure, Business Service, Classifications and Event module. The Digital Twin module describes various objects that are moving around during transportation activities, such as transport vehicles and cargo objects. The Physical Infrastructure module provides details on infrastructures that are related to logistical activities, such as train stations and physical locations. The Business Service module contains information elements on the type of service that is provided and the roles that actors may have in these services. The Classifications module allows for describing information according to specific standards or classifications. Finally, the Events module describes the events that occur, such as loading and unloading events, arrival and departure events and more. The Event module contains details about events, such as the type of an event, the datetime of an event as well as the type of datetime (e.g., whether an event has occurred at the datetime or whether an event is expected to happen at that datetime). Additional details on any event are to be described by using the information entities provided in the other modules, such as the arrival of a specific transport vehicle at a specific physical infrastructure.

[22] Digital Transport and Logistics Forum (DTLF) (europa.eu)

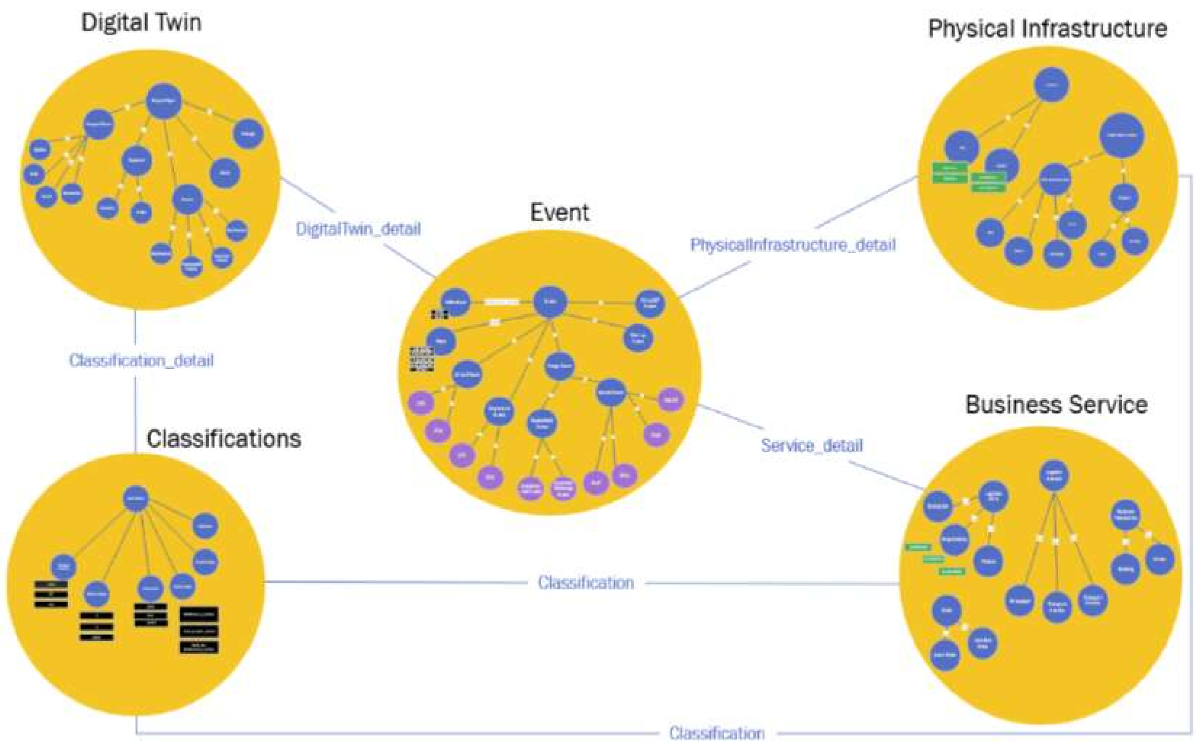


Figure 9 Relevant components of the FEDeRATED ontology

SAREF and its extensions

The Smart Applications REFerence (SAREF) ontology[23] has been developed in accordance with standardisation frameworks from ETSI. Different Specialist Task Forces have maintained and helped in the evolution of the ontology, including the interests of relevant industry stakeholders. The aim of SAREF is to enable interoperability between Internet of Things (IoT) applications in different sectors and produced by different manufacturers.

The ontology contains a core ontology (SAREF) that provides basic recurring concepts that are reused in several more specialized modules. We briefly describe the SAREF core ontology and two specialized modules that we consider especially relevant for the current MAGPIE use cases and tools, namely a module focused on the energy domain (SAREF4ENER) and a module specialized for the automotive domain (SAREF4AUTO).

The basic concept in the SAREF core ontology is the entity "Device", which is any physical object, such as a sensor, heat pump or light switch, that executes "Functions" to achieve "Tasks". For example, a washing machine is a device that executes a start/stop function to achieve the task of washing. Other basic concepts are "Measurement", "Property" and "UnitofMeasure". Namely, devices may perform measurements of properties according to a specific unit of measurement. Such concepts can be reused in all kinds of domains where (interactions between) smart devices and appliances are to be described.

The SAREF4AUTO extension models an extensive set of concepts, properties and relations that can be used to describe information related to the automotive domain. Objects that are

[23] SAREF: the Smart Applications REFerence ontology (etsi.org)

described include vehicles such as cars, trucks, public transport vehicles and two wheeler vehicles, but also other automotive objects such as traffic management centres and road equipment. Another class of relevant concepts provided by the module are specifications and dynamic features of vehicles, such as the position, weight, shape, speed and brake capacity.

