Ref. Ares(2022)7550021 - 31/10/2022

AI-PROFICIENT

Artificial intelligence for improved production efficiency, quality and maintenance

Deliverable 5.2

D5.2: Semantic data model for integrated digital twins

WP 5: AI-PROFICIENT system integration and deployment

T5.2: Semantic knowledge graph for integrated digital twins

Version: 1.0

Dissemination Level: PU

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957391.

Table of Contents

List of Figures

List of Tables

Disclaimer

This document contains description of the AI-PROFICIENT project work and findings.

The authors of this document have taken any available measure in order for its content to be accurate, consistent and lawful. However, neither the project consortium as a whole nor the individual partners that implicitly or explicitly participated in the creation and publication of this document hold any responsibility for actions that might occur as a result of using its content.

This publication has been produced with the assistance of the European Union (EU). The content of this publication is the sole responsibility of the AI-PROFICIENT consortium and can in no way be taken to reflect the views of the European Union.

The European Union is established in accordance with the Treaty on European Union (Maastricht). There are currently 28 Member States of the Union. It is based on the European Communities and the Member States cooperation in the fields of Common Foreign and Security Policy and Justice and Home Affairs. The five main institutions of the European Union are the European Parliament, the Council of Ministers, the European Commission, the Court of Justice and the Court of Auditors (http://europa.eu/).

AI-PROFICIENT has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957391.

Title: D5.2 Semantic data model for integrated digital twins

Lead Beneficiary	TF
Due Date	31.10.2022.
Submission Date	30.10.2022.
Status	Final
Description	Semantic knowledge graph for smart data handling and data fusion to introduce relations to system components and digital twins
Authors	IMP, VTT, TEK, IBE, INOS, TF
Type	Report
Review Status	PC + TL accepted
Action Requested	For acknowledgment by partners

VERSION	ACTION	OWNER	DATE
0.1	Initial table of contents	TF	06.09.2022
0.2	Contribution	TF	22.09.2022
0.3	Contribution	TEK	18.10.2022
0.7	Editorial corrections	TF	24.10.2022
0.8	Contribution & corrections	TF	27.10.2022
0.9	Reviewer's comments	TF	28.10.2022
1.0	Final version submitted	UL	31.10.2022

Executive Summary

The Deliverable D5.2 is a public document of AI-PROFICIENT project delivered in the context of WP5, Task 5.2 (Semantic knowledge graph for integrated digital twins), with regard to the data semantification within the use cases which has been developed by the involved partners.

This document provides an insight into the process how the sensor measurements and the sensor equipment are ontologically expressed. It documents the challenges and the considerations that were raised during this process. The data model is implemented in data layer for the AI-PROFICIENT AI-services. This integration is outlined in this deliverable.

The document also presents the FIDES ontology which targets the improvement of the understanding and trustworthiness on AI-systems when they are supporting decision making. The deployed AI-services are for supporting the decision making in the production process. Instead of relying blindly on the deployed systems, it is good to be able to get an insight into how these AI-systems are combined to each other and how the decision-making process happens. That knowledge will increase the trust in the provided decision by the AI-service to the operator.

1 Introduction

1.1 Scope

The document describes activities around capturing the terminology being used within design Digital Twins for the manufacturing. The need for means to share information about the manufacturing processes in a way that is facilitating the communication between the human process experts and the data scientists in a coherency with the digital data pipelines that must be setup for AI systems has been widely understood.

In this area Semantic Technology is a well-established method and approach to aid in building the bridges between the human and the machine perspective, but also between generic and problem specific perspectives.

In this deliverable the focus is on one aspect of information with a Digital Twin: namely to provide context to the measurements that sensors provide.

1.2 Confidentiality

This deliverable is a public deliverable. Since the activities are about capturing the information of the actual manufacturing processes, the actual data specifications cannot be present in this deliverable because of risking to disclose more information of the manufacturing process than is allowed by the confidentiality agreement. Therefore, this deliverable will present the approach, its challenges, and opportunities encountered throughout the activity, rather than the actual data specification. In the deliverable, non-public accessible pointers are given to the location of the data specifications that fall under this consideration.

1.3 Audience

The intended audience for this document is the members of AI-PROFICIENT consortium, Project Officer as well as the general public, since this document is public. More specifically, the consortium members in charge of development, integration, AI service deployment and platform implementation are expected to use the content presented here.

1.4 Relations to other tasks and work packages

The initial specifications and the implementation requirements stipulated in Task 1.5 serve as the input for defining the data interfaces and impact the semantic vocabularies selection. Task 5.2 has coherent cooperation among the WP5 tasks. Particularly, Task 5.1 provides the detailed specification for the AI-PROFICIENT platform up and down data streams supported by the deployed service-oriented communication middleware at the factory premises. The integration and deployment of the overall AI-PROFICIENT platform within Task 5.4 require continuous integration with Task 5.2 throughout the whole WP5 duration. Furthermore, the information delivered by the Digital Twins in Task 5.2 will be converted to AI services and decision-making services developed by Tasks 3.1 and 3.5. The human-machine interfaces adapted to the use in the individual use cases in Tasks 4.2 and 4.3 will be fed by the semantically enriched data streams. Similarly, the decision-making through Task 4.4 Explainable artificial intelligence (XAI) services are based on the various database layers of Task 5.2. The ethics considerations in Task 6.2 are expected to impact and enrich the relationships and the logic between the use cases entities that are involved in Figure 1 summarizes the aforementioned dependency of Task 5.2 with the adjacent tasks and work packages of the AI-PROFICIENT project.

