eman ta zabal zazu

Universidad del País Vasco | Euskal Herriko Unibertsitatea

DOCTORAL THESIS

CONTRIBUTIONS FOR THE EXPLOITATION OF SEMANTIC TECHNOLOGIES IN INDUSTRY 4.0

Víctor Julio Ramírez Durán

Supervised by

Dr. Idoia Berges

Prof. Dr. M. Aranzazu Illarramendi

Donostia-San Sebastián, June 2021.

eman ta zabal zazu

Universidad del País Vasco — Euskal Herriko Unibertsitatea

DOCTORAL THESIS

CONTRIBUTIONS FOR THE EXPLOITATION OF SEMANTIC TECHNOLOGIES IN INDUSTRY 4.0

Víctor Julio Ramírez Durán

Supervised by

Dr. Idoia Berges

Prof. Dr. M. Aranzazu Illarramendi

A dissertation submitted to the Department of Computer Languages and Systems of the University of the Basque Country UPV/EHU

Donostia-San Sebastián, June 2021.

Universidad del País Vasco / Euskal Herriko Unibertsitatea
NAZIOARTEKO BIKAINTASUN CAMPUSA / CAMPUS DE EXCELENCIA INTERNACIONAL

TESI ZUZENDARIAREN BAIMENA TESIA AURKEZTEKO	AUTORIZACIÓN DEL/LA DIRECTORA/A DE TESIS PARA SU PRESENTACIÓN

1. Zuzendariaren izen-abizenak /Nombre y apellidos de la directora 1: Idoia Berges González

IFZ /NIF: 44561356Y

2. Zuzendariaren izen-abizenak /Nombre y apellidos de la directora 2: María Aranzazu Illarramendi Echave

IFZ /NIF: 15939031P

Tesiaren izenburua / Título de la tesis: Contributions for the exploitation of Semantic Technologies in Industry 4.0

Doktorego programa / Programa de doctorado: Ingeniería Informática

Doktoregaiaren izen-abizenak / Nombre y apellidos del doctorando: Víctor Julio Ramírez Durán

Unibertsitateak horretarako jartzen duen tresnak emandako ANTZEKOTASUN TXOSTENA ikusita, baimena ematen dut goian aipatzen den tesia aurkez dadin, horretarako baldintza guztiak betetzen baititu.

Visto el INFORME DE SIMILITUD obtenido de la herramienta que a tal efecto pone a disposición la universidad, autorizo la presentación de la tesis doctoral arriba indicada, dado que reúne las condiciones necesarias para su defensa.

Tokia eta data / Lugar y fecha: Donostia, 26 de mayo de 2021

Sin. / Fdo.: Tesiaren zuzendariak / Las directoras de la tesis

Universidad del País Vasco · Euskal Herriko Unibertsitatea
NAZIOARTEKO BIKAINTASUN CAMPUSA
CAMPUS DE EXCELENCIA INTERNACIONAL

AUTORIZACIÓN DE LA COMISIÓN ACADÉMICA DEL PROGRAMA DE DOCTORADO

La Comisión Académica del Programa de Doctorado en **Ingeniería Informática** en reunión celebrada el día **24 de mayo de 2021** ha acordado dar la conformidad a la presentación de la Tesis Doctoral titulada: **Contributions for the exploitation of Semantic Technologies in Industry 4.0** dirigida por las **Dras. Idoia Berges González** y **María Aranzazu Illarramendi Echave y** presentada por **Don Víctor Julio Ramírez Durán** adscrito al Departamento de **Lenguajes y Sistemas Informáticos.**

En Donostia a 24 de mayo de 2021

EL/LA RESPONSABLE DEL PROGRAMA DE DOCTORADO

Firmado por ELENA LAZKANO ORTEGA - 34091823G el día 24/05/2021 con un certificado emitido por Herritar eta Erakundeen CA - CA de Ciudadanos y Entidades (4)

Fdo.: Elena Lazkano Ortega

Universidad del País Vasco Euskal Herriko Unibertsitatea

NAZIOARTEKO BIKAINTASUN CAMPUSA
CAMPUS DE EXCELENCIA INTERNACIONAL

AUTORIZACIÓN DEL DEPARTAMENTO

El Consejo del Departamento de **Lenguajes y Sistemas Informáticos** en reunión celebrada el día **26 de mayo de 2021** ha acordado dar la conformidad a la admisión a trámite de presentación de la Tesis Doctoral titulada: **Contributions for the exploitation of Semantic Technologies in Industry 4.0** dirigida por las **Dras. Idoia Berges González** y **María Aranzazu Illarramendi Echave y** presentada por **Víctor Julio Ramírez Durán** ante este Departamento.

En Donostia a 26 de mayo de 2021

VºBº DIRECTOR/A DEL DEPARTAMENTO

Fdo.: MONTSERRAT HERMO

SECRETARIO/A DEL DEPARTAMENTO

Fdo.: ANA ARRUARTE LASA

LENGOAIA ETA SISTEMA INFORMATIKOAK SAILA U.P.V. E.H.U. DPTO. DE LENGUAJES Y SISTEMAS INFORMÁTICOS

Universidad del País Vasco · Euskal Herriko Unibertsitatea
NAZIOARTEKO BIKAINTASUN CAMPUSA · CAMPUS DE EXCELENCIA INTERNACIONAL

ACTA DE GRADO DE DOCTOR O DOCTORA
ACTA DE DEFENSA DE TESIS DOCTORAL

DOCTORANDO DON: **Víctor Julio Ramírez Durán**____

TITULO DE LA TESIS: **Contributions for the exploitation of Semantic Technologies in Industry 4.0** ____

____

El Tribunal designado por la Comisión de Postgrado de la UPV/EHU para calificar la Tesis Doctoral arriba indicada y reunido en el día de la fecha, una vez efectuada la defensa por el/la doctorando/a y contestadas las objeciones y/o sugerencias que se le han formulado, ha otorgado por___________________la calificación de:
unanimidad ó mayoría

SOBRESALIENTE / NOTABLE / APROBADO / NO APTO

Idioma/s de defensa (en caso de más de un idioma, especificar porcentaje defendido en cada idioma):

Castellano ____

Euskera ____

Otros Idiomas (especificar cuál/cuales y porcentaje) ______________________________

En __________________a _____de ____________________de _____

EL/LA PRESIDENTE/A,

Fdo.:

Dr/a: ____________________

EL/LA SECRETARIO/A,

Fdo.:

Dr/a: ______________________

VOCAL 1º,

Fdo.:

Dr/a: ______________________

EL/LA DOCTORANDO/A,

Fdo.: ______________________

A Dios, a mi familia y a mis amigos.

Summary

New technological advances such as the Internet of Things (IoT), robotics, Artificial Intelligence, among others, have driven a fourth industrial revolution, in which the data obtained from sensors located in product manufacturing machines are transmitted to data centers, where they can be analyzed in order to extract knowledge that allows improving manufactured products and adding innovation in manufacturing processes, as well as predicting future behavior and avoiding failures in said processes. An additional impetus in said fourth revolution may come from the use of semantic technologies to represent domain knowledge in an integrated, structured, and interconnected manner; in such a way that it can be interpreted by both humans and computers, thus allowing cooperative work.

In this research work, the use of Semantic technologies is promoted, in the Industry 4.0 environment, through three contributions focused on topics corresponding to intelligent manufacturing: the enriched descriptions of components, the data visualization and analysis, and the implementation of Industry 4.0 in SMEs.

The first contribution is an ontology called ExtruOnt, which contains semantic descriptions of one type of manufacturing machine (the extruder). This ontology describes the components, their spatial connections, their features, their three-dimensional representations and, finally, the sensors used to capture the data. The second contribution corresponds to a visual query system in which the ExtruOnt ontology and a 2D representation of the extruder are used to make it easier for domain experts to visualize and extract knowledge about the manufacturing process quickly and simple. The third contribution consists of a methodology for the implementation of Industry 4.0 in SMEs, oriented to the customer life cycle and enhanced by the use of Semantic technologies and 3D rendering technologies.

The contributions have been developed, applied and validated under a real manufacturing scenario. This scenario is representative in the manufacturing sector for which the application of the contributions may be relevant.

Resumen

Los nuevos avances tecnológicos como el internet de las cosas (IoT), la robótica, la inteligencia artificial entre otros, han impulsado una cuarta revolución industrial, en la que los datos obtenidos de los sensores localizados en máquinas de fabricación de productos, son transmitidos a centros de datos, donde pueden ser analizados con el fin de extraer conocimiento que permita mejorar los productos fabricados y añadir innovación en los procesos de fabricación, así como también predecir el comportamiento futuro y evitar fallos en dichos procesos. Un impulso adicional en dicha cuarta revolución puede provenir del uso de tecnologías semánticas para representar el conocimiento del dominio de una forma integrada, estructurada e interconectada; de tal forma que pueda ser interpretable tanto por los humanos como por las computadoras, permitiendo así el trabajo cooperativo.

En este trabajo de investigación se promueve la utilización de las tecnologías semánticas, en el entorno de la Industria 4.0, a través de tres contribuciones enfocadas en temas correspondientes a la fabricación inteligente: las descripciones enriquecidas de componentes, la visualización y el análisis de los datos, y la implementación de la Industria 4.0 en PyMEs.

La primera contribución es una ontología llamada ExtruOnt, la cual contiene descripciones semánticas de un tipo de máquina de fabricación (la extrusora). En esta ontología se describen los componentes, sus conexiones espaciales, sus características, sus representaciones en tres dimensiones y, finalmente, los sensores utilizados para capturar los datos. La segunda contribución corresponde a un sistema de consulta visual en el cual se utiliza la ontología ExtruOnt y una representación en 2D de la extrusora para facilitar a los expertos de dominio la visualización y la extracción de conocimiento sobre el proceso de fabricación de una manera rápida y sencilla. La

tercera contribución consiste en una metodología para la implementación de la Industria 4.0 en PyMEs, orientada al ciclo de vida del cliente y potenciada por el uso de tecnologías Semánticas y tecnologías de renderizado 3D.

Las contribuciones han sido desarrolladas, aplicadas y validadas bajo un escenario de fabricación real. Este escenario es representativo en el sector manufacturero para el cual la aplicación de las contribuciones puede ser relevante.

Laburpena

Aurrerapen teknologiko berriek, hau da, gauzen internet (IoT), robotika eta adimen artifiziala, besteak beste, laugarren industria-iraultza bultzatu dute. Produktuak fabrikatzeko makinetan kokatutako sentsoreetatik lortutako datuak datu-zentroetara transmititzen dira eta bertan aztertu daitezke, fabrikatutako produktuak hobetzea eta fabrikazio-prozesuetan berrikuntza gehitzea ahalbidetzen duen ezagutza ateratzeko, eta baita etorkizuneko portaera aurreikusteko eta aipatutako prozesuetan hutsegiteak ekiditeko ere. Teknologia semantikoen erabilerak bultzada gehigarri bat eman diezaioke laugarren iraultzari, domeinuko ezagutza modu integratu, egituratu eta elkarri lotuta irudikatuz, gizakiek zein ordenagailuek interpretatu ahal izateko moduan, eta era honetan lan kooperatiboa ahalbidetuz.