The SAREF4ENER extension ontology models concepts of smart appliances and their energy-related capabilities to enable more efficient and sustainable use of energy resources. An example concept is the "PowerSequence", the specification of a task where energy is consumed and about which various information can be described, such as its current state (e.g., running, scheduled, inactive), its duration time or its estimated energy consumption.

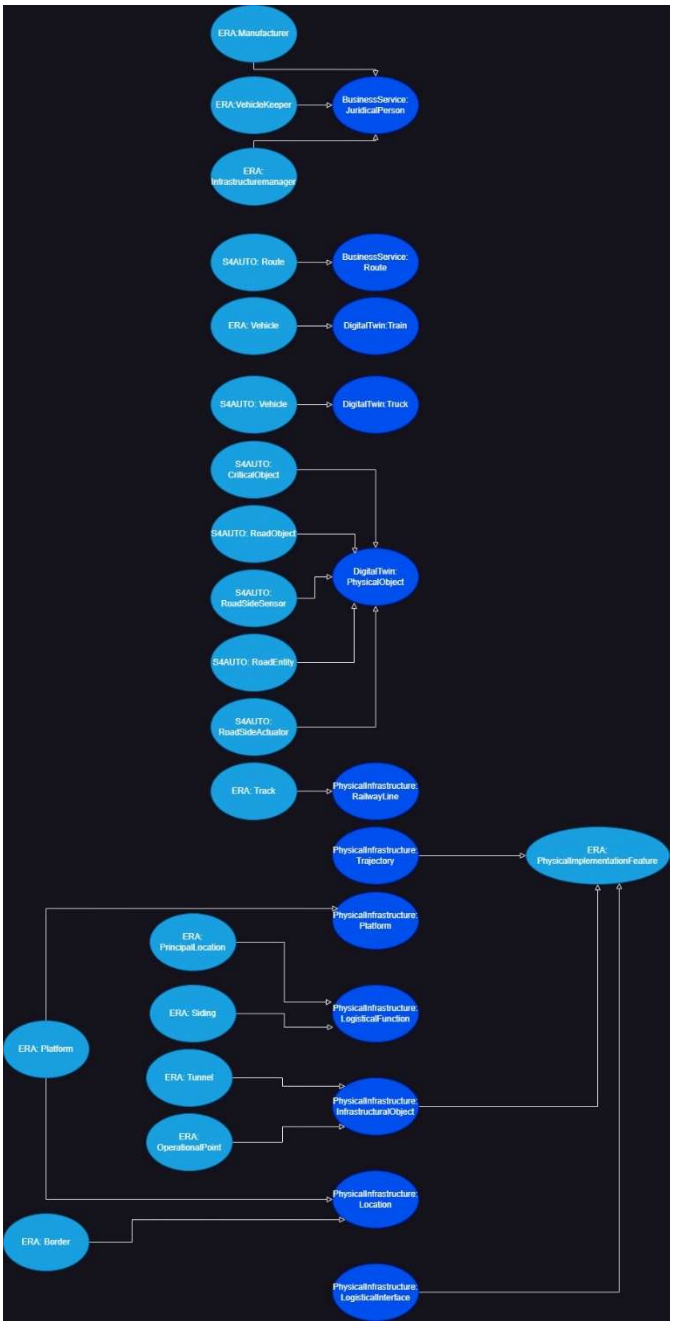


Figure 10. Overview of alignment of both SAREF4AUTO and ERA modules (light blue) to the core ontology (dark blue).

IATA OneRecord

The IATA OneRecord ontology models transport and logistics concepts relevant to administration of transactions and shipments in the air cargo industry. Its purpose is to provide a standard for data sharing and enable the establishment of a uniform record view of shipments. The ontology therefore does not provide much detail on vehicle specifications, but rather focusses on the administrative information relevant to shipments between parties.

ERA Vocabulary KG

The ERA Vocabulary Knowledge Graph is governed by the European Union Agency for Railways. It models the European railways infrastructure as well as the vehicles that operate over it. Examples include information on railway tracks, sidings and operational points where train services are provided.

EU KG and Kadaster

The Dutch Kadaster Knowledge Graph is developed by the Netherlands' Kadaster Land Registry and Mapping Agency (in short "Kadaster") and integrates various public data sources on administrative and spatial information in the Netherlands. It includes information on Dutch topography, a register of all Dutch addresses and buildings, and the judicial rights and restrictions associated with ownership of specific spatial assets.

FOAF

The "Friend Of A Friend" ontology has been developed starting 2000 and it models a description of a person and their relationships with other people. The FOAF ontology makes use of the OWL and RDF specifications to offer a collection of basic terms that can be facilitated through semantic technologies.

Appendix 2: Use Case INESC TEC

This Appendix provides an example where the ontology development methodology is applied by INESC TEC to extend the MAGPIE core ontology to support their use case. The use case describes the interaction between "eBarges" and "Trains", as well as the interaction between "eBarges" and "eTrucks". Specifically, the Containers' loading and unloading processes in the Port are shown. While the imported containers from the "eBarge" are loaded onto "eTrucks" and "Trains", the exporting containers from "eTrucks" and "Trains" are loaded into the "eBarge". In addition, the start and end of the charging events of the various objects involved in the processes are also presented.

Natural language description

It all starts when the "Ship-to-shore gantries" unload the containers to be imported from a "Ship" onto the "Automated terminal tractors" that take the containers to the "Terminal" where they are unloaded by "Transshipment vehicles". There, those containers will wait.