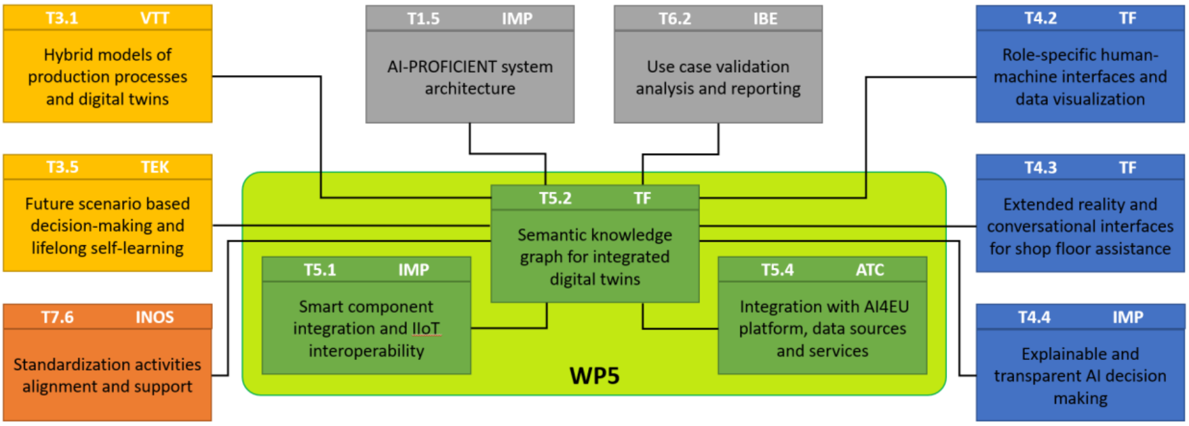

Figure 1: Relation of Task 5.2 with other tasks

1.5 Structure

The present document is divided into 2 main parts: the first part concerns the sensor data that is obtained from the manufacturing process and how to provide a consolidated view on that. This is covered by sections 2 and 3. The second concerns the ability to understand the interactions between AI systems. That activity has led to the design of the FIDES ontology. Section 4 describes this work.

2 Sensor data contextualisation

The starting point for an integration of a manufacturing process is access to the sensor data. This data is coming from devices into a data repository via device's protocols and communication channels.

When aggregating data from sensors throughout the manufacturing process in one collection, the need for additional context of this increases. At the same time, such aggregation has as side effect to harmonise the sensor data in a structure that is more easily to penetrate and can be used for all kinds of sensors. This level of abstraction is the scope of the semantic data specifications in this document.

It is assumed that the device engineers are capable to convert the data into a form compatible with the semantical data specification.

In addition, it is a target to add the necessary hooks and means to add context to the sensor data. In particular, it should be possible to easily retrieve background information about the sensor device that has produced the measurement. The nature of that information is different from a device measurement. Usually, it is stored in a different information source. The objective is to interconnect both.

The Figure 2 provides a high-level perspective of the information flows and the artefacts part of this activity. On the lefthand side, the plant with its manufacturing process is depicted. Central in the green area, the elements that are touched by this activity are shown, while at the bottom, in yellow, the AI Services that are consuming the sensor data are depicted. This implements the data layer as described in Deliverable D1.5, section 4.1.2.

The Figure 2 shows that the green layer is hiding the differences (and thus the complexity) between accessing the sensors. It thus forms a first layer into building a digital twin for the manufacturing process.

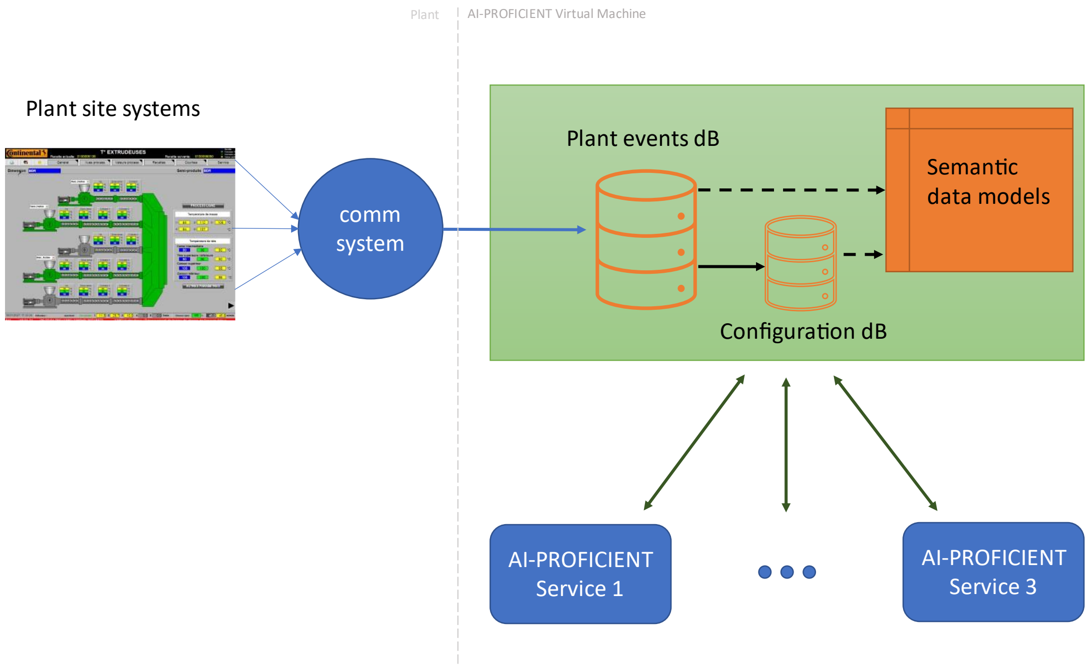

Figure 2: AI-PROFICIENT data layer for sensor measurements

2.1 The semantic data specification of sensor data

2.1.1 Reusing international standards

To get a grip on sensor data the international community worked out several standards within distinct standardisation bodies early 2000 (ISO, OGC, W3C). Each of these standards have their focal points, but where difficult to combine together. More recently, at W3C, a coherent integration which addresses these differences in audience and scope has been realized. The OGC/W3C SSN/SOSA recommendation provides a Semantic Web based approach for sensor data which is extensive but lightweight at the same time.

SSN/SOSA provides means to describe coherently the system properties of a sensor, but doing so it abstracts away the actual physical device. In AI-PROFICIENT the connection with the physical world is maintained by reusing SAREF. SAREF offers a technical perspective on a sensor network, and that is exploited here.