Ikerketa lan honetan teknologia semantikoen erabilera sustatzen da 4.0 Industriaren ingurunean, fabrikazio-adimendunari dagozkion gaietan oinarritutako hiru ekarpenen bidez: osagaien deskribapen aberastuak, bistaratzea eta datuen analisia, eta 4.0 Industriaren inplementazioa ETEetan.

Lehen ekarpena ExtruOnt izeneko ontologia da, fabrikazio-makina mota baten (estrusio-makina) deskribapen semantikoak biltzen dituena. Ontologia honek osagaiak, haien konexio espaziala, haien ezaugarriak, hiru dimentsiotako irudikapenak eta, azkenik, datuak jasotzeko erabilitako sentsoreak deskribatzen ditu. Bigarren ekarpena ikusizko kontsulta-sistemari dagokio, ExtruOnt ontologia eta estrusio-makinaren 2D irudikapena erabiltzen dituena, domeinuko adituei fabrikazio-prozesuari buruzko ezagutza-erauzketa eta bistaratzea era azkar eta xamur batean egitea ahalbidetuko diena. Hirugarren ekarpena 4.0 Industria ETEetan inplementatzeko metodologia bat da, bezeroaren bizi-ziklora zuzenduta eta teknologia semantikoen eta 3D errendatze-teknologien erabilerarekin hobetua.

Ekarpenak fabrikazio-agertoki erreal batean garatu, aplikatu eta balioztatu dira. Agertoki hori fabrikazio-sektorearen adierazgarria da, non ekarpenen aplikazioa garrantzitsua izan daitekeen.

Acknowledgements

I would like to begin this dissertation by thanking several people who were fundamental in the realization of this doctoral thesis.

First and foremost, I would like to express my gratitude to my supervisors, Arantza Illarramendi and Idoia Berges, for providing me with the opportunity to pursue this PhD and for their invaluable support and guidance throughout the research process. Each of these investigations was greatly enriched by their suggestions. I would also like to express my gratitude to Jesús Bermúdez for recommending me for this project and for the advice and assistance he provided.

Many thanks to all my colleagues in the BDI research group, especially Borja Diez and Kevin Villalobos for their help with the development of user interfaces and the collection of relevant data. Thanks also to Ernesto Jiménez-Ruiz at the Department of Computer Science of City, University of London for accepting me as a remote visiting researcher, for his support and guidance; the online experience was very enriching. This work would not have been possible without the cooperation of industrial partners related to this project. Thank you so much to Fernando Sáenz and Gurutz Glafarsoro for their willingness to help and generosity.

Finally, I want to express my heartfelt gratitude and love to my family, and especially Azucena, my wonderful wife, for their unwavering support and affection and for accompanying me on this journey.

Funding

The author of this dissertation wants to thank the Spanish Ministry of Economy and Competitiveness for their support funding this research.

Contents

List of Figures

List of Tables

Part I

Work Description

1. Introduction

Industry is known as the part of the economy that produces highly mechanized and automated material goods [LFK+14]. It has undergone considerable technological leaps, even since the beginning of industrialization. These leaps have led to multiple paradigm shifts over time and have been known as industrial revolutions. The first industrial revolution included mechanization, which greatly influenced the proliferation of the nascent industry and the reduction of production times. The second focused on the exploitation of new energy sources such as gas and electricity, as well as the inclusion of new materials such as steel and oil. The third included the advances in information and communication technologies, the use of renewable energies and the development of the intelligent electrical energy distribution network (Smart Grid) [Mow09]. Today, we are witnessing the emergence of a fourth industrial revolution, which is the result of the convergence of a number of exponential technologies such as additive manufacturing, augmented reality, Artificial Intelligence, Big Data, and Internet of Things (IoT) among others, which are erasing the divisions between the digital and the physical.

1.1 Industry 4.0

The term Industry 4.0 refers to the so-called Fourth Industrial Revolution, which is born from the technological evolution driven by the development of embedded systems, their connectivity and the timely convergence of the virtual and physical world. It provides capabilities for the integration of things, information and people which can lead to a qualitative leap in the production and use of goods and services. This term (Industry 4.0) appears for the first time with the presentation of the document

"Recommendations for implementing the strategic initiative Industry 4.0", published in April 2013 [KWH13], and responds to a German strategy that seeks to continue leading the supply of equipment and solutions for industrial production and their application in industrial environments through the integration of value chains and the digitization of the entire production process.

From that moment on, many countries and regions have been working aligning efforts and policies to take advantage of the opportunities offered by this new scenario. In parallel, in the field of manufacturing and related to this movement, Europe is promoting the so-called "Factories of the Future" through the EFFRA[1] (European Factories of the Future Research Association) as a public-private collaboration for the development of innovative projects, an initiative that was launched in 2008.

1.1.1 Impact of Industry 4.0

The Industry 4.0 initiative has been driven by a new type of economy derived from this fourth industrial revolution, where the data obtained from the interconnection of ubiquitous sensors and Internet of Things (IoT) devices are transformed into manufacturing intelligence in order to achieve meaningful improvements in all aspects of manufacturing. As in Europe, this data-driven economy is being implemented around the world through different initiatives, among which we can find "Smart Manufacturing" in USA, "Made in CHINA 2025" and "Future Manufacturing" in UK [Kus18]. These initiatives enable important business opportunities for the manufacturers, having a greater impact in three key areas: product, supply chain and customer.

Regarding the product, the use of technologies such as sensors, machine learning or robotics can transform the way in which products are manufactured. In addition, the compilation of manufacturing files with structured data on the characteristics, quality, history and status of the products allows the application of methods and tools for the exploitation of these data in order to allow manufacturers to manage, plan and predict specific circumstances that positively affect production. The application of Big Data technologies allows managing the enormous amount of data received from the production process in order to fully monitor them in real time, allowing operators to check the status of products throughout their manufacture. It is well known

[1] https://www.effra.eu/

that the data pre-processing and analysis phases consume most of the data processing efforts, therefore it is important to have efficient tools for the presentation and explanation of the results based on visualization metaphors and intelligent visual interaction paradigms. With respect to the supply chain, advanced forecasting techniques relying on internal (e.g. demand) and external (e.g. market trends) data can allow for a faster delivery time [ARS16]. Moreover, real-time information about the supply network and the logistics capabilities can allow for a more flexible planning and inventory processes, which can react to changing demand or supply situations. Finally, in the case of the customer, technologies derived from Industry 4.0 can help to obtain a better understanding of the customers, their purchasing habits and their preferences, transforming this knowledge into a better experience when selecting the right product as well as offering a better post-sale service.

1.1.2 Industry 4.0 adoption challenges in SMEs

Although the adoption of Industry 4.0 for large organizations has led to an improvement in sales results and productivity, most Small and Medium-sized Enterprises (SMEs) face new challenges in order not to lose position compared to their competitors. These challenges materialize in the form of barriers of various kinds[2]:

- **Technical:** There is a wide range of technologies that can be implemented, so the selection of the most suitable one can be overwhelming, which in turn entails a considerable effort. On the other hand, most of the factories are located in areas far from large urban centers, making the coverage of communication networks limited.
- **Economical:** Transforming a factory into a smart factory involves a considerable economic investment.
- **Cultural:** The fact that machines are becoming more and more important can have psychological repercussions on workers, who may feel displaced, a secondary actor that is nothing more than an extension of technology.
- **Legal:** The regulation of the technological aspects involved in Industry 4.0 has always lagged behind progress in this field. To solve the problem of standard-

[2] https://www.viafirma.do/barreras-pymes-industria-4-0/

ization, organizations such as the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) have been working for years on drafting standards related to the different aspects of Industry 4.0, such as cybersecurity, connectivity and robotics, among others.

1.1.3 Industry 4.0 scenario for this research work

This research work has been carried out in a real world scenario, using real data provided by Savvy Data Systems enterprise[3] in partnership with Urola Solutions enterprise[4]. It considers the initiative of implementing a data-driven servitization strategy in an extrusion-based manufacturing sector distributed around the world.

Savvy Data Systems enterprise is an information technology service provider (ITS provider) that supplies smart services to various Smart Manufacturing scenarios. It has been accompanying its clients globally in their digitization processes for 10 years, transforming the data of their production processes into valuable information, speeding up decision-making and generating digital products and services. Its objective is to encourage its clients to adopt digital technologies and strategies.

Urola Solutions is a Capital Equipment Manufacturer (CEM) that creates advanced solutions for the packaging industry by utilizing blow moulding technology. Among the products they provide, there exist various types of extruders based on their production capacity and the type of plastic used for packaging.

1.2 Semantic technologies

Semantics is defined as the branch of linguistics and logic concerned with meaning [5]. Some research date the use of semantics in computer science from the 60's [Für16], with works like the one shown in [Flo67] where semantics was used for the description of non-deterministic algorithms.

Over the years, technological advances led to the emergence of the Internet and the World Wide Web, whose content was intended exclusively for human consump-

[3] https://www.savvydatasystems.com/
[4] https://www.urolasolutions.com/
[5] Taken from the Google's English dictionary provided by Oxford Languages

tion. However, in 2001 Tim Berners-Lee presented his vision of what he would call the "Semantic Web", an evolution of the World Wide Web where information would be represented and interconnected in such a way that it was not only interpretable by humans but also by computers, thus allowing cooperative work between these actors [BLHL01].

Some of the main Semantic technologies that have helped to implement the Semantic Web and whose application has also been proven in different sectors (energy, manufacturing, medicine, etc.) are presented below.

1.2.1 Resource Description framework

The Resource Description Framework (RDF) [MM04] is a collection of World Wide Web Consortium (W3C) standards that were initially developed as a data model for metadata but are now used to describe deeply integrated data. Representing the data in RDF allows the information to be identified, disambiguated and interconnected by software and systems agents, facilitating its reading and analysis. Each RDF statement has a three-part structure (subject, predicate and object) that is made up of resources, where each resource is identified by an IRI (Internationalized Resource Identifier). IRIs are a superset of URIs (Uniform Resource Identifiers) that are intended to replace URIs in defining resources where the Universal Coded Character Set is supported. An RDF statement can state facts, relationships and data by linking resources of a different kind. Hence, using its uniform structure, just about anything can be expressed.

For example, in figure 1.1 is represented the fact that "An extruder is a type of manufacturing machine" as an RDF statement where "An extruder" is the subject of the statement, "is a type of" is the predicate and "manufacturing machine" is the object.