Later, when an "eBarge" arrives at the Port, it must ask the port authority for permission to enter before it can dock. Given specific requirements, the port authority will give a "Berth" to the "eBarge" permitting it to dock.

Once docked, the "eBarge" proceeds to charge its battery. Then, the "Ship-to-shore gantries" unload the containers to be exported to the "Automated terminal vehicles" that take them to the terminal for the "Transshipment vehicles" to place them in stacks.

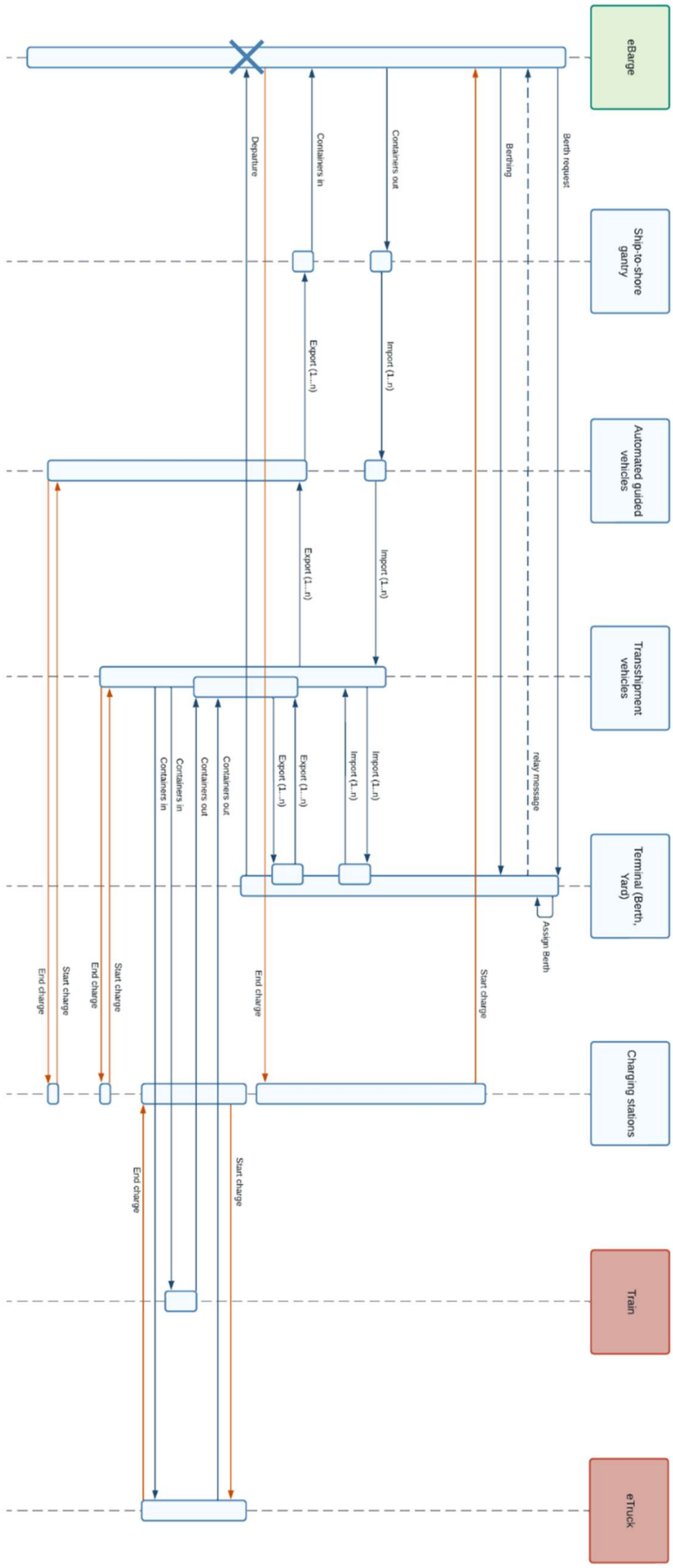
Eventually, these containers will be removed from the stacks by the "Transshipment vehicles" and moved to the "Automated terminal Vehicles", where they will be unloaded by "ship-to-shore gantries" and transported to a ship for export. The "Ship" then finishes loading its battery and proceeds with its cargo to ports in other countries.

When the "eBarge" finally empties, it is ready to receive new containers imported from "Inland Ports". Furthermore, the imported containers from the "Ship" leave the stacks with the help of the "Transshipment vehicles" and are placed onto the "Automated terminal vehicles" to be loaded into the "eBarge" by "Ship-to-shore gantries". Thus, the "eBarge" charges its battery and moves to its intended destination.

Again, with all tasks done, both the "Transshipment vehicles", and "Automated terminal vehicles" proceed to the "Charging stations".

Sequence Diagram

The next page provides a sequence diagram describing the sequence of information that is shared between the different objects.


eBarge
Ship-to-shore gantry
Automated guided vehicles
Transshipment vehicles
Terminal (Berth, Yard)
Charging stations
Train
eTruck
Berth request
Berthing
relay message
Assign Berth
Start charge
Containers out
Containers in
Departure
End charge
Import (1...n)
Export (1...n)
Containers out
Containers in
Containers in
Start charge
End charge
Start charge
End charge
End charge
Start charge

Competency questions

What does a terminal look like? What objects are in a terminal?