- SSN/SOSA, OGC/W3C Semantic Sensor Network Ontology. https://www.w3.org/TR/vocab-ssn
- OM, ISO 19156 Observations and Measurements https://www.iso.org/standard/32574.html
- SAREF, https://saref.etsi.org/core/v3.1.1/

2.1.2 Core model

The AI PROFICIENT Core Model for sensor data consists of 3 main interconnected notions:

- Observation
- Sensor
- Production Equipment

An Observation is the determination of the value of a particular characteristic of an entity at a given time or between two times. More commonly worded: it is the measurement made by the sensor. The Sensor is the device, agent (including humans), or software (simulation) involved in, or implementing, a procedure to make an Observation. Note that a sensor is not necessary solely a device, but also can be a human providing the measurement. The notion of production equipment is introduced to provide context to the sensors. It denotes entities that can produce items in a manufacturing process, describing the entity in terms of functions, states and services. Knowledge of where the sensor is located in the manufacturing process is key for interpreting the sensor observations correctly.

A UML diagram of the Core Model is shown below (see Figure 3). The actual core model is published at [https://data.ai-proficient.eu/doc/applicationprofile/core.](https://data.ai-proficient.eu/doc/applicationprofile/core) In the diagram the notion Observation shown on the left-hand side, the notion Sensor is shown in the middle and the notion Production Equipment on the righthand side.

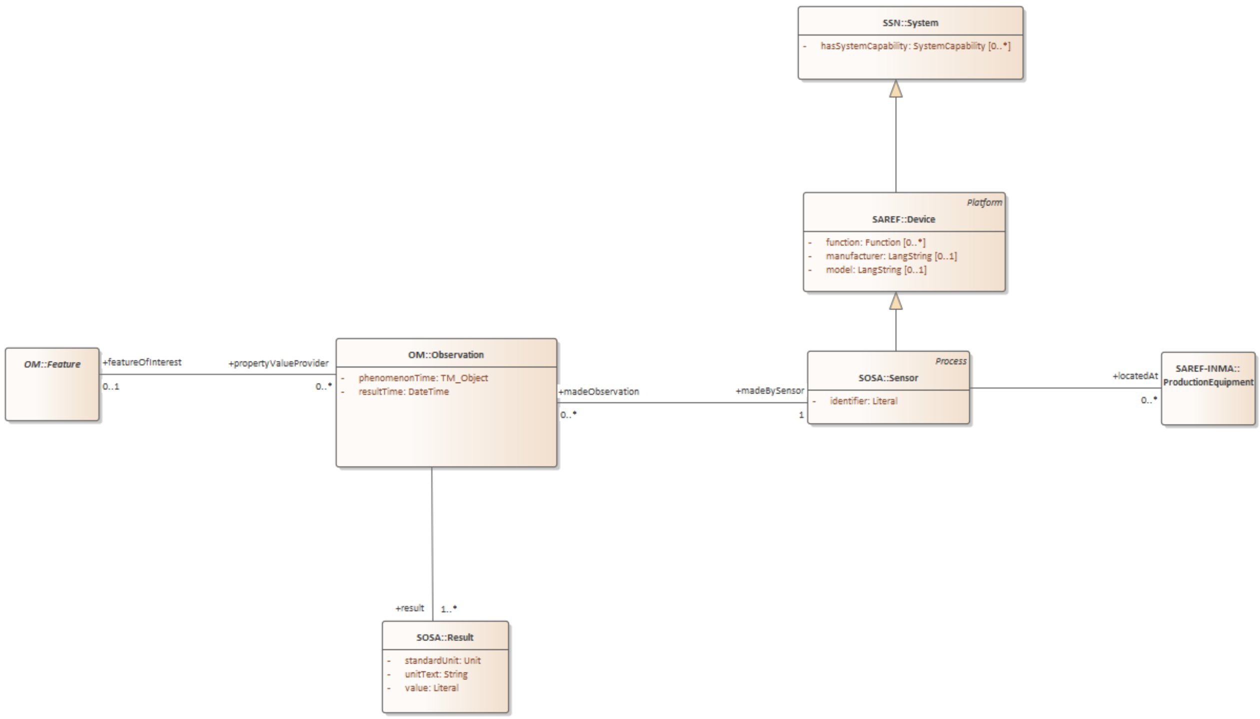

Figure 3 : UML diagram of Core model

2.1.3 Challenges and extensions

Where it was rather straightforward to determine the general umbrella under which this approach could fit, the usage of the SSN/SOSA model to capture the information provided was less straightforward.

The following knowledge engineering challenges had to be resolved:

1. Connecting sensors with the correct production equipment: how to express that a sensor SN1 is located at the Production Equipment of Type Z.
2. Sensors are in relationship with each other: some sensors are setting the target value, while other measure the real value.
3. The unit of the observation made by the sensor.

Challenge 1 – location

The key location for the sensor device is the production equipment to which it is attached. In the data modeling two patterns can be applied for encoding the sentence “the temperature sensor is located at the extruder”. In Figure 4 both patterns are illustrated. The first pattern (*pattern 1 show on top of the figure*) is to use classes to denote a group with the same characteristic: e.g., in the example sentence, it would correspond to the class Extruder. The consequence of this pattern is that each time a new production equipment is introduced a new class is added to the semantic data model. The alternative pattern (*pattern 2*) is to introduce a property *type* or *category to* indicate the kind. In this pattern the distinction between production equipment is only visible at instance level of the data. It is the associated category value that determines that a production equipment is an extruder. Using the second pattern the model does not change when a new production equipment category is added; it is the associated codelist with the property that is extended.