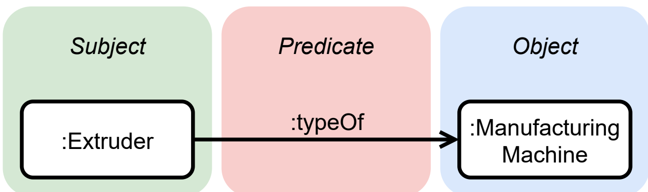


Figure 1.1: Example of an RDF triple

RDF can be serialized in a variety of formats, the most well-known is *RDF/XML* since XML serialization already existed at the time of RDF development and, therefore, several programs were able to parse, store, and serialize XML by design. However, *RDF/XML* is conceptually difficult and highly verbose compared to other standards, which makes it difficult for humans to read. Another format is *Notation3* (N3), introduced by Tim Berners-Lee in 2011, which adds functional predicates, logical implication and variables to the RDF data model, as well as offering an alternative textual syntax different to RDF/XML. One of the most easily readable format is *Turtle*, which eliminates some of the syntactic characteristics of N3 making it simpler and therefore more popular. This format also simplifies editing by hand. The next is *N-Triples*, a very basic subset of Turtle that lacks prefixes and other fancy features. However, the style is lengthy and difficult to read due to the absence of prefixes and shorthands. Lastly, *JSON-LD* (JavaScript Object Notation for Linked Data) is a JSON-based method for encoding linked data. One of their targets was to make it as easy as possible for developers to convert their current JSON to *JSON-LD*. This enables data to be serialized in a similar manner to standard JSON. Table 1 shows examples of the different named serialization formats to describe an individual of class Frequency, from OM ontology [RvAT13], representing the frequency of a motor.

1.2.2 SPARQL Protocol and RDF Query Language

SPARQL (SPARQL Protocol and RDF Query Language) [C+13] is an RDF query language and protocol. SPARQL includes a basic communication protocol, created by the W3C RDF Data Access Working Group (DAWG), that clients can use to issue SPARQL queries against endpoints, as well as a syntactically SQL-like language for querying RDF graphs via pattern matching [VC11]. It uses a method based on WSDL 2.0 for transmitting SPARQL queries to a SPARQL query processing service and returning the query results to the corresponding entity.

A SPARQL query is composed of five parts: prefix declarations (optional), a query result clause, FROM or FROM NAMED clauses (optional), a WHERE clause, and query modifiers (optional). Furthermore, a SPARQL query can have four forms: ASK, CONSTRUCT, DESCRIBE and SELECT. As if the SPARQL query were an SQL query executed against a relational database, SELECT queries return results in tabular format. The ASK form determines whether the SPARQL endpoint can return at least one result; if it can,

Serialization — Example

RDF/XML

```xml
<owl:NamedIndividual rdf:about="http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01">
  <rdf:type rdf:resource="http://www.ontology-of-units-of-measure.org/resource/om-2/Frequency"/>
  <om-2:hasValue rdf:resource="http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequencyMeasure01"/>
  <terms:description>Frequency</terms:description>
  <rdfs:comment>Frequency of a motor</rdfs:comment>
  <rdfs:label>_MotorFrequency01</rdfs:label>
</owl:NamedIndividual>
```

N3, Turtle

```
:_0MotorFrequency01
  terms:description "Frequency" ;
  om-2:hasValue :_0MotorFrequencyMeasure01 ;
  a om-2:Frequency, owl:NamedIndividual ;
  rdfs:comment "Frequency of a motor" ;
  rdfs:label "_MotorFrequency01" .
```

N-Triples

```
<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2002/07/owl#NamedIndividual> .
<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.ontology-of-units-of-measure.org/resource/om-2/Frequency> .
<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01> <http://www.ontology-of-units-of-measure.org/resource/om-2/hasValue> <http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequencyMeasure01> .
<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01> <http://purl.org/dc/terms/description> "Frequency" .
<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01> <http://www.w3.org/2000/01/rdf-schema#comment> "Frequency of a motor" .
<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01> <http://www.w3.org/2000/01/rdf-schema#label> "_MotorFrequency01" .
```

JSON-LD

```json
[{
    "@id": "http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequency01",
    "@type": [
      "http://www.w3.org/2002/07/owl#NamedIndividual",
      "http://www.ontology-of-units-of-measure.org/resource/om-2/Frequency"
    ],
    "http://purl.org/dc/terms/description": [
      { "@value": "Frequency" }
    ],
    "http://www.ontology-of-units-of-measure.org/resource/om-2/hasValue": [
      {
        "@id": "http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Demo#_0MotorFrequencyMeasure01"
      }
    ],
    "http://www.w3.org/2000/01/rdf-schema#comment": [
      { "@value": "Frequency of a motor" }
    ],
    "http://www.w3.org/2000/01/rdf-schema#label": [
      { "@value": "_MotorFrequency01" }
    ]
}]
```

Table 1.1: Examples of different RDF serialization formats

the query is answered *YES*; otherwise, it is answered *NO*. The CONSTRUCT method is almost identical to the SELECT form, but it returns an RDF graph as the response to the query. The DESCRIBE type is designed to extract data of one or more resources from a SPARQL endpoint without the need to know their underlying structure, resulting in an RDF graph. The FROM or FROM NAMED clauses are optional and specify the dataset on which the query is run.

A SPARQL query revolves around the WHERE clause. It is described in terms of a set of triple patterns, which are used to choose the triples that make up the final result. Finally, before generating the result, the set of optional query modifiers is applied to the triples chosen by the WHERE clause. The ORDER BY clause orders the results set, and the LIMIT and OFFSET clauses allow to get results in blocks, just like in SQL [VC11]. In figure 1.2 the structure of a SPARQL query to obtain the components that overlaps with the barrel BAR01 of an extruder is shown.

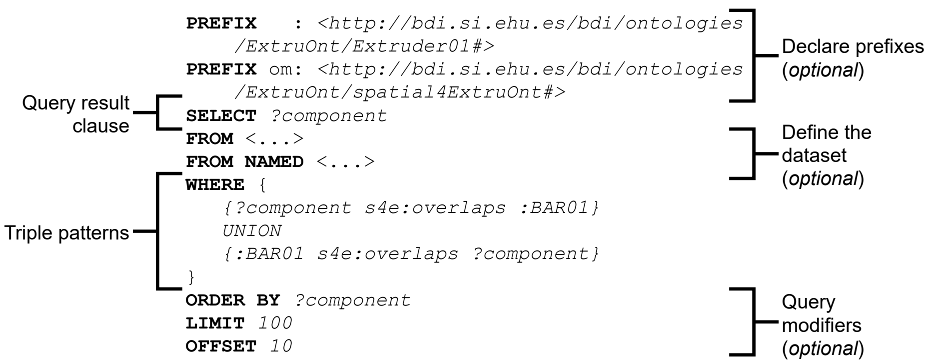


Figure 1.2: Structure of a SPARQL query

1.2.3 Web Ontology Language

The W3C Web Ontology Language (OWL) [MVH+04] is a Semantic Web language for representing rich and complex knowledge about things, set of things, and their relationships. OWL is a computational logic-based language that allows computer programs to exploit knowledge expressed using it, for example, to make implicit knowledge explicit or to check the accuracy of that knowledge. OWL is part of the W3C's Semantic Web technology stack, which includes RDF, RDFS, SPARQL, etc (see figure 1.3).

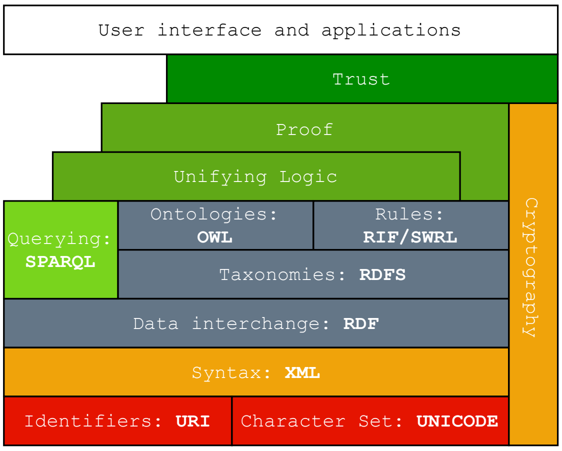


Figure 1.3: Semantic Web stack

The W3C OWL Working Group developed the current version of OWL, also known as "OWL 2", and released it in 2009, with a Second Edition published in 2012. OWL 2 is an expansion and modification of the W3C Web Ontology Working Group's original version of OWL, which was released in 2004. The OWL 2 specification includes a Document Overview[6], which acts as an introduction to OWL 2, outlines the relationship between OWL 1 and OWL 2, and provides a Documentation Roadmap[7] as an entry point to the remaining deliverables.

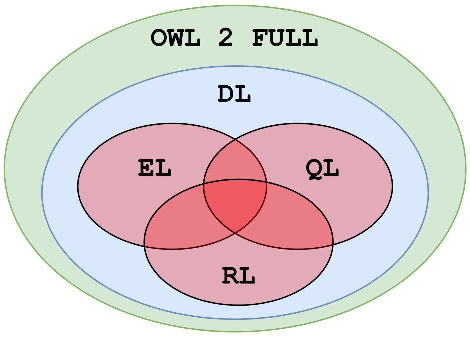


Figure 1.4: Relation between OWL 2 profiles

There are trimmed versions of OWL 2 (figure 1.4), also called profiles, fragments or sublanguages, that limit the power of expressiveness to improve the efficiency of reasoning [MGH+09]. Those profiles are:

[6] http://www.w3.org/TR/2012/REC-owl2-overview-20121211/
[7] http://www.w3.org/TR/2012/REC-owl2-overview-20121211/#Documentation_Roadmap

- **OWL 2 EL:** Useful for describing concepts with a large number of properties, relationships and individuals. There are reasoning algorithms for this profile which have shown high scalability when implemented.

- **OWL 2 QL:** It is designed for applications that work with a lot of instance data and where query answering is the most important activity. Rewriting requests to a traditional relational domain language may also be used to enforce query answering.

- **OWL 2 RL:** Oriented to applications that can sacrifice a bit of expressive power in exchange for scalable reasoning. It fits for those descriptions with a large number of individuals but very few concepts.

The choice of which profile is the most suitable for an application, depends on the structure to be used to describe the concepts and the reasoning tasks to be implemented.

1.2.4 Ontologies

In 1993, Thomas R. Gruber defined an ontology as *"an explicit specification of a conceptualization"* [Gru93]. However, in 1997 and using the definition given by Gruber, Willem N. Borst defined an ontology as *"a formal specification of a shared conceptualization"* emphasizing the fact that there must be agreement on the conceptualization that is specified [BB97]. Finally, Rudi Studer et al. in 1998 [SBF98], defined an ontology as *"a formal, explicit specification of a shared conceptualisation"* as a consensus between the previous definitions and also, specifying the meaning of the keywords in the definition:

- **Formal:** refers to the ontology's ability to be read by machines.

- **Explicit:** means that the types of definitions used and the conditions under which they can be used are well specified.

- **Shared:** reflects the idea that an ontology contains consensual wisdom. For example, knowledge that is not exclusive to a single individual but is agreed by a group.

- **Conceptualization:** refers to an abstract model of a real-world phenomenon that defines the relevant concepts for that phenomenon.