- What is the **Quantity** of available **Berths** in **Port** x?
- Which **Barge** is currently assigned to the **Berth** x?
- What **Transshipment Vehicle** operate at **Terminal** x?
- What is the **Maximum Capacity** of the **Yard** x?
- What is the **Latitude** and **Longitude** of **Terminal** x?
- Which **Yard** offers **Container Handling** or **Bulk Cargo Handling**?
- What is the **Maximum Cargo** capacity of the **Barge** x?
- Which **Transshipment Vehicle** has the **Highest Speed**?
- How many **Transshipment Vehicles** of a certain type are currently available?
- What is the **Maximum Capacity** of a specific **Transshipment Vehicle**?
- Does a certain type of **Ship** need a **Tug** to dock in the **Berth** x?
- Does the **Berth** have capacity to berth **Ship** of **Length** x?
- Is the **Weight** of **Container** x supported by the **Crane** x?
- How many **Cranes** can support more than x tons of **Weight**?
- How many **Berths** have the minimum **Water Height** of x meters?

When does the ship arrive? When does it go to maintenance?

- How many **Trucks** are available at **Terminal x** at **Time y**?
- What is the **Average Speed** of a **Barge**?
- What is the **Estimated Time of Arrival** of **Barge x** at the **Quay y**?
- How many **Hours** does it take for a **Transshipment Vehicle** to charge?
- What is the **Average Waiting Time** for a **Barge** to be assigned to **Berth**?

Energy usage: What is the current charging? how many power? What type of connection?

Object: Charging Station

- What is the **Energy Consumption** of a **Transshipment Vehicle** during operations?
- Which **Transshipment vehicle** has the **Highest Energy Efficiency**?
- What is the expected **Energy Consumption** of the **Trucks** on **Day x**?
- What is the **Estimated Time of Arrival** of the Truck to the **Charging Station**?
- What is the **Battery Level** after charging the **Truck**?
- How many **Charging Stations** will be available on **Day** y and **Hour z**?
- What is the current **Power** and **Currency** availability of the **Charging Stations** on **Day** x and **Hour** y?

Data models

1. Physical Object

Class	SubClassOf	Label	Description
PhysicalObject	Thing	Physical Object	Represents a high-level, abstract concept that includes any tangible, material entity that occupies space and can be perceived by our senses. It serves as a fundamental class from which more specific classes can be derived or subclassed, representing more types of physical objects.

Data Property	Data Type	Description
hasWeight	Float	The height of the object.
hasHeight	Float	The length of the object.
hasLength	Float	The width of the object.
hasWidth	Float	The weight of the object.
hasManufacturer	String	The manufacturer of the object.
hasModel	String	The model of the object.
hasSerialNumber	String	The serial number of the object.
hasProductionDate	Timestamp	The date of production of the object.
hasOwner	String	The date of production of the object.

1.2 Equipment

Class	SubClassOf	Label	Description

Equipment	PhysicalObject	Equipment	Various types of equipment used in port operations.

Data Property	Data Type	Description
hasElectricalEfficiency	Sting	Electrical efficiency of a specific piece of equipment. Representing how effectively it utilizes electrical energy to perform its intended function or operation.
HasAutonomyLevel	String	Level of autonomy ranges from remotely controlled to fully autonomous.
HasPowerSource	String	Powered by fossil fuel, electric or hybrid sources.
HasEnvironmentalCertification	String	Certification of complying with standards in environmental management and sustainability.
HasFuelCapacity	String	Capacity of the fuel tank, applicable only for equipment that uses fuel.
HasBatteryLife	Float	How long the equipment can operate before requiring a recharge.

1.2.1 Cargo Handling Equipment

Class	SubClassOf	Label	Description
CargoHandlingEquipment	Equipment	CargoHandling Equipment	Equipment specifically designed for the handling and movement of cargo within a terminal.

Data Property	Data Type	Description
IsTransshipmentVehicle	Boolean	Specialized cargo handling vehicles used for transferring cargo between different modes of transportation within a port or terminal.
HasMaximumLoadCapacity	Float	The maximum weight or load that the equipment can handle.
HasMaximumSpeed	Float	The maximum speed at which the equipment can travel and operate.

HasMaximumContainerSize	Float	The size of containers that the equipment can handle.
HasContanierWightCapacity	Float	The maximum weight of containers that the equipment can handle.
HasMaximumStackingHeight	Float	The maximum number of containers the can vertically stack.

1.2.1.1 Automated Guided Vehicle

Class	SubClassOf	Label	Description
AutomatedGuidedVehicle	CargoHandlingEquipment	Automated Guided Vehicle	Programmable vehicles that transport containers or other cargo within the port without human intervention. Also known as AGV.
LiftAGV	CargoHandlingEquipment	Lift AGV	This AGV is equipped with lifting mechanism to handle cargo that requires vertical movement.
ContainerAGV	CargoHandlingEquipment	Container AGV	This AGV is designed to handle containers.
PalletAGV	CargoHandlingEquipment	Pallet AGV	This AGV can pick up and transports palletized cargo.
TuggerAGV	CargoHandlingEquipment	Tugger AGV	This AGV is designed to be versatile as it can tow different carts or trailers.

Data Property	Data Type	Description
HasTrajectoryFollowing	Boolean	AGV's ability to navigate along a predefined programmed path or trajectory within a terminal.
HasObstacleDetection	Boolean	AGV's ability to detect and identify obstacles and

		obstructions in its path so it can better adjust its trajectory.