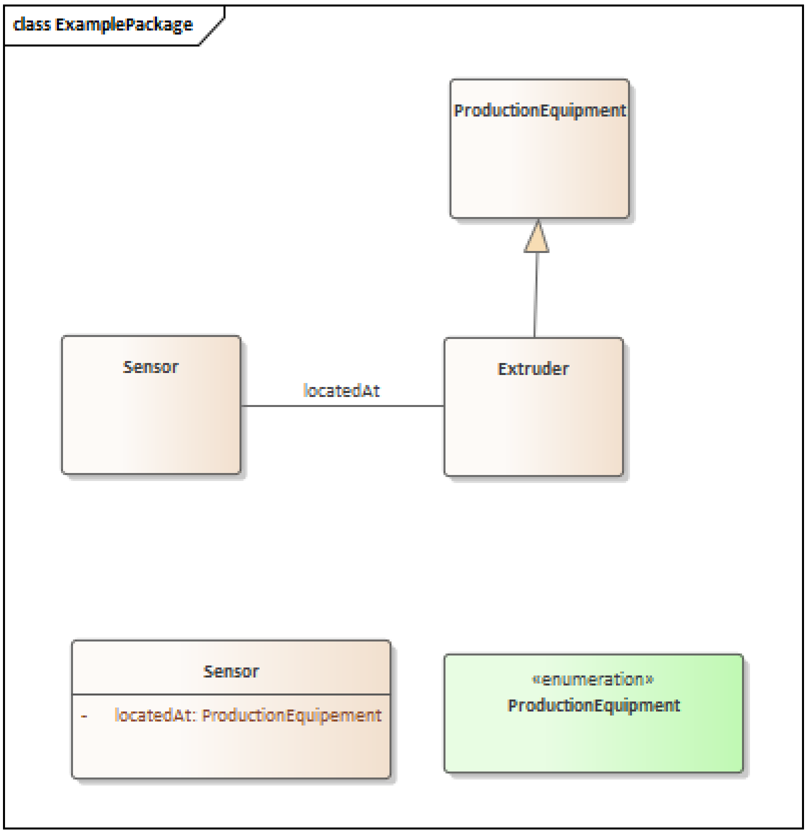

Figure 4: illustrating data modeling patterns 1 and 2

Both patterns have their benefits and drawbacks. It was decided to apply pattern 1 with explicit classes to make two aspects possible:

- Explicit representation in the UML graphical language that a sensor, mounted on an extruder, is located at an extruder
- To ease the expression of unique details about the kind of production equipment

The choice for pattern 1 has an additional confidentiality drawback. By adding the production equipment named in the data specification, the data model becomes subject to confidentiality. It may not reveal any critical information, but it is at least an entry point to the manufacturing process by naming production equipment. Pattern 2 is at the first glance less susceptible for a breach of confidentiality. The difference between both has not been further investigated, as the objective is not to make a data model that is NDA safe, but a data model that is capturing the provided information adequately. Nevertheless, it is important for developers of semantical models of manufacturing processes to understand that their work is subject to confidentiality and that this impacts the design of the model and also the publishing possibilities.

Challenge 2 - related sensors

Although sensors are physically independent devices, their value and use are often dependent on each other. E.g., one sensor sets the target temperature while the other is measuring the actual value. Despite physically distinct, there is a strong relationship between both.

Neither SSN/SOSA nor SAREF Core does provide this kind of relationship in their generic approach. Therefore, this has been added with the project namespace. The setPointFor denotes that the sensor is setting the target value for the other sensor. The inverse relationship is isActualFor denotes that the sensor is providing the actual value for the sensor that is setting the target.

These relationships capture a frequently occurring relationship between sensors.

Challenge 3 – unit of measurement

The unit of measurement is according to SSN/SOSA associated with the actual value of the observation (the result). In the practice the unit is not part of the information the device is providing. It is an external interpretation the data layer has to apply. From the device perspective, it is part of the characteristics of the sensor: namely it provides data values in this unit of measurement. Where-as from the data perspective it is a characteristic from the observation (result).

This difference has to be resolved either at the data integration connector of the device with the data platform by adding per sensor the corresponding unit of measurement as a fixed value, either by extending the data model with a new sensor property (ssn:hasSystemProperty) denoting the unit of measurement the values are being provided.

By preference, the first approach is taken, but if technical reasons prevent this to be achieved easily the second approach could be taken.

2.1.4 Publishing the model

At the level of publishing the semantic data model as a human readable specification we rely on the OSLO toolchain[1]. This Open-Source software is a tooling developed and maintained by the Flemish Government, Belgium, as part of their data interoperability program, Open Standards for Linked Organisations. Its objective is to provide a coherent way to turn UML diagrams into a coherent semantical data standard. The toolchain is being adopted by the EU data interoperability program SEMIC[2] .

The OSLO toolchain fits the purpose of providing human readable representation of the semantics of the sensor data. Nevertheless, there were two main contributions made to the toolchain by AI-PROFICIENT:

1. The creation of the W3C RESPEC[3] style for the look and feel of data specifications
2. The ability to handle private GitHub repositories.

[1] https://github.com/Informatievlaanderen/OSLO-publicationenvironment-template
[2] https://semic.eu
[3] https://respec.org/docs/

The first is a natural adaptation to not rely on the Flemish Government, Belgium, look and feel. But use a look and feel which has been adopted by the Semantic Web community as a good practice to describe data standards. This contribution to the OSLO toolchain provides that community with an alternative to produce data standards with a high quality “neutral” look and feel.

The second is more motivated from the confidentiality nature of the project. If the content of the semantical model is sensitive, then not only the final result should have restricted access, but also the source should be restricted. Technically the OSLO toolchain is a setup as an CI/CD automation between public GitHub repositories. AI-PROFICIENT has adapted the automation to also access the GitHub repositories via the GitHub API. This allows to use the security features of GitHub and thus enable the use of private repositories. This extension allows thus to address the confidentiality of the data standard.

To provide an external perspective to the partners of the sensor data, the data specifications are also published on a dedicated domain from the project: https://data.ai-proficient.eu. This domain is only accessible on a need-to-know basis, to reduce the risk of confidentiality breach. But it enables to become a shared knowledge about the sensor information available.

2.2 The sensor data

As explained in section 2, Figure 5: the data layer implemented based on the semantic data model shows there are two knowledge bases: the first concerning the description of sensors with their location within the manufacturing process and the second regarding the actual values measured by the operating sensors.