Ontologies are created to provide a machine-processable semantics of data sources that can be shared among agents (software and humans). Furthermore, ontologies can be published on the Internet and can refer to or be related to other ontologies.

An ontology uses concepts, relations, instances and axioms to represent a certain phenomenon, topic, or subject area:

- **Concepts:** they are the basic ideas that are intended to formalize. The concepts can be classes of objects, methods, plans, strategies, reasoning processes, etc.
- **Relations:** represent the interaction and link between the concepts of the domain. For example: subclass-of, part-of, exhaust-part-of, connected-to, etc.
- **Instances:** they are used to represent certain objects (individuals) of a concept.
- **Axioms:** theorems on relationships that elements of ontology must satisfy. For example: "An *Extruder* has exactly 1 *Drive System*", "A *Drive System* has minimum 1 *Motor*", and so on.

Ontologies can be classified according to their level of generality. For example, in [Gua97], ontologies are classified in:

- **Top-level ontologies.** Those ontologies describe very broad concepts such as action, event, object, matter, time, space, and so on, which are unaffected by a particular problem or domain.
- **Domain ontologies** and **task ontologies.** By specializing the concepts belonging to top-level ontologies, they describe the vocabulary related to a specific domain (such as manufacturing or agriculture) or a generic activity or task (such as prediction or visualization).
- **Application ontologies.** Those ontologies describe concepts that are dependent on both a domain and a task, and are frequently specializations of relevant ontologies. These concepts frequently refer to the roles that domain entities play while performing a specific activity, such as *replaceable unit* or *spare component*.

There are a large number of Top-level ontologies available on the Web like the Dublin Core (DC) ontology[8], a light weight RDFS vocabulary for describing generic metadata; The Friend Of A Friend (FOAF) ontology[9], used to describe people and social relationship on the Web; the Dolce+DnS Ultralite (DUL) ontology[10], an adaptation in OWL of the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE), and other domain ontologies that are focused on a particular field, for example: in medicine, the NCI Thesaurus ontology[11] developed by the US National Cancer Institute's Center for Bioinformatics and the SNOMED Clinical Terms ontology[12], developed at Department of Information and Communication Engineering of the Inha University; in energy, the EEPSA ontology[13] focused on energy eficiency and thermal comfort in tertiary buildings; and in agriculture, Agrontology[14] which provides a set of domain properties to the AGROVOC thesaurus, etc. There exist several ontologies in the manufacturing domain which are presented in section 3.

1.2.5 RDF data storage systems

RDF data storage systems are specially designed for the storage and retrieval of RDF triples through semantic queries. Those systems keep and retrieve information through a query language (like SPARQL) just like a relational system (in this case with SQL language). In addition to queries, triples can be imported/exported in Resource Description Framework (RDF) and other formats. Some RDF data storage systems were built as database engines from the ground up, while others were based on commercial relational database engines (such as SQL-based engines) or database engines that are document-oriented (No SQL). Furthermore, there are general purpose database engines with triple storage capabilities. On the DB-Engines website[15], a ranking of the most popular RDF storage systems is shown, some of them are described below.

Virtuoso[16] is a fast and scalable Multi-Model RDBMS, Linked Data Deployment,

[8] http://purl.org/dc/terms/
[9] http://xmlns.com/foaf/spec/
[10] http://www.ontologydesignpatterns.org/ont/dul/DUL.owl
[11] https://bioportal.bioontology.org/ontologies/NCIT
[12] https://bioportal.bioontology.org/ontologies/SNOMEDCT
[13] https://w3id.org/eepsa
[14] http://aims.fao.org/aos/agrontology
[15] https://db-engines.com/en/ranking/rdf+store
[16] http://vos.openlinksw.com/owiki/wiki/VOS

Data Integration Middleware, and HTTP Application Server Platform for administering data represented as tables (tabular relations) and/or RDF graphs (sentence collections). It has an open-source edition with most of the characteristics from the commercial version, however, the custom inference rules capability (using SPIN language) is missing. A 1-month evaluation license is available for the commercial edition (Virtuoso Universal Server).

RDFox[17] is an in-memory RDF triple store and semantic reasoning engine that is highly scalable. Moreover, it supports shared memory parallel reasoning. It is a cross-platform C++ software with a Java wrapper that makes it simple to integrate with any Java-based solution. It runs on Windows, Mac OS X, and Linux and, instead of using disk space, it stores data in Random Access Memory (RAM). This feature speeds up data loading compared to other RDF stores, but it comes at a cost in terms of memory usage due to poor compression. Oxford Semantic Technologies developed RDFox, and a one-month trial license can be obtained from the company's website.

Stardog[18] is a Knowledge Graph platform that combines virtualization and graph storage for cost-effective and flexible data integration. It supports a graph data model based on RDF and has the OWL2-DL Pellet reasoner embedded. Stardog has a free edition and also includes Stardog Studio, an Integrated Development Environment (IDE) for data modelers, developers and administrators to build and manage their knowledge graphs.

Neo4j[19] is a native graph database platform. It stores the relationships between data records, providing a flexible structure in contrast to traditional databases, which store data in rows, columns and tables. Neo4j performs faster queries with complex connections, and with more depth, than other databases. Although Neo4j does not support RDF natively, it offers a plugin named NeoSemantics (n10s) that enables the use of RDF and its associated vocabularies like: OWL, RDFS, SKOS and others.

In the development of this research work, different versions of the RDF storage systems mentioned above were used and compared. They were selected by taking into account the performance results shown in [Add19b] and [Add19a]: Virtuoso (version 08.03.3314), RDFox (Version 2.1.1) and Stardog (Version 7.1.1). Also, the Neo4j (Version 4.1.3) graph database was evaluated.

[17] https://docs.oxfordsemantic.tech/
[18] https://docs.stardog.com/
[19] https://neo4j.com/

1.3 Industry 4.0 powered by Semantics technologies

Semantic technologies have proven to be useful in the integration of different domains, functions, and processes. Industry is one of these domains where Semantic technologies can be used to support interoperability, collaboration and improvement of manufacturing processes. In [JSH+20], six stages in the development of a factory are presented, that reflect its degree of maturity with respect to the implementation of Industry 4.0 (figure 1.5). The first two correspond to pre-requisites (computerization and connectivity) that factories must accomplish to begin with an Industry 4.0 transformation and the other four (visibility, transparency, predictive capacity and adaptability) are part of Industry 4.0. Next, the contribution that Semantic technologies can make to each of these last four stages is described. Moreover, two more key requisites (modularity and interoperability) that benefit from the use of Semantic technologies are also presented.

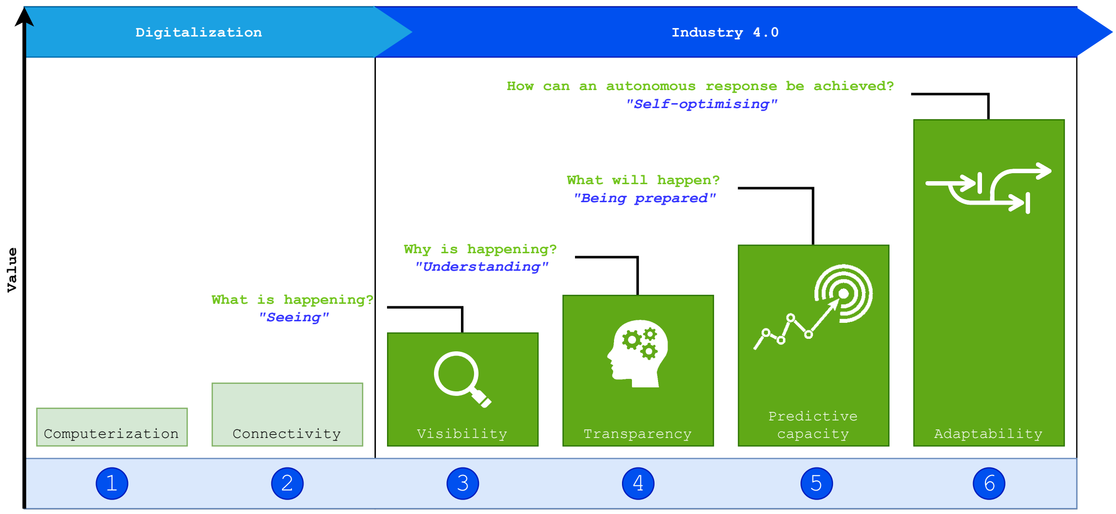


Figure 1.5: Stages in the Industry 4.0 development path [JSH+20]

Visibility refers to the fact of showing what is happening in the production chain. The design and integration of real-time dashboard systems create not only visibility but big concerns: who can visualize the data? how should be the data presented on screen? Which data is relevant for the data annalist? A domain ontology that fully describes the production chain processes (including the relevant data, the employee profiles that can access the data and the best visualization format) can help to retrieve the appropriate components enhanced with relevant information to the visualization tool, improving operational performance and monitoring.

Transparency means understanding why something is happening and using this information to improve processes. Generating this understanding requires the processing of very large volumes of data, for which it is necessary to use big data paradigms. Although Semantic technologies were not created for Big Data, they provide several advantages such as their ability to describe and integrate heterogeneous data and to infer new knowledge [TS15]. The maintenance of rule-based systems presents a big challenge when, over the years, a large set of rules has been accumulated. Having the ability to perform automated reasoning allows the development of validation schemes that help with those maintenance challenges. That automated reasoning can be achieved modeling the data and rules as ontologies using OWL 2 DL [GLDK12].

Predictive capacity refers to the company's ability to fully understand its processes and be able to foresee its behaviours in the future. This requires the integration of simulation modeling paradigms. The introduction of probabilistic algorithms of machine learning has improved the performance in the execution of simulation models, however, there is the problem of the black box, making machine decisions non-transparent and incomprehensible. The solution to that problem relies in creating *context adaptive systems* as a combination of ontologies with probabilistic machine learning [HKWT18], adding explanation to machine decisions.

Adaptability means using real-time data to make the best decision in the shortest time, which can be simple or highly complex. Unplanned events disturb manufacturing processes and require a quick rescheduling of the affected operations. However, rescheduling planning processes are complex tasks, in which the employee has to be supported. In [BLK18], an approach for an event-driven PPC (Production Planning and Control) based on a manufacturing ontology is presented. The implementation has been tested on an assembly line and validated with good results.

Modularity allows to reconfigure systems by exchanging resources and components to achieve an adaptive and responsive production. For the automation of reconfiguration decisions, it is essential to create a model that describes all the characteristics of the resources (capabilities, properties, functionalities, constraints, etc). Ontologies like MaRCO [JSHL19] supports the representation and inference of combined capabilities based on the description of simple capabilities of resources.

Interoperability refers to the ability of cooperating and sharing heterogeneous information between different systems in order to optimize their performance. By means of ontologies is possible to address the interoperability issue, describing the necessary semantics mappings to establish the correct communication between systems. Although there are many ontologies that have been developed for the industrial manufacturing domain, these ontologies are focused on specific characteristics of the manufacturing processes and turn out to be incompatible with each other. To solve this problem, the Industrial Ontology Foundry (IOF)[20] has been formed to create a set of reference ontologies that span the entire domain of industrial manufacturing. In this way, promote interoperability between the different disciplines of manufacturing, engineering and supply chain.