References

https://www.terbergspecialvehicles.com/en/development/automation/

https://www.konecranes.com/cargo-handling-journey

https://www.gaussin.com/apm-autonomous

https://easymile.com/news/autonomous-terminal-tractor-successfully-tested-mixed-traffic-terminal-operations-vlissingen

https://www.terbergspecialvehicles.com/en/

https://www.konecranes.com/port-solutions-equipment

1.2.1.2 Conveyor

Class	SubClassOf	Label	Description
Conveyor	CargoHandling-Equipment	Conveyor Systems	A conveyor is a mechanical system used to move or transport cargo horizontally, vertically, or at an inclined angle.
ShipLoader Conveyor	Conveyor	Ship Loader Conveyor	This conveyor is used to transfer various types of bulk materials or containers onto vessels.
StackerConveyor	Conveyor	Stacker Conveyor	This conveyor is used to create stockpiles of different types of cargo, such as bulk materials, containers, or palletized goods.
BeltConveyor	Conveyor	Belt Conveyor	This conveyor is suitable for a wide range of cargo, including bulk materials, packaged goods, or even individual items.
ShipUnloaderConveyor	Conveyor	Ship Unloader Conveyor	This conveyor is used to unload bulk materials or containers from the vessel cargo holds.
Continuous ShipUnloaderConveyor	Conveyor	Continuous Ship Unloader Conveyor	This conveyor is designed for efficient and continuous unloading of bulk materials from large vessels.

Data Property	Data Type	Description
IsPortable	Boolean	Indicates whether the conveyor is designed to be movable or portable.
HasLoading Speed	Float	The speed at which the conveyor can load cargo onto a vessel.
HasUnloadingSpeed	Float	The speed at which the conveyor can unloaded cargo from a vessel.
HasMaximumLoadCapacity	Float	The maximum load capacity of the belt conveyor.
HasClearanceHeight	Float	The vertical clearance height required for the conveyor to operate efficiently without obstruction.
HasTotalLength	Float	The total length of the conveyor.
HasCargoCompatibility	String	The types of cargo or materials the conveyor can handle.
HasMaximumAngle	Float	The maximum angle the conveyor can be inclined or declined.

References

https://www.beumergroup.com/pd/port-technology/

https://superior.widen.net/s/rldgh7qbsz/telestacker-conveyor-ship-loading-solutions-literature-low-resolution

https://www.skecon.com/product/ship-loader/

https://www.skecon.com/product/stacker-conveyor/

1.2.1.3 Crane

Class	SubClassOf	Label	Description
Crane	Cargo-Handling-Equipment	Crane	Crane is a type of handling equipment designed for lifting and moving heavy objects or materials.
Ship-to-shoreGantryCrane	Crane	Ship-to-shore Gantry Crane	A type of crane used to load and unload cargo containers from ships at seaports. Also known as Quay Crane.
StackingCrane	Crane	Stacking Crane	A type of crane used in storage yards to stack containers vertically, one on top of the other, and to move them from one location to another.
AutomatedStackingCrane	Crane	Automated Stacking Crane	A type of crane fully automated that moves containers in

			and out of stacks at storage yards.
MobileHarborCrane	Crane	Mobile Harbor	A type of crane that can move around the port area, used for loading/unloading various types of cargo.
RailMountedGantry	Crane	Rail-Mounted Gantry	A type of crane that moves on rails, used for handling containers in storage areas or stacking yards.
RubberTiredGantry	Crane	Rubber-Tired Gantry	A type of crane similar to rail-mounted gantry cranes but with tires, allowing for more flexibility in movement.
FloatingCrane	Crane	Floating Crane	A type of crane mounted on barges or other floating platforms, used for shipbuilding, salvage, or heavy lifting.
JibCrane	Crane	Jib Crane	A type of crane with a horizontal arm (jib) that can rotate, used for handling general cargo.
TelescopicCrane	Crane	Telescopic Crane	A type of crane with a telescoping boom, used for various lifting tasks that require a compact and maneuverable crane.
TowerCrane	Crane	Tower Crane	A type of tall and fixed crane used for construction activities or heavy lifting tasks in a limited area.

Data Property	Data Type	Description
HasMaximumLiftingCapacity	Float	The maximum weight that the crane can safely lift. Measure in Kilogram (kg).
HasMaximumLiftingHeight	Float	The maximum height to which the crane can lift objects. Measure in Kilogram (kg).

HasMaximumWorkingRadius	Float	The maximum distance from the crane's centre at which it can operate. Measure in Meters (m).
HasBoomLength	Float	The length of the crane's boom or arm. Measure in Meters (m).
HasCounterweight	Float	The weight used to stabilize the crane during operation. Measure in Kilogram (kg).
HasBaseDimensions	Float	The dimensions of the crane's base or footprint. Measure in Meters (m).

1.2.1.4 Forklift

Class	SubClassOf	Label	Description
Forklift	CargoHandling-Equipment	Forklift	Forklifts are versatile cargo handling vehicles used for lifting, moving, and stacking various loads. Also known as lift trucks.
ReachStackers	Forklift	Reach Stackers	This forklift has a telescopic boom that can extend both forwards and upwards, allowing them to reach containers stacked in multiple rows. Also known as container forklift.
LadenContainerHandler	Forklift	Laden Container Handler	This forklift has a hydraulic system and specialized attachments designed for handling laden (loaded) containers. Also known as Loader Container Handler.
CounterbalanceForklift	Forklift	Counterbalance Forklift	This forklift has a counterbalance weight at its rear so it can handle a wide range of materials and goods.
Telescopic Forklift	Forklift	Telescopic Forklift	This forklift has a set of attachments at the end of the boom so it can handle a variety

			of loads. Can lift both vertically and horizontally. Also known as telehandler.
HighCapacityForklift	Forklift	High Capacity Forklift	This forklift is designed to lift and transport heavy cargoes, such as large machinery, containers, and other heavy industrial materials.