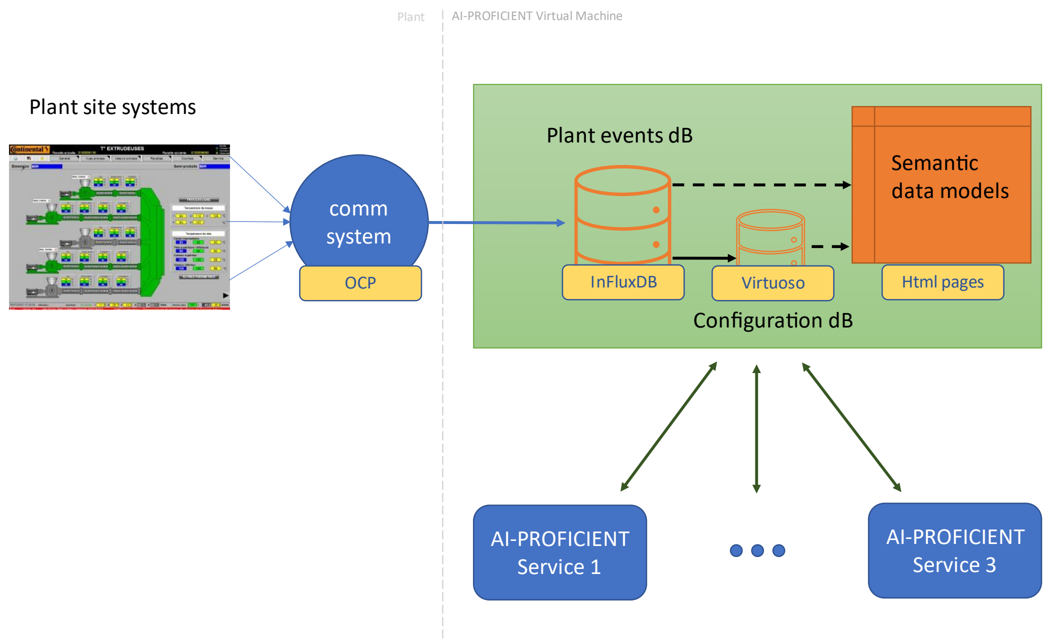

Figure 5: the data layer implemented based on the semantic data model

2.2.1 Configuration DB and the Semantic data models

The first knowledge base is to a largely human created. The production equipment engineer is responsible to encode in a digital system (e.g., a CAD/CAM system) the actual setup of the sensors. Within the AI-PROFICIENT platform this knowledge will be made available using an RDF (Resource description framework) store. The RDF store is adequate for the objective to provide the context to a measurement. But obviously it is not a replacement for a CAD/CAM system. The provided data by the use case partners Continental and INEOS is in the line of this setting: sufficient to provide context, but at the detail of a CAD/CAM drawing.

An RDF store is a database which manages data provided as RDF. An RDF store provides a query interface according to the SPARQL standard. RDF stores can be considered as NoSQL databases or graph databases satisfying the RDF data format. RDF stores are not so well-known as their relational counterparts, despite they already being used in commercial services for more than 2 decades. A well-known public RDF store is DBPedia. DBPedia is an RDF representation of Wikipedia. This illustrates that if managed, RDF stores can provide reliable services.

The advantage of using an RDF store for the sensor descriptions and their location in the manufacturing process, is that no pre-functional analysis of the usage pattern of the data must be done. SPARQL provides the means to start a query from any node in the knowledge graph.

2.2.2 Sensor measurements database (plant events)

The second knowledge are the measurements made by the sensors. This knowledge is not produced by humans, but by devices. Although it depends on the production process and the capabilities of the sensors, this can be considered as a stream of data that is being ingested in the data platform. An appropriate database technology is required to capture this need. In Deliverable D5.1 the selection for InfluxDB has been made.

2.2.3 An interconnected data layers

The usage of two distinct knowledge bases for storing interconnected data requires agreements between both. These agreements have as purpose to provide a common data experience for the data user, even though the technology and associated query language are distinct.

Semantic Web Technology addresses this challenge by using system neutral identifiers: URIs. By using URIs as identifiers the identifier is not bound anymore to the storage system, but to the data accessibility system.

In AI-PROFICIENT, the provided sensor observation data consists of the information shown in Example 1. The native data format is not a json, but for the purpose of the explanation the activity it is represented here in a compatible json representation.

```
{
    "madeBySensor": "sensor-T-E-234",
    "result":
        "value" : "50.3"
      },
    "resultTime": "2022-05-12T23:04:12Z",
 }
```

Example 1: sensor data from the OCP

To interconnect this representation with the semantic repository which contains the description of the sensor *sensor-T-E-234,* a bridge has to be provided. This bridge is JSON-LD. JSON-LD is a mapping language that allows to convert a JSON structure into an RDF structure. The mapping is called the JSON-LD context.

In our AI-PROFICIENT case the mapping of the above example to interconnect with the AI-PROFICIENT core data model (SSN/SOSA observation) is rather straightforward. This bridge can be technically made using JSON-LD. JSON-LD is a W3C recommendation that defines a language for mapping JSON structures into the Semantic Web. With the language a standard transformation is defined so that the result of the transformation is well-defined.

The application of this methodology implies the following:

1. Provide a JSON-LD context, i.e., the mapping from JSON to the semantic data model, and
2. Enhance the information with additional contextual information to facilitate the mapping and the users of the data.
3. Define a shared URI identifier space

Example 1 is then transformed into Example 2.

A detailed reference specification for a serialization of Linked Data in JSON-LD can found here: https://www.w3.org/TR/json-ld/

```
{

"@context”: “https://system.com/observation.jsonld”,
"@type” : “Observation",
“@id” : “sensor-T-E-234+2022-05-12T23+04+12Z”,

"madeBySensor": "sensor-T-E-234",
   "result": {
       "value" : "50.3”
      },
"resultTime": "2022-05-12T23:04:12Z"

 }
```

Example 2: semantical mapping of the sensor data

Three elements have been added:

- The explicit indication that the JSON structure an Observation represents
- An identifier for the observation. Each observation is unique in its own right, and thus can be referenced to. The identifier is built from the sensor and the time of measurement.
- The context. In the example, the URL will contain the actual context mapping.