1.4 Contributions of this research work

This research work is aimed at promoting the use of Semantic technologies in the Industry 4.0 environment. For that purpose it provides on the one hand, artifacts that allow adding value to the processes present in smart manufacturing, and on the other hand, it offers a roadmap to facilitate the implementation of Industry 4.0 in SMEs where, for reasons of various kinds (economic, cultural, technological, etc.), there are barriers that prevent this industrial evolution. To this end, each of the contributions of this research work is explained below.

- *C.1* **An ontology for describing a type of manufacturing machine for Industry 4.0 systems (an extruder).** This ontology provides sound descriptions over: main components, spatial connections, main features, 3D representations and the great variety of sensors that belong to this type of manufacturing machine.
- *C.2* **A Visual Query System based on semantics descriptions.** This system provides a 2D digital representation of a manufacturing machine and dynamically customized forms to formulate queries. In this way, domain experts can gain value and insights out of the captured data as rapidly as possible, minimizing the need to contact Information Technology experts.
- *C.3* **A methodology for the implementation of Industry 4.0 in SMEs.** Oriented to the customer's life cycle and enhanced by the use of Semantic technologies

[20] https://www.industrialontologies.org/

and 3D digital technology, this methodology can help software engineers of SME manufacturing scenarios to achieve a successful transformation towards Industry 4.0. Moreover, the methodology has been applied step by step in a real manufacturing enterprise, showing how to generate a series of services that positively affect the relationship with the customer in two of the three phases of the customer life cycle.

2. Research method

For the development of the contributions presented in this research work, it was necessary to follow a research method that would allow to obtain a better understanding of the context in which the research work is supported (Industry 4.0 and semantic technologies) and identify relevant previous works that leave open the possibility of improvement and, therefore, the generation of important contributions. Consequently, the research method of this work is designed combining the key element of two research methodologies with solid conceptualizations: *Design Science Research* [HMPR04, Hev07] and *Case Study Research* [Eis89].

Design Science Research (DSR) is a rigorous research approach that proposes the construction of artifacts to provide a useful and effective solution to a problem in a given domain. The artifact must be an innovative solution to a non-trivial problem. The development of the artifact involves a cycle of design-build-evaluation activities, which iterate as many times as necessary before the artifact is finally validated and communicated for use.

Case Study Research (CSR) is based on a descriptive inquiry, usually with experimental material collected over a period of time from a well-determined case to offer a study of the environment and related processes in the phenomenon. In [Yin94], a case study is defined as an experimental investigation activity that, through the use of versatile experimental material collected in many different ways, reviews a specific event or action of today in a limited area. The purpose of the case analysis is to make an intense inquiry about a specific case, such as a subject, a group, an institute or a society.

The integration and application of these methodologies in the research work has allowed to:

Frame the context where this research work has been focused. On the one hand, large and solid enterprises that have already made their transition to Smart Manufacturing (contributions *C.2* and *C.1*) and, on the other hand, Small and Medium-sized Enterprises (SME) with the intention of making their transition but that encounter different barriers that do not allow it (contribution *C.3*).

Interact with relevant actors belonging to the defined context, generating synergies to identify and overcome the problems, challenges and opportunities that arise. Domain experts, R&D directors and IBDS (Industrial Big Data Services) provider executives have been consulted for the development of the contributions in this research work.

Identify drawbacks that can be solved with the use of Semantic technologies. It is the case of the limited data analysis that domain experts can do due to the few information obtained from predefined dashboards (contribution *C.2*); the lack of sound descriptions of manufacturing machines that happen to be accessible, interoperable, and reusable (contribution *C.1*); the lack of a well defined methodology to SMEs in which the transition to Industry 4.0 can be carried out using minimal economic resources and taking advantage of technologies that were not previously accessible (contribution *C.3*).

Review the existing knowledge base to identify the related literature and previous contributions that lead to opportunities for improvement. The compilation of antecedents, previous investigations and theoretical considerations on which this research work is based are exposed in section 3.

Develop the necessary artifacts to solve the identified drawbacks. For the purpose of this research, four artifacts were created: ① an ontology to describe a type of manufacturing machine (contribution *C.1*); ② a VQS system supported by semantics (contribution *C.2*); ③ a methodology to ease the process of implementing Industry 4.0 in SMEs (Contribution *C.3*); and ④ a series of services oriented to the customer life cycle and developed following the methodology (Contribution *C.3*).

Define the case study where the artifacts were tested to validate their effectiveness. The contributions have been materialized for a real smart manufacturing scenario (in particular, a plastic bottle production factory that follows an extrusion process) in different scopes (i.e., production line, supply chain, data analysis, customer management) and have been tested by domain experts.

3. Antecedents and previous work

In this section, different previous works and referenced sources in the context where this research work takes place are addressed. This includes the semantic representation of machines and processes, the visual exploration of data, and the different initiatives existing within SME and the Industry 4.0 environment.

Related to the semantic representation of machines and processes, several ontologies related to the Smart Manufacturing area can be found. Those ontologies were defined with distinct purposes and, therefore, describe different types of information related to that area. For example, the PSL (Process Specification Language) ontology [Grü09] includes fundamental concepts for representing manufacturing processes. The foundational elements of the core of the PSL ontology are four primitive classes (*activity*, *activity-occurrence*, *timepoint*, *object*), three primitive relations (*participates-in*, *before*, *occurrence-of*) and two primitive functions (*beginof*, *endof*). The MASON (Manufacturing's Semantics Ontology) ontology [LSDS06] is an upper ontology for representing what authors consider the core concepts of the manufacturing domain: products, processes and resources. As a result, the main classes of MASON are *Entity* (for specifying the product), *Operation* (for describing all processes linked to manufacturing) and *Resource* (for representing concepts regarding machine-tools, tools, human resources and geographic resources). The SIMPM (Semantically Integrated Manufacturing Planning Model) ontology [SS19] is an upper ontology that models the fundamental constraints of manufacturing process planning: manufacturing activities and resources, time and aggregation. MaRCO (Manufacturing Resource Capability Ontology) [JSHL19] defines capabilities of manufacturing resources.

Its main class is *Capability*, which is specialized to cover both, simple capabilities (e.g. *Fixturing*, *SpinningTool*) and combined capabilities (those that require a combination of two or more simple capabilities, e.g. *PickAndPlace*, which requires *FingerGrasping* or *Vacuum Grasping*, *Moving* and *Releasing*). The MSDL (Manufacturing Service Description Language) ontology [AD06] allows to describe manufacturing services. More precisely, a *Manufacturing Service* is seen as a *Service* that is provided by a *Supplier* and that has some *Manufacturing Capability*, which is enabled by some *Manufacturing Resource* and delivered by some *Manufacturing Process*. The P-PSO (Politecnico di MilanoProduction Systems) ontology [GF12] considers three aspects in the manufacturing domain: the physical aspect (the material definition of the system), the technological aspect (the operational view of the system) and the control aspect (the management activities), for information exchange, design, control, simulation and other applications. Thus, its main classes are *component*, *operation* and *controller*, which model the aforementioned three aspects, as well as *part*, *operator* and *subsystem*. OntoSTEP (ONTOlogy of Standard for the Exchange of Product model data) [BKS+12] allows the description of product information mainly related to geometry. MCCO (Manufacturing Core Concepts Ontology) [UYC+11] focuses on interoperability across the production and design domains of product lifecycle. It provides some core classes in categories such as *ManufacturingProcess*, *ManufacturingFacility*, *ManufacturingResource* and *Feature*. Finally, SAREF4INMA [ETS17] pursues favouring interoperability with industry standards. Some of its main classes are *ProductionEquipment*, *Factory*, *Item* and *MaterialCategory*. In this thesis, an ontology to describe a type of manufacturing machine is introduced. Unlike other manufacturing ontologies, this ontology depicts a manufacturing machine (specifically an extruder) in fine-grained detail, with descriptions for main components, spatial relations, features, 3D representations, and sensors.

In terms of visual data exploration, Key Performance Indicator (KPI) dashboards have been widely used in a variety of industries, including manufacturing. For instance, [RLS20] shows a digital control room with multitouch and multiuser-based annotation dashboards for analyzing manufacturing data. The issues and outliers that occur during the manufacturing process are visualized and identified using a calendar view in [CCL+16]. A machine learning-based technique is used in [ILM+20] to assist real-time visualization of data.

However, visualization tools must also support a customization functionality of various user-defined discovery scenarios and preferences based on research requirements [BPP19]. Visual Query Systems has a proven track record in this field. They have been used for querying databases [CCLB97], for retrieving data from the Web [Llo17] and also for visual exploration of time series. In this last case, there are approaches that advocate for the use of example-based methods such as [EF13], and [CPS+19], which proposes a multilevel map-based visualizations of geolocated time series.

Different proposals can also be found among systems that deal with semantic data, such as SparqlFilterFlow [HLBE14], which employs a diagram-based approach to represent the queries, and Rhizomer [BGA13], which employs a form based approach. Moreover, OptiqueVQS [SKZ+18] is a semantic-based visual query system that exploits ontology projection techniques to enable graph-based navigation over an ontology during query construction and sampled data to enhance selection of data values for some data attributes. It shows all the classes defined in the loaded domain ontology to the users as a starting point for queries formulation. This forces domain experts to gather experience in the ontology before using the system. Other works such as [AV18] focus on tools such as OWLViz4 or LODLive [CMA12] for visualizing the content of ontologies and knowledge bases in the form of a graph but not for querying them, or on tools that allow querying but require a good knowledge of the underlying ontology (e.g., [LNHE16]). This thesis proposes a semantic-based visual query system based on two main considerations: the lack of visual query systems for smart manufacturing scenarios and the desire to incorporate semantics techniques into those scenarios.

Finally, regarding the implementation of Industry 4.0, there exists several technologies that contribute to this end, such as Internet of Things (IoT), Cloud Computing, Cybersecurity, Big Data and Analytics, Additive Manufacturing, Simulation, and Robotics. In the specialized literature, several projects and case studies consider the incorporation of Industry 4.0 technologies interesting in the context of manufacturing SMEs. For example, project ESMERA (European SMEs Robotic Applications [IWZ+18]) of the European Commission's Horizon 2020 Research and Innovation Programme aims to boost robotics innovation for European SMEs by funding projects such as REFLECT [CAS20], which tackles the assembly of deformable parts in dishwashers by using a robotic system. Also under the Horizon 2020 Programme, project CloudiFacturing [Clo20] aims to optimize production processes and producibility in

SMEs using Cloud/HPC-based modeling and simulation. More precisely, it supports projects such as 3D-CPAM (3D Clothing Production by Additive Manufacturing [TDS+20]), which uses advanced HPC/Cloud services and modern 3D printing technologies to optimize the 3D fashion design manufacturing process, or D2Twin [D2L20], which uses big data analytics to improve quality control and maintenance. This thesis outlines a methodology for implementing Industry 4.0 in SMEs. The methodology aims to develop a set of services that companies can use throughout the customer life cycle to build stronger, longer-lasting relationships with customers. These services are supported by the use of semantic and 3D digital technologies.