Data Property	Data Type	Description
Maximum Lifting Capacity	Float	Maximum weight it can safely handle and lift.
Maximum Stacking Height	Float	Maximum height to which it can vertically stack.
Has Operator Cabin	Boolean	Indicates whether it is equipped with an operator cabin.
Cabin Position	String	The position the operator cabin can have.
Attachment Option	String	Attachment options that can be use. Enables it to handle different types of cargo and carry one or more containers.

References

https://www.kclifttrucks.com/equipment/konecranes-lift-trucks

1.2.2 Charging Station

Class	SubClassOf	Label	Description
ChargingStation	Equipment	Charging Station	Charging stations are locations where battery-powered devices can be charged or recharged.
OnShorePower-ChargingStation	ChargingStation	Onshore Power Charging-Station	A type of charging station used to provide electrical power to ships at port, allowing them to turn off their diesel engines and reduce emissions.

OffShoreChargingBuoy	ChargingStation	Offshore Charging Buoy	A type of charging station located in a designated offshore area near a port. It's used for used for charging electric vessels or other marine equipment.
ElectricVehicle-ChargingStation	ChargingStation	Electric Vehicle Charging Station	A type of charging station used for electric vehicles, such as cars, trucks, or cargo-handling equipment.
HybridPower-ChargingStation	ChargingStation	Hybrid Power Charging Station	A type of charging station designed to provide power to hybrid vessels with diesel engines and electric motors.
BatteryChargingStation	ChargingStation	Battery Charging Station	A type of charging station designed to charge batteries for electric or hybrid vessels.
FuelCell-ChargingStation	ChargingStation	Fuel Cell Charging Station	A type of charging station used to charge fuel cells, an energy storage device that can generate electricity from hydrogen or other fuels.
AGVChargingStation	ChargingStation	AGV Charging Station	A type of charging station designed to ensure the continuous operation of automated guided vehicles.

Data Property	Data Type	Description
hasPlugType	String	The type of plug used by the charging station.
hasConnectorType	String	The type of connector used by the charging station.
hasChargingCapacity	Float	The maximum charging capacity of the station.
hasMaxPowerOutput	Float	The maximum power output of the charging station. Measure in Kilowatt (kW).

hasNumberOfConnectors	Integer	The number of connectors available on the charging station.
hasMaxVoltage	Float	The maximum voltage the charging station can provide during a charging session. Measure in Volts (V).
hasMinVoltage	Float	The minimum voltage the charging station can provide during a charging session. Measure in Volts (V).
hasMaxCurrent	Float	The maximum current the charging station can provide during a charging session. Measure in Amperes (A).
hasMinCurrent	Float	The minimum current the charging station can provide during a charging session. Measure in Amperes (A).

1.3 Transport Means

Class	SubClassOf	Label	Description
TransportMeans	PhysicalObject	Transport Means	These are the vehicles that transport the cargo, such as tricks, vessels, trains, airplanes, barges etc.

Data Property	Data Type	Description
hasMode	String	Specific type: road, railroad, sea, inland waterways.
hasCapacity	Float	Total weight of the load that can be safely moved by a vehicle (payload).
hasMaximumSpeed	Float	Maximum attainable speed of the vehicle in [km/h] (if it is in knots should be converted).
hasEnergySource	String	Energy source of the vehicle (independently of the propulsion system).
hasEnergyConsumption	Float	Nominal consumption rate of the vehicle in [l/km], [kWh/km], [kg/km] (also can be given per unit of time like kg/h).
hasFuelCapacity	Float	The capacity of the fuel tank, the fuell cell or in the case of electric cars, the battery.
hasFuelTankStatus	Float	The current level of fuel/charge of the vehicle.
hasEmissionFactor	Float	Amount of emission released [g/km] or [g/h].
hasLoadFactor	Float	Indicate the load carried by the vehicle relative to the maximum weight allowed.
hasAvailabilityStatus	String	Indicate if a vehicle is available to perform a task or not. The availability

		status could be due to a Load Factor different of 0, or because several other factors: failure, maintenance, ubication, etc.

1.3.4 Vessel

Class	SubClassOf	Label	Description
Vessel	TransportMode	Vessel	Watercraft, also known as water vessels or waterborne vessels, are vehicles used in water, including boats, ships, hovercraft and submarines. Watercraft usually have a propulsive capability whether by sail, oar, paddle or engine and hence are distinct from a simple device that merely floats, such as a log raft.