As explained in challenge 3, the unit of measurement is not present in the source data. Ideally the final JSON should become as illustrated in example 3:

```
{

"@context”: “https://system.com/observation.jsonld”,
"@type” : “Observation",
“@id” : “sensor-T-E-234+2022-05-12T23+04+12Z”,
"madeBySensor": "sensor-T-E-234",
   "result": {
     "value" : "50.3”,
     "unitText": "Cel",
     "standardUnit" : "http://qudt.org/vocab/unit/DEG_C"

      },

"resultTime": "2022-05-12T23:04:12Z"

  }
```

Example 3: sensor data extended with unit of measurement

The last step to create a common data space is the use of the same identifiers. In the Semantic Web RDF identifiers are URIs. The identifiers provided by the manufacturing are unique references like “sensor-T-E-234”. In Continental, but the same approach also occurs in INEOS Geel, such references are an encoding of the sensor, its basic capabilities and location. They become in many cases part of the common speak of that plant.

Instead of applying a visible transformation in the data to match the URI requirements by the Semantic Web, the transformation of the identifier to a URI is included in the JSON-LD context. In this way the JSON representation stays closer to the source representation and it is up to the users that need to perform queries across the two knowledge bases to apply the context mapping.

The shared JSON-LD context is shown below in Figure 6.

```
"@context":
  {
  "sosa": "http://www.w3.org/ns/sosa/",
  "Observation": "sosa:Observation",
    "@base": "https://conti.com/id/",
    "@vocab": "https://conti.com/ns/",
    "madeBySensor": "sosa:madeBySensor",
    "result": "sosa:hasResult",
    "resultTime": "sosa:resultTime",
    "Result": "sosa:Result",
    "value" : "https://schema.org/value",
    "unitText": "https://schema.org/unitText",
    "standardUnit": https://schema.org/unitCode
},
```

Figure 6 : the JSON-LD context

3 Use Cases implementation of sensor data platform

3.1 CONTI

The AI PROFICIENT sensor data model has been deployed at the facility of Continental. Most of the technical considerations resulted from this application.

The following future activities could be investigated:

- Providing a French perspective to the data instead of an English.
- Dereferenceable URIs
- Support for image observations

The semantical model is of interest as a technical documentation to the data that is being accumulated in the sensor data knowledge bases. However there exists a language barrier where within the plant the lingua franca is French, while for the AI-PROFICIENT researchers, this is English. The semantical model is a way to bridge this, certainly, if the same data specification can be represented in different languages. The publication toolchain is capable for that, but it requires the language expertise from the partners to translate the information in French.

Deferenceable URIs are considered a best practice within the Semantic Web[4]. For the terminology in the data model is can be reached via the OSLO Toolchain publishing and having a (public) domain, in our case data.ai-proficient.eu. But it can also be applied for the data entities like sensors or observations. That means that instead of performing a DB query to the knowledge base to find the details of a sensor, a simple HTTP(S) lookup can be performed.

For some use-cases image information will be analysed to make decisions upon. One can consider these images also as observations made by a sensor. And thus, one could consider to provide access to the image information via the same data sensor platform. This has not been elaborated and tested as in contrast to a temperature the amount of data an image represents is different.

3.2 INEOS

Due to different levels of progress, at the time of writing of the deliverable only a high-level assessment of the applicability to the INEOS use cases has been performed. The Use Case INEOS 2 provides a very similar data landscape: sensors that are measuring the chemical process operations.

4 FIDES ontology for AI Accountability

Where-as section 2 was devoted to a common data perspective for sensor data, this section addresses the challenge of improving the understanding and trustworthiness on AI-systems when they are supporting decision making.

Accountability is a relevant factor to advance in this trustworthiness aspect, as it enables discovering the causes that derived a given decision or suggestion made by an artificial intelligence system. Accountability can be defined as the ability to determine whether a decision was made in accordance with procedural and substantive standards and to hold someone responsible if those standards are not met[5]. This means that with an accountable AI system, the causes that derived a given decision can be discovered, even if its underlying model's details are not fully known or must be kept secret. In other

[4] https://www.w3.org/TR/dwbp/#DataIdentifiers

[5] J.A. Kroll, S. Barocas, E.W. Felten, J.R. Reidenberg, D.G. Robinson and H. Yu, Accountable algorithms, U. Pa. L. Rev. 165 (2016), 633–705.

words, the person, group or company in charge of the AI system should be able to answer questions that are related, not only to the obtained outputs (e.g., what the output result is or when the output is generated), but also to the AI procedures that led to such outputs (e.g., which data set(s) are used to train the AI system or how well the AI system performs in terms of accuracy).

However, the information needed to answer these questions is hardly ever accessible in a straightforward way. This information is scattered across multiple files, repositories, and systems, and in the worst-case scenario, is not even registered. That means that, if the person, group, or company in charge of the AI system wanted to answer the aforementioned questions, it would be very time consuming, as the person would have to be an expert or have the help of experts in different frameworks, systems, data models, repositories and query languages. As a matter of fact, the regular performance of these accountancy tasks would be infeasible.

Therefore, it seems reasonable to consider that the adequate representation of data, processes and workflows involved in AI systems could contribute to make them accountable in an easier and systematic manner. There are a variety of technologies that offer conceptual modelling capabilities to describe a domain of interest, but only ontologies combine this feature with Web compliance, formality, and reasoning capabilities[6] .

In AI-PROFICIENT, based on this semantic approach, FIDES[7] Ontology has been developed. FIDES aims to represent, structure, and set formal relations among the ML-based models and the forecasts/suggestions that conform a ML system, disposing all the necessary information to answer all the pertinent questions in terms of systems' accountability. For developing FIDES ontology, LOT (Linked Open Terms) methodology has been followed. This methodology follows two main steps for development: a first step of requirements specifications and a second implementation step. And two additional steps for publication and maintenance aspects.

The aim of the requirements specification process is to state why the ontology is being built and to identify and define the requirements the ontology should fulfil. Taking as input the documentation and data provided by domain experts and users, the ontology development team generates a first proposal of ontological requirements written in the form of competency questions or statements.

While the implementation process aims to build the ontology using a formal language, based on the ontological requirements identified by the domain experts, enhancing the reuse of existing ontologies whenever it is possible. An evaluation comprising syntactic, modelling, or semantic errors and that the ontology fulfil all the requirements scheduled for the ontology during the requirements specification activity.