Regarding 3D digital technologies, Virtual Reality (VR), Augmented Reality (AR) and Mixed Reality (MR) are considered relevant technologies for the new generation of intelligent manufacturing [ZLLY19]. Virtual reality is a high-end human-computer interface that allows interaction with simulated environments in real time and through multiple sensorial channels [LSS19]. The users believe to be inside a reality that does not exist in truth, but they act like in the real world [Sla09]. Augmented Reality has been defined as a system which supplements the real world with virtual objects (computer generated) that appear to coexist in the same space as the real world [BV19, ME19]. It provides benefits especially in designing products and production systems. While VR requires inhabiting an entirely virtual environment, AR uses existing natural environment and overlays virtual information on top of it. Finally, Mixed Reality like augmented reality, places digital or virtual objects in the real world. However, with mixed reality, users can quickly and easily interact with those digital objects to enhance their experience of reality or improve efficiency with certain tasks [GFPC+17]. These technologies can been used for several purposes in the industrial and manufacturing environment, for example in the process of product design [MZV18, GZC+18], for assembly simulations [TLL21, AAAAD16], for training purposes [ORGS15, TLL+19], for factory layout planning [GBB+19, HRR+18] or for improving maintenance services [RSP+21, Aba20]. Moreover, since 3D modeling has shown a realistic description of manufactured products by generating high-quality textures and proper lighting, it can be used for showcasing purposes. In this sense, it can be seen how some companies such as Schneider Electric[1] or Reid Supply[2] show their technical products in a interactive 3D product catalogue.

[1] https://sketchfab.com/blogs/enterprise/customer-stories/electronics/3d-viewer-schneider-electric

[2] https://www.reidsupply.com/

4. Hypothesis and objectives

Once some concepts about Industry 4.0 and Semantic technologies related to the work carried out in this thesis have been exposed, an overview of the contributions has been shown and previous works have been reviewed, the hypothesis and the objectives on which this research work is based are presented. Each of them is detailed below.

4.1 Hypothesis and general objective

The hypothesis of this research work has been defined as:

The use of Semantic technologies helps to generate value in the Smart Manufacturing processes where they are applied.

The confirmation of this hypothesis has been based on the achievement of the following general objective:

Creation of a series of artifacts that demonstrate the positive influence of the use of Semantic technologies when applied to a real Smart Manufacturing scenario (defined in section 1.1.3)

It should be mentioned that the previous objective may be too broad since it covers all possible applications within Smart Manufacturing. Hence, it is necessary to define specific objectives, which have been considered in the research work.

4.2 Specific objectives

The specific objectives have been defined taking into account three selected topics: enriched descriptions of Smart Manufacturing components, data visualization and analysis, and implementation of Industry 4.0 in SMEs. These objectives are specified below.

1. Create an ontology that provides sound, accessible, interoperable, and reusable descriptions of a type of manufacturing machine (addressed in publication 7.1).
2. Create a semantic-based artifact that enables experts to visualize and analyze data. The artifact must include rich annotations that aid in the identification of patterns and insights, resulting in a better understanding of the data and facilitating the analysis of the data. (addressed in publication 7.2).
3. Create a methodology that provides a guide for the creation of services oriented to the customer life cycle for the implementation of Industry 4.0 in Small and Medium-sized Enterprises (SMEs) using Semantic and 3D technologies (addressed in publication 7.3).
4. Create a proof-of-concept artifact following the steps of the proposed methodology to validate its effectiveness (addressed in publication 7.3).

With the achievement of these specific objectives, it is possible to confirm that the established hypothesis is valid and, therefore, that Semantic technologies can be used in the Smart Manufacturing environment to improve data analysis and positively affect the customer life cycle.

5. Research work summary

This section outlines the tasks that were completed in order to achieve each of the above-mentioned specific objectives. Furthermore, the tests carried out and the results obtained as part of the creation process of the artifacts are also specified.

5.1 An ontology to describe a type of manufacturing machine that performs an extrusion process

As mentioned above, this research work is based on a real manufacturing scenario. In this scenario, the data extracted from the sensors embedded in the manufacturing machines are transferred to an ITS provider for storage, visualization and analysis. However, the data obtained from a sensor are presented in a disconnected and independent way, since it comes in timestamp-value tuples, preventing important information from being used for knowledge extraction, such as: type of sensor (e.g., temperature, pressure), units of the observation value (e.g., Fahrenheit, Kelvin), optimal sensor operating ranges, etc. As a first approximation, an ontology was created in which the sensors and observations that were produced from an extruder were modeled. This ontology reuses concepts from the SOSA/SSN ontology [HJC+18] and adds a specialization on the types of sensors and observations, allowing in this way query formulation and visualization of enriched data. Nevertheless, the sensors were expressed in the ontology as independent elements belonging to a simple individual (i.e., *Machine*), lacking the corresponding descriptions that identify this type of manufacturing machine: type and quantity of components, characteristics of these components, spatial relations between components, location of the sensors along the machine, etc. Ontologies such as MaRCO, present a specialization of various machines used in manufacturing processes (see figure 5.1). However, to the best of our knowledge, there is no an ontology for describing specific industrial machine types with a fine-grained detail, and more particularly, extruder machines. For this reason, it was decided to create an ontology to compensate for this deficiency.

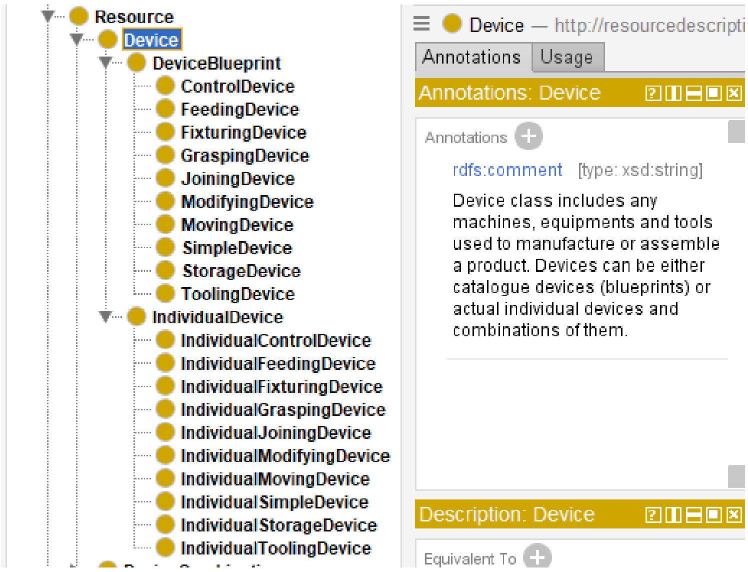


Figure 5.1: machine description in MaRCO ontology

Different methodologies were reviewed for the development of the ontology, with the NeOn methodology being chosen due to the variety of scenarios it considers and the detailed description of the activities to be followed. Following the NeOn methodology, the Ontology Requirements Specification Document (ORSD) was defined including the competency questions that the ontology must answer, which where classified in five different dimensions: the components of an extruder, the spatial connections between those components, their features, their 3D description and the sensors that capture information about several indicators.

Existing ontological resources were consulted for each of these dimensions in order to favor reuse, and if these ontological resources could not be found, non-ontological resources such as existing literature were consulted. In this way, the description of the components of an extruder were extracted mainly from a non-ontological resource [GMW04] which allocated a complete chapter for this purpose. For the spatial connections between the components, the GeoSPARQL ontology [PH12] was used, which contains the descriptions of the Region Connection Calculus [CBGG97, RCC92], specifically the RCC8 relationships. To describe the characteristics of the components, a research work was taken into account that evaluated different ontologies of measurements [KS18], among which the QUDT [HK11] and OM [RvAT13] ontologies stand out. Finally it was decided to use an excerpt of the OM ontology since the QUDT had many errors and fewer concept descriptions than OM. Related to the 3D representation of the components, the 3D Modeling Ontology (3DMO) [Sik17] was used because it maps the entire XSD-based vocabulary of the industry standard X3D[1] to OWL 2. Finally, to describe the sensors that capture information about the indicators, the ontology created initially as a first approximation was used, which reuses concepts from the SOSA/SSN ontology.

Each of the aforementioned dimensions was expressed as an independent module in a main ontology called *ExtruOnt* (see figure 5.2), which provides the necessary interconnections between these modules. Other types of manufacturing machines can be described using this modular configuration by simply replacing the module containing the extruder's own specifications (i.e., components) with a module containing the descriptions inherent to the new type of machine and modifying the connections with other modules in the main ontology.

The evaluation of the ontology was carried out with the help of three experts: A R&D director of a company that design and produce extruder machines, a director of an IBDS (Industrial Big Data Services) Provider company and an expert in developing and managing ontologies who works in a technology center specialized in the industrial domain. Two approaches in the evaluation were considered: *Domain coverage* and *Quality of the modeling*.

Regarding *Domain coverage*, the experts suggested adding alignments with upper ontologies such as DUL[2] and MASON [LSDS06]. In the final version of the ExtruOnt ontology, 95% of the related vocabulary extracted from ontological and non-ontological resources was included, the remaining 5% corresponds to terms out of the ontology scope or without significant value (e.g., parts of obsolete extruder models). For *Quality of the modeling*, the ontology metrics were extracted and compared with the metrics from other well-known manufacturing domain ontologies. Moreover, the ontology was evaluated using the Ontology Pitfall scanner (OOPS!) [PVGPSF14]. The detected flaws were corrected, however, two minor pitfalls belonging to external on-

[1] http://www.web3d.org/what-x3d-graphics

[2] http://ontologydesignpatterns.org/wiki/Ontology:DOLCE+DnS_Ultralite

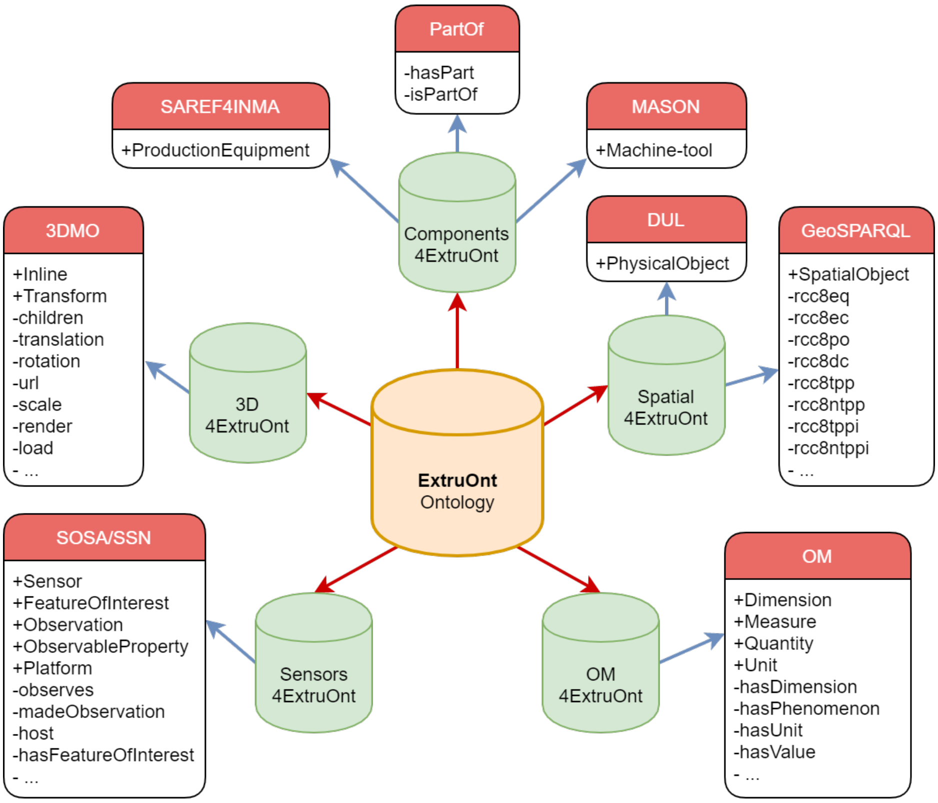


Figure 5.2: *ExtruOnt* ontology diagram (from publication 7.1)

tology imports remain.