1.3.4.1 Barge

Class	SubClassOf	Label	Description
Barge	Vessel	Barge	A type of Vessel used for transporting cargo on flat-bottomed inland waterways.
DryBulkCargoBarges	Barge	Dry Bulk Cargo Barges	A type of barge used to haul and ferry dry cargo such as sand, food, grains, coal, and minerals.
BarracksBarge	Barge	Barracks Barge	A type of barge known as a houseboat and mainly used for residential purposes.
Car-floatBarges	Barge	Car-float Barges	A type of barge used to ferry rail carts.
HopperBarge	Barge	Split Hopper Barge	A type of barge used for marine construction and known for carrying dredged materials.
PowerBarge	Barge	Power barge	A type of barge that serves as a moveable power plant.
DeckBarges	Barge	Deck Barges	A type of barge that carries cargo on deck and is known to be a work platform for workers and machinery.
Spud&CraneBarges	Barge	Spud & Crane Barges	A type of barge that can support a working crane.
ShaleBarges	Barge	Shale Barges	A type of barge that has cargo bins and open

			hoppers that can carry nonhazardous oilfield waste.
TankBarges	Barge	Tank Barges	A type of barge designed with bin tanks capable of circulating and discharging fluids such as petroleum, petroleum products, ethanol, chemicals, and fertilizer.
LNGBunkerBarge	Barge	LNG Bunker Barge	A type of barge used to store, transport, and supply liquefied natural gas to ships.
ContainerBarge	Barge	Container Barge	A type of barge that carries containers in its hall.
PortFeederBarge	Barge	Port Feeder Barge	It is a type of barge equipped with a container crane that does not need a quay crane to unload its containers.
ArticulatedTugBarges	Barge	Articulated Tug Barges	A type of barge that is joined together with a tug.

Data Property	Data Type	Description
HasAutonomyLevel	String	A barge's Level of autonomy ranges from remotely controlled to fully autonomous.
HasEnvironmental Certification	String	Certification of complying with standards in environmental management and sustainability.
HasPowerSource	String	Powered by fossil fuel, electric or hybrid sources.
HasPropulsion	Boolean	Ranging from non-self-propelled (needs a tug) to self-propelled.
HasCargoHandlingEquipment	String	Types of equipment or mechanisms for loading and unloading cargo, like a crane.
HasFuelCapacity	String	Capacity of the fuel tank, applicable only for Barges that use fuel.
HasBatteryLife	Float	How long the Barge can operate before requiring a recharge

References

https://www.marinelink.com/news/installing-rotterdam396646
https://www.greenaward.org/inland-shipping/about-green-award/
https://www.konecranes.com/port-equipment-services/offering-for-shipyards

https://www.researchgate.net/publication/280763184_Port_Feeder_Barge_Advanced_Waterborne_Container_Logistics_for_Ports
https://fincantierimarinegroup.com/product_cat/lng-bunker-barge/
https://fincantierimarinegroup.com/product_cat/articulated-tug-barges/
https://www.dw.com/en/are-electric-vessels-the-wave-of-the-future-in-shipping/a-43046309

2.2 Port

Class	SubClass Of	Label	Description
Port	PhysicalObject	Port	A Port is a location where a vessel can dock to load and unload cargo or passengers.
Seaport	Port	Seaport	Located along coastlines, this Port facilitates the loading and unloading of various vessels, connecting international trade routes.
InlandPort	Port	Inland Port	Located on rivers or lakes, this Port links Seaports to Inland Ports, providing access to waterways for cargo transportation.
InlandContainerDepot	Port	Inland Container Depot Port	This Inland Port connects to Seaports via rail or road, offering container handling and customs clearance to streamline cargo movement.
RiverPort	Port	River Port	Located on rivers, this Port facilitates cargo and passenger transport via inland waterways.
ContainerPort	Port	Container Port	This Port is designed for containerized cargo as it offers equipment for more efficient handling of containers.

Data Property	Data Type	Description
Haslocation	Float	The coordinates or address of the Port.
HasIntermodalConnectivity	String	The different transportation modes available for connection in a Port, such as rail, road, or waterway links to other Ports.

2.2.1 Terminal

Class	SubClassOf	Label	Description
Terminal	Port	Terminal	A terminal is a section dedicated to the handling, storing, and moving of cargo between different modes of transportation within a port.

ContainerTerminal	Terminal	Container Terminal	Terminal for containerized cargo; import and export containers grounded.
LiquidBulkTerminal	Terminal	Liquid Bulk Terminal	Terminal for petroleum products, chemicals, biofuels, and refining.
DryBulkTerminal	Terminal	Dry Bulk Terminal	Terminals that handle minerals, chemicals and related products, metal products, waste, and scrap materials.
PassengerTerminal	Terminal	Passenger Terminal	Terminal for passengers boarding and leaving water vessels such as ferries, cruise ships, and ocean liners.
BreakbulkTerminal	Terminal	Breakbulk Terminal	Terminal for breakbulk commodities that cannot be shipped via container.
AutomobileTerminal	Terminal	Automobile Terminal	Terminal for vehicle processing and logistics services.
GeneralUseTerminal	Terminal	General Use Terminal	Terminal for management purposes, water taxi, emergency response, and others.
FerryTerminal	Terminal	Ferry Terminal	Terminal for unloading passengers, vehicles, and cargo onto ferry vessels.

Data Property	Data Type	Description
HasMaximumCapacity		The maximum capacity and capabilities of a Terminal in terms of cargo handling.

2.2.1.1 Quay

Class	SubClassOf	Label	Description
Quay	Terminal	Quay	A quay is a platform constructed parallel to the shoreline providing a docking space or platform for vessels to moor. It describes the infrastructure itself.

Data Property	Data Type	Description
HasName	String	Distinct alphanumeric code or a combination of letters and numbers.

HasMaxVesselLength	Float	The maximum length of vessels that can be accommodated.
HasMaxVesselDepth	Float	The Maximum depth of water required for vessels to navigate and dock safely.