The publication process aims to ensure the ontology to be in an accessible both as a human-readable documentation and a machine-readable file from its URI, well documented. While the maintenance will ensure the update of the ontology when new requirements appear, or errors are identified.

The following sections provide the details for each of the LOT three first steps, all except the maintenance since it will happen in the future.

4.1 FIDES Requirements

FIDES envisions the representation of two main knowable topics: the forecast/suggestion made by the ML-based model, and the procedure followed by such a ML-based model for making the forecast/suggestion. To formalise the information requirements for each knowable topic, the Competency Questions (CQ) were used as proposed by LOT. For CQs definition, a team of 4 AI experts and 2 ontologists met several times.

[6] D. Oberle, How ontologies benefit enterprise applications 5(6) (2014), 473–491. doi:10.3233/SW-130114.
[7] Fides was the Roman goddess of trust.

For the procedure followed to construct ML-based model for making forecasts, the team stablished that the relevant information could be divided in, on the one hand, the information addressing the data used to train the ML-based model, and on the other, the information concerning the details of the procedure implemented by the ML-based model. Determining for the characterization of the training data in terms of the features, including the amount of data used, the dependent and independent variables considered, as well as statistical characteristics such as the variance, mean or median of the data. Some CQ examples are listed below:

- Which is the frequency of a given model's training data?
- Which is number of observations used for training a given model?
- When was the last data point within a given model's training data collected?

Regarding the ML-based model's procedure details, information related to the algorithm used and its hyperparameters was identified as relevant, as well as the performance assessed in development time. This can be procured by CQs of the following style:

- Which is the base algorithm of the predictive model?
- Which is are the hyperparameter values of the predictive model?
- Which is the RMSE of the predictive model?

For the forecasts/suggestions made by a ML-based model, the team determined that details of the prediction/suggestion should be available as well as the context on when it was occurred. Some examples could be the following ones:

- Which is the value of a given forecast?
- When was a given forecast generated?
- What is the forecast’s error metric/value?

After several rounds a set of 28 CQs for the procedure topic and 9 CQs for the deployment and execution part were stablished as the starting CQs set.

4.2 FIDES Implementation

Starting from the 37 CQs from the requirements steps, a list of terms presented in Table 1 was generated representing the main concepts that should be present in the FIDES ontology.

Table 1 List of relevant terms for the ontology

Software
Version
Creator
Contributor
Docker Container
Source
Feature
Response feature
Run
Input
Dataset Characteristic
Operating System (OS)
Associated OS
Procedure
Triggered on
Stores
Prediction value
Prediction error value
Generation time

Temporal context
Result

Considering the terms, and following the ontology reuse best practice[8] potential ontologies to be reused search was carried out consulting: Linked Open Vocabularies (LOV) and LOV4IoT ontology catalogues, and Google Scholar and ScienceDirect research databases. The selected ones for the implementation were the following 3 Ontology Design Patterns (ODPs) and 2 ontologies:

- AffectedBy ODP: is intended to support data analysts in the discovery of relevant variables that affect the environment of a given space or another feature of interest.
- EEP ODP: extends the AffectedBy ODP to support data analysts making further queries to discover sensors or actuators that observe or act on a given property of a space.
- RC ODP: aims to represent the results of the executions defined in the EEP ODP as well as their contexts
- ML-Schema Ontology: provides a set of classes, properties, and restrictions to represent different aspects of ML processes. On the one hand, resources to describe the data used as input and their characteristics and quality are offered. On the other, resources to describe the implementations, algorithms used to develop models and their hyperparameters are defined. Finally, the developed models, their characteristics and the evaluation obtained in the training phase can also be represented with this ontology.
- MEX Performance Ontology: MEX is a vocabulary modeled to facilitate interoperability between results of machine learning experiments published on the Web. It is a lightweight and flexible schema for publishing machine learning outputs metadata

With these resources an important part of the necessary concepts and relations for FIDES were covered. However, mainly relations, where not possible to define through some of these existing ontologies. In those cases, new classes or properties have been added to FIDES.

The result is an ontology with 43 classes. 32 object properties and 34 data properties. An excerpt is presented in

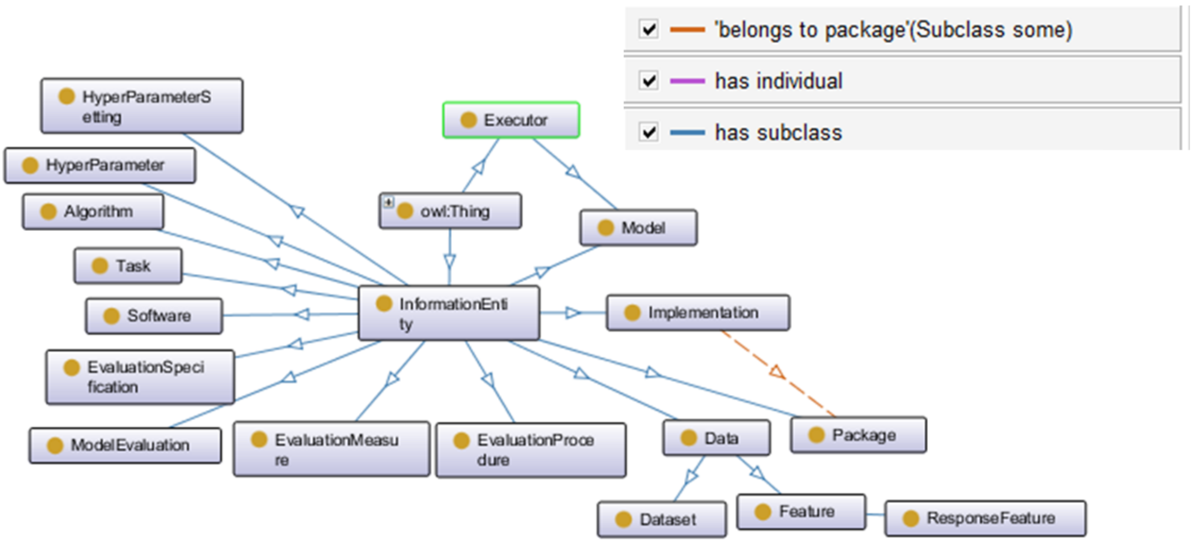

Figure 7: FIDES Ontology excerpt

[8] E. Simperl, Reusing ontologies on the Semantic Web: A feasibility study, Data & Knowledge Engineering 68(10) (2009), 905–925. doi:10.1016/j.datak.2009.02.002.