5.2 A semantic-based artifact for data visualization and analysis

For the development of this specific objective, the creation of a Visual Query System (VQS) based on semantics has been chosen, since this type of system offers high customization capabilities. Visual Query Systems have been used to extract information from the web, consult databases, and perform visual exploration of time series. However, its application in the industrial environment is limited, and finding semantic-based VQS systems is even rarer.

The developed system is supported by the *ExtruOnt* ontology, specifically by the sensors module. The data extracted from the sensors installed in the extruders are

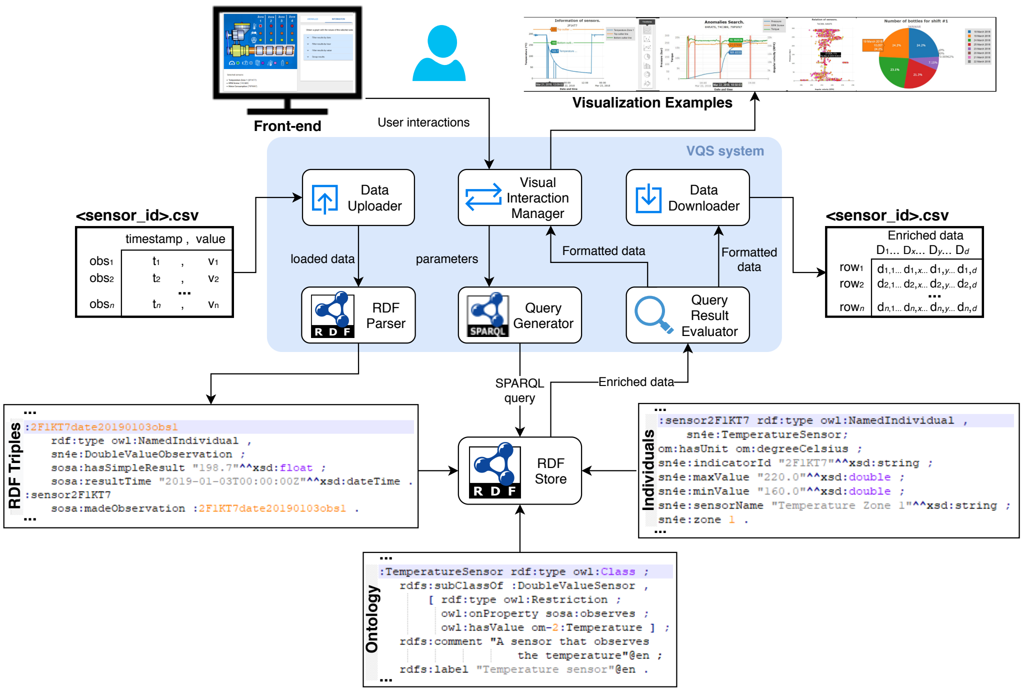


Figure 5.3: VQS system overview (from publication 7.2)

annotated using the terms included in the ontology and stored in an RDF store (see figure 5.3). For user interaction with the system, a 2D representation of the extruder was designed where the different sensors belonging to an extruder were obtained from the existing descriptions in the RDF store and displayed dynamically. In this way different types of extruders with different sensor configurations can be visualized. Through the 2D representation of the extruder, those sensors on which the user wishes to obtain data can easily be selected. Although the final ontology supports the three-dimensional description of the components of an extruder, at the time of developing the VQS this information was not available.

For data exploration, three types of queries were designed in collaboration with domain experts: information queries, relation queries and anomalies queries. Each of these queries has an individual form that is dynamically created depending on the annotated descriptions of the selected sensors. For example, aggregation functions are only available for those sensors that record numerical values such as temperature sensors. Information queries are the simplest queries, in which the data related to one or more sensors are obtained, limited by the different constrains selected (e.g., date, time, aggregation functions). Relation queries are used to ask for the observations made by some specific sensors when certain values hold in the observations made at the same timestamp by some other sensors. Anomalies queries indicate certain correlations between the values of different sensors that are supposed to hold under normal conditions. The system allows users to run customized or predefined anomalies queries. Once the form corresponding to the selected type of query is filled, a SPARQL query is generated and executed against the stored data. In addition, a module was included for the download of enriched data with which it is possible to carry out additional analysis of the data using external tools (e.g., Tableau, PowerBI).

A visualization module has been developed and imported into the sensors module of the ontology, where several recommendations for data visualization have been described, in order to select the most appropriate representation depending on the nature of the query and the sensors involved. This visualization module uses terms from the Semanticscience Integrated Ontology[3] (SIO) to describe graph and chart types (e.g., pie chart, bar graph, scatter plot, line graph).

Finally, in the context of the case study, an empirical evaluation was conducted from the perspectives of usability and behavior. For the quantitative usability evaluation of the system,users were asked about how intuitive was the system for performing the queries that they wanted. They provided positive responses with regard to the intuitiveness of the system for performing queries. They also commented on the speed and suitability of the visualization of the results. The system was found to be very intuitive and easy to use with a high degree of customization.

For the evaluation of the behavior, three different RDF storage systems where selected: Virtuoso (version 08.03.3314), Stardog (Version 7.1.1) and RDFox (Version 2.1.1). Also, the Neo4j (Version 4.1.3) graph database engine was considered. They were tested on the same hardware specifications over two metrics: data storage space and query response time. For data storage space, when compared to Stardog and RDFox, Virtuoso provides better space management, reducing the amount of space required to store the series by 37.99% and 78.4%, respectively. Related to query response time, Virtuoso and RDFox were the fastest for information and relation queries, respectively. Anomalies queries need to use the rule-based inference capabilities of RDF storage systems. In this sense, RDFox shows a remarkable performance as it materializes these inferences. Nevertheless, Virtuoso shows a noteworthy performance

[3] https://bioportal.bioontology.org/ontologies/SIO

even using the query rewriting method. A complete description of the data storage space and query response time evaluations, including the amount of data and query types used, is detailed in publication 7.2.

5.3 A methodology for the implementation of Industry 4.0 in SMEs

When it comes to making the transition to Industry 4.0, Small and Medium-sized Enterprises (SMEs) run into a variety of problems. This type of enterprises is hesitant to adopt this change due to economic, technical, and cultural barriers, among other factors. One of the objectives of this research work is to give these companies a structured guide in the form of a methodology so that they can achieve closer and longer-lasting customer relationships through the implementation of new services aligned with the four industrial revolutions (Industry 4.0). The proposed methodology takes into account the phases of a typical information system's life cycle, but what sets it apart are the detailed instructions it provides for incorporating semantic descriptions and 3D visualization. Moreover, in order to provide a thorough explanation of the methodology, it is divided into stages, which are then divided into phases, which in turn are made up of activities (see figure 5.4).

The methodology consists of six main stages:

- **Definition of objectives and goals.** The aim of this stage is to define a roadmap for the introduction of new services based on the company's interest in implementing Industry 4.0 to improve customer relationships.
- **Build of semantic descriptions.** In this stage, the activities concerning the search and reuse of ontological resources are explained. Activities are also addressed for the creation of ontologies (if there is no suitable one) and for the selection and evaluation of the knowledge platforms where the information will be stored.
- **Build of the 3D visualization.** This stage deals with the activities concerning the search or creation of the 3D representations of the products, as well as the evaluation and selection of the most suitable rendering interface.

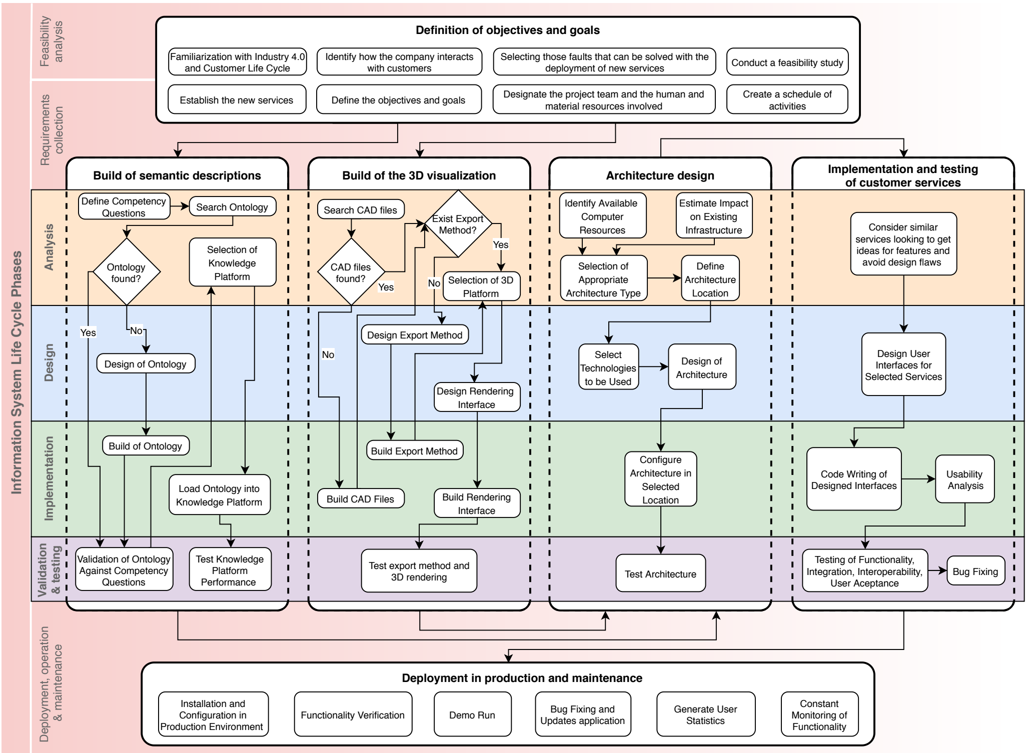


Figure 5.4: Methodology diagram (from publication 7.3)

- **Architecture design**. The activities related to the definition of an architecture to support the semantic descriptions and the 3D rendering interface are presented. The definition of the architecture takes into account a number of factors, including the identification of available information resources, the effect on existing infrastructure, and the final location of selected architecture (own datacenter or in the cloud).