2.2.1.1.1 Berth

Class	SubClassOf	Label	Description
Berth	Quay	Berth	A berth is an area along a waterfront where a vessel can dock, load, unload, or remain moored. It describes the specific assigned space in a quay.
ContainerBerth	Berth	Container Berth	This berth allows for more efficient loading and unloading of containers, serving different container terminals within the port.
DryBulkBerth	Berth	Dry Bulk Berths	This berth is designed for handling dry bulk commodities such as iron ore and grains and facilitating bulk cargo transfer.
LiquidBulkBerth	Berth	Liquid Bulk Berths	This berth is equipped to handle various types of liquid bulk cargo, such as petroleum products and chemicals.
PassengerShipBerth	Berth	Passenger Ship Berth	This berth provides service for passenger embarkation, disembarkation,

			and port facilities for passenger ships.
FerryBerth	Berth	Ferry Berth	This berth provides safe and efficient embarkation and disembarkation of passengers and vehicles with boarding bridges and transportation connections.
BreakBulkBerth	Berth	Break Bulk Berth	This berth is designed to handle the loading and unloading of break bulk cargo and offers specialized equipment for transferring items between vessels and the port.
DolphinBerth	Berth	Dolphin Berth	This berth is an offshore facility located in the sea designed to transship liquid and dry bulk cargo.
HybridBerth	Berth	Hybrid Berth	This berth is designed to support vessels that operate using a combination of conventional fuels and alternative energy sources such as liquefied natural gas (LNG), hydrogen, battery power, or other forms of renewable energy.

Data Property	Data Type	Description
HasNumberOfBerths	Float	The number of berths a Port has.
HasStorageCapacity	Float	The capacity of storage areas available at the berth for storing cargo.
HasAvalability	Float	Available spaces for vessels to dock and manoeuvre.
HasServices	String	Services provided such as fuelling, provisioning, waste disposal, and others.
HasHandlingEquipment	String	Types of equipment available for cargo handling operations.
HasLoadingUnloadingRate	Float	The rate at which cargo can be loaded or unloaded from vessels.
HasLocation	Float	Geographic location within the terminal or port area, typically using coordinates.

2.2.1.2 Yard

Class	SubClassOf	Label	Description
Yard	Terminal	Yard	A terminal has one or more associated yards that serve as an area where goods are stored, sorted, and organized for efficient handling and distribution.
ContainerYard	Yard	Container Yard	This yard is where shipping containers are temporarily stored in stacks or rows. Also known as Container Depot.
InspectionYard	Yard	Inspection Yard	This yard is where cargo undergoes inspection from customs officials.
X-rayYard	Yard	X-ray Yard	This yard is where containers are subjected to non-intrusive inspection using X-ray technology. Also known

			as Container Scanning Yard.
EmptyContainerYard	Yard	Empty Container Yard	This yard is dedicated to storing empty containers.
IntermodalYard	Yard	Intermodal Yard	This yard facilitates the transfer of containers between different modes of transportation.
Refrigerated-ContainerYard	Yard	Refrigerated Container Yard	This yard is designed for storing refrigerated containers that require electricity to maintain specific temperatures. Also known as Reefer Yard.
Roll-onRoll-offYard	Yard	Roll-on/Roll-off Yard	This yard is designed to accommodate vehicles that can be loaded and unloaded onto or off vessels. Also known as Ro-Ro Yard.
MaintenanceYard	Yard	Maintenance Yard	This yard is designated and dedicated to equipment maintenance, repair, updating, and servicing.

Data Property	Data Type	Description
HasMaximumCapacity	Integer	The maximum number of containers a yard can accommodate.
HasOccupancy	Float	The current occupancy level of a yard
HasEquipment	String	The type of transshipment equipment available in a yard.
HasLocation	Float	The geographical location of a yard within a port.
HasMaximumHeight	Float	Containers are stacked with a maximum height (ISO standard 1496-1:2019).
HasPlugCapacity	Integer	The number of electric power outlets available.

2.2.1.3 Warehouse

Class	SubClassOf	Label	Description
Warehouse	Terminal	Warehouse	A warehouse serves as a temporary storage facility for a variety of cargo.
General Warehouse	Warehouse	General Warehouse	Warehouse for a variety of goods in pallets, packaged.

RefrigeratedWarehouse	Warehouse	Refrigerated Warehouse	A warehouse equipped with temperature-controlled refrigeration systems for perishable goods such as food and pharmaceuticals. Also known as Cold Storage Warehouse.
HazardousMaterialsWarehouse	Warehouse	Hazardous Materials Warehouse	Warehouse that adheres to strict safety regulations and guidelines to ensure proper handling, storage, and containment of hazardous materials and chemicals. Also known as Hazardous Materials Storage.
BondedWarehouse	Warehouse	Bonded Warehouse	Warehouse that only authorized personnel can enter. For imported goods awaiting clearance from customs.

References

https://www.portodelisboa.pt/en/glossario#letterT
https://openjicareport.jica.go.jp/pdf/11681590_04.pdf
https://www.portofrotterdam.com/en/logistics/storage-and-transhipment/warehousing
https://www.portofrotterdam.com/en/logistics/storage-and-transhipment/container-depots

Data Property	Description
HasMaximumCapacity	The total storage capacity of a warehouse, measured in square meters or cubic meters.
HasSecurityMeasures	The specifies the security measures in place of a warehouse, such as surveillance systems, access control, and alarm system.
HasTemperatureRange	The temperature ranges a warehouse is maintained at.

HasSafetyProtocols	The safety protocols and procedures in place for handling, storing, and transporting goods in a warehouse.