4.3 FIDES Publication

The complete implementation of FIDES is available in https://w3id.org/fides. For this, the new classes and properties have been first properly documented, the corresponding checking ensuring there is no syntactic neither semantic error has been performed, and using WIDOCO tool, the necessary.

4.4 In Use

To validate the ontology, CONTI2 use case predictive model has been used to populate the ontology and to validate that all the necessary information can be properly represented according to FIDES to get a correct answer for the defined CQs, at least for the model development procedure. The validation for the prediction/suggestion part will be further performed when model is running and providing forecasts.

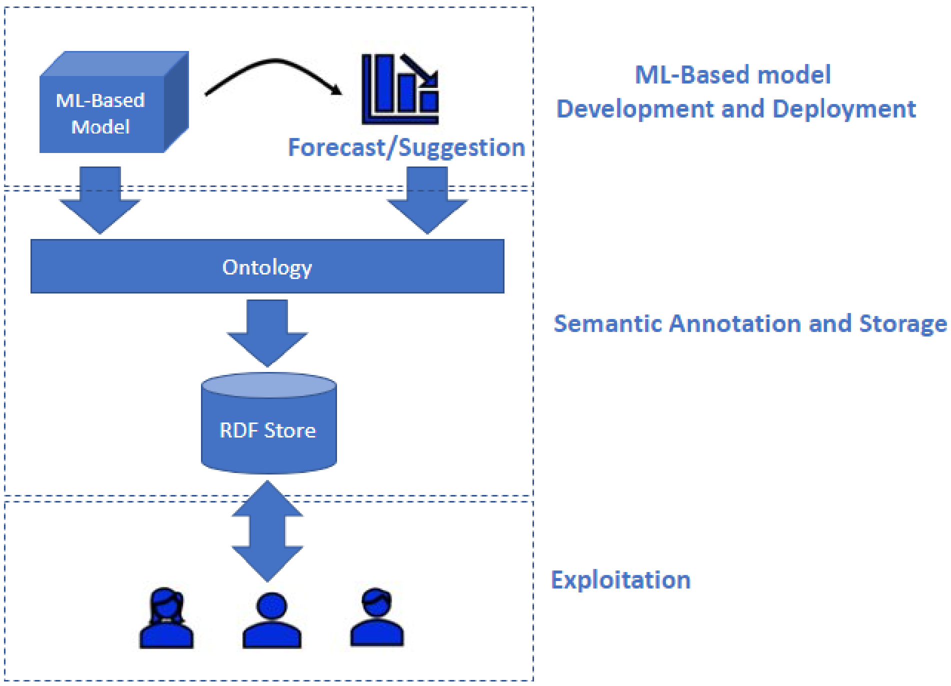

Figure 8: AI-Models Accountability Semantic Approach

All in all, FIDES ontology is pretended to be used in the context of T4.4, as core element of a semantic framework for AI-models accountability. The semantic approach consists of three phases as shown in Figure 4. The first phase is related to the development of the AI-based model that will solve the problem at hand, and when ready it will be deployed. This phase has nothing to do with semantic aspects. In the second phase, by contrast, the relevant information related to the developed ML-based model will be annotated according to FIDES ontology and stored in an RDF store. In a similar way, every forecast/suggestion will be also mapped to FIDES ontology and included in the RDF store, as they come. Finally, in the last phase, the accountability information in the RDF repository will be consumed, by an end user's GUI (Graphical User Interface) that facilitates the retrieval of the information about ML systems accountability.

5 Conclusion

This report documented the activities around making manufacturing data more understandable, and shareable. This consisted of two distinct areas, yet connected within AI-PROFICIENT. The first targeted the sensor data readings, while the second the decision-making process within the AI-systems.

The AI-PROFICIENT experience has learned that while sensor data seems any easy target given the existence of a good international basis of standards, there are details to be resolved to create a wellfounded semantical basis. The FIDES ontology targets a more unknown area, and therefore it is more explorative of nature.

This raises future work that remains is to investigate: how both areas can be interconnected with each other.

6 Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957391.

7 References

[1] K. Janowicz, A. Haller, S. Cox, D. Phuoc and M. Lefrançois, SOSA: A lightweight ontology for sensors, observations, samples, and actuators, Journal of Web Semantics, 2018.06.003..

[2] “Virtuoso Universal Server,” [Online]. Available: https://virtuoso.openlinksw.com/.

[3] “JSON-LD,” [Online]. Available: https://json-ld.org/.

[4] “RDF,” [Online]. Available: https://www.w3.org/TR/rdf11-concepts/.

[5] “SPARQL,” [Online]. Available: https://www.w3.org/TR/sparql11-query/.

[6] “SHACL,” [Online]. Available: https://www.w3.org/TR/shacl/.

[7] “Purdue model,” [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.194.6112&rep=rep1&type=pdf.

[8] “InfluxDB,” [Online]. Available: https://www.influxdata.com/.

[9] “Grafana,” [Online]. Available: https://grafana.com/.

[10] “Merriam Webster - Interoperability,” [Online]. Available: https://www.merriam-webster.com/dictionary/interoperability.

[11] H. Kubicek, R. Cimander and H. J. Scholl, “ Interoperability in Government,” in *Organizational Interoperability in E-Government*, Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-22502-4_2, 2011.

[12] “SSN/SOSA, OGC/W3C Semantic Sensor Network Ontology.,” [Online]. Available: https://www.w3.org/TR/vocab-ssn .

[13] “OM, ISO 19156 Observations and Measurements,” [Online]. Available: https://www.iso.org/standard/32574.html .

[14] “SAREF,” [Online]. Available: https://saref.etsi.org/core/v3.1.1/ .

[15] “OSLO Toolchain,” [Online]. Available: https://github.com/Informatievlaanderen/OSLO-publicationenvironment-template.