- **Implementation and testing of customer services** The implementation and testing of the services defined in the *Definition of objectives and goals* stage are carried out in this stage. These services must be targeted at ensuring an improvement in the consumer life cycle by incorporating the technologies chosen in previous stages.

- **Deployment in production and maintenance**. In this last stage, the activities about the deployment in production of software artifacts are described, such as the verification of the functionality, the correction of bugs, the periodic generation of statistics and the monitoring and maintenance tasks.

Additionally, the expected inputs and outputs are specified for each stage. They represent the necessary prerequisites to achieve the desired result by carrying out the activities outlined in each stage of the proposed methodology.

5.4 A proof-of-concept artifact to validate methodology effectiveness

For the evaluation of the methodology effectiveness, a case study was carried out in a real manufacturing company where the detected faults related to the customer's life cycle were addressed applying the proposed methodology.

In the first stage (*Definition of objectives and goals*), an analysis of the customer relationship with the company was done. In this, it was observed that the company's website offered limited information about products and, due to a lack of context about the case when requesting help, remote assistance had delays. Taking into account those faults, it was decided to improve the customer experience in two phases of the customer life cycle (i.e., Discover & Shop and Use & Service) through the creation of three services: catalogue, searching module and virtual technician. Furthermore, a feasibility study was conducted and the activity planning was done.

The definition of the competency questions was made in the second stage (*Build of semantic descriptions*). The competency questions were intended to describe the main products of the company (i.e., extruders) and represent the questions that customers can ask about those products when using the new services. Next, The ExtruOnt ontology was assessed in relation to the competency questions, making appropriate modifications and additions to correctly answer all questions. It then proceeded to select the knowledge platform where the ontology and descriptions corresponding to the individuals to be modeled would be stored after the ontology had been selected and validated. Three RDF storage systems were considered for this: Virtuoso, Stardog, and RDFox, with Virtuoso winning out due to its quick response times and the possibility of an open-source version. Finally, Virtuoso was deployed in a Google Cloud virtual machine, where its correct operation was verified.

The company's ability to create 3D models of its products via CAD application was verified in the third stage (*Build of the 3D visualization*). However, the CAD application's license was very expensive, and the free update period had expired. It was then decided to migrate to a cloud-based CAD application for creating, modifying, and storing product 3D models. Among the options considered, Onshape was chosen because it offered a free version (ideal for this proof of concept) and an API for model extraction. The next step was to choose a rendering framework for the OnShape models that had been extracted. Three.js was chosen from among the evaluated frameworks because it had a short learning curve and a large and well-structured library. Finally, a test application was used to conduct communication tests between the OnShape API and the rendering framework.

The available hardware resources were examined in the fourth stage (*Architecture design*). Because production resources are outsourced, the decision was made to use a development environment based on Google Cloud that the company already had. The 3-Tier architecture was chosen due to the project's tier separation requirements. FireBase, as a hosting service; React, for the presentation layer; and NodeJs and ExpressJs, for the application layer, are among the frameworks and libraries used in the architecture.

In the fifth stage (*Implementation of customer services*), three services were implemented that guarantee an improvement in the customer life cycle and integrate the technologies chosen in the previous stages. A catalog, a search module, and a virtual technician are included. The first two services are related to the Discover & Shop phase of the customer life cycle, while the last one is related to the Use & Service phase. Before moving forward with the design of the services, the methodology required an analysis of similar services. The results of this activity provided important guidelines on how to approach new services, taking into account the characteristics, benefits, and drawbacks of the service types examined. The detailed description of the created services is exposed in publication 7.3. The company is currently conducting usability analysis and testing of functionality, integration, interoperability, and user acceptance at the time of writing this dissertation. As a result, once the fifth stage is completed, the activities corresponding to the sixth stage (*Deployment in production and maintenance*) will be carried out.

Part II

Conclusions

6. Conclusions

The development of new technologies such as the Internet of Things (IoT), ubiquitous systems, Big Data, and Artificial Intelligence among others, has led to a new industrial revolution where the barriers between the physical and the digital have been reduced considerably. This new revolution, named Industry 4.0, has allowed the generation of a new economy based on data and the implementation of Smart Manufacturing, where the data obtained through the sensors installed in the manufacturing machines can be transmitted to large data centers where they are treated and analyzed in order to extract knowledge to predict and improve all aspects of the production chain. Numerous contributions based on these new technologies have been produced, aimed at improving the analysis of data in Smart Manufacturing. However, the use of Semantic technologies has not been promoted as strongly as other technologies such as Artificial Intelligence and Big Data, leaving a wide spectrum to be explored. On the other hand, the implementation of Industry 4.0 has been elusive for those Small and Medium-sized enterprises (SMEs) where, due to various economic, social, technical obstacles, etc., there exist great challenges for its implementation.

Through the contributions presented in this research work, the use of Semantic technologies in the field of Smart Manufacturing is promoted, taking into account aspects such as the generation, visualization and analysis of enriched data. In addition, the problem of the implementation of Industry 4.0 in SMEs is also addressed, but focusing on the customer life cycle, since most of the contributions found in the literature are product-oriented, leaving this area aside. The conclusions regarding the contributions of this research work are presented below.

6.1 ExtruOnt: An ontology for describing a type of manufacturing machine for Industry 4.0 systems (publication 7.1)

The aim of this contribution is to introduce the *ExtruOnt* ontology, which includes terminology for describing a type of manufacturing machine used for extrusion processes (extruder). It is made up of five modules: *components4ExtruOnt* for representing the components of an extruder, *spatial4ExtruOnt* for representing spatial relationships among those components, *OM4ExtruOnt* for representing the features of those components, *3D4ExtruOnt* for representing 3D models of the components, and *sensors4ExtruOnt* to reflect data collected by sensors. While the *ExtruOnt* ontology is focused to describe extruders, it has been designed in such a way that it can be customized to describe other forms of manufacturing machines.

The *ExtruOnt* ontology's descriptions will allow various types of users to become more acquainted with the extrusion process, to easily collaborate with other manufacturing companies, to create personalized 3D images of extruder machines, and to assist in the exploration of data collected by sensors.

The *ExtruOnt* ontology has been documented and made publicly accessible. It was evaluated by humans and software artifacts based on two assessment goals: domain scope and modeling accuracy. The evaluation demonstrates that *ExtruOnt* can answer the specified competency questions, meeting the proposed specifications and proving that its modules are correctly designed. It is also compatible with similar ontologies, allowing for easier interoperability.

6.2 A Semantic Approach for Big Data Exploration in Industry 4.0 (publication 7.2)

A problem that is being considered in manufacturing scenarios is the creation of software tools that support customization capabilities that enable data discovery and visualization for different users based on their research needs. Furthermore, the exploration and visualization of recorded time-series data leads to a better understanding of the indicators used in system monitoring. A semantic-based visual query system is presented in this contribution, which allows domain experts to formulate queries involving dynamically generated forms and a customized digital representation of the manufacturing machine. The system also allows users to export data that has been enriched. Furthermore, depending on the nature of the data, it offers a customized visualization. The entire process is assisted by the *ExtruOnt* ontology that describes the machine's key components and sensors.

Despite the fact that disk space is an important consideration in big data scenarios, the advantages of semantic data annotation for data analysis are not comparable to the limited information derived directly from raw data. As a result, the additional storage required as a result of semantic annotation is tolerable, and the query response times obtained are manageable.

6.3 Towards the implementation of Industry 4.0: a methodology based approach oriented to the customer life cycle (publication 7.3)

In this contribution, a novel methodology centered on the customer life cycle to help Small and Medium-sized Enterprises in conventional manufacturing sectors adopt Industry 4.0, despite their limited resources and the high level of uncertainty that this entails, is presented.

This methodology promotes the use of Semantic technologies and 3D visualization, both of which have been thoroughly researched separately but, to our knowledge, have not been combined. On the one hand, Semantic technologies, such as ontologies, allow for a great deal of versatility in the representation of information, as well as inference and reasoning capabilities that conventional databases struggle to achieve. 3D modeling technologies, on the other hand, offer a more enhanced visual representation with better graphics and navigation controls, enabling the user to have a more immersive and improved experience. All of these benefits aimed at the customer, can strengthen the relationship between the customer and the company over their life cycle, resulting in a high level of loyalty.

An additional contribution is a system that was implemented as a proof of concept in a real manufacturing organization, and uses the previously mentioned methodology to produce a set of services that positively impact the customer relationship in two of the three phases of the customer life cycle. This system is based on the *ExtruOnt* ontology that allows for the accurate and versatile description of extrusion machines.

6.4 Future work

With regard to future work, we have identified two main research directions: ① Introduction of improvements in the system created in the proof of concept, and ② Exploration of the benefits of using Semantic technologies in the learning process of probabilistic models.

Concerning the system created using the proposed methodology, it is necessary to look at the improvements that can be made in the second phase of the customer life cycle, Buy & Install. The objective pursued is to extend the number of provided services by including new channel partner services (for dealers or distributors), such as visualizing and analyzing real-time data that can help them to make better operational decisions.

Regarding the benefits of using Semantic technologies in the area of Machine Learning, within the scope of this thesis, a research line is open on the one hand, in the explicability problem by extracting the rules of Machine Learning models (e.g., Decision Tree, Random Forest) and replacing them with SWRL semantic rules that allow describing the explanation of the results. On the other hand, related to the improvement of probabilistic models for time-series multi-class classification, the work progress is oriented to use additional Semantic information extracted from time series. The insertion of this information in the layers of Deep Learning models (Convolutional Neural Networks that only use the values of the observations) has shown a significant improvement (improved F1 score).

Part III

Publications

7. Publications

7.1 ExtruOnt: An ontology for describing a type of manufacturing machine for Industry 4.0 systems

Title: ExtruOnt: An ontology for describing a type of manufacturing machine for Industry 4.0 systems

Authors: Víctor Julio Ramírez-Durán, Idoia Berges, Arantza Illarramendi

Journal: Semantic Web

Impact factor (2019): 3.182

Rank in category COMPUTER SCIENCE, THEORY & METHODS: 22/108 (Q1)

Publisher: IOS Press

Year: 2020

DOI: http://dx.doi.org/10.3233/SW-200376

Semantic Web 11 (2020) 887–909
DOI 10.3233/SW-200376
IOS Press
887

ExtruOnt: An ontology for describing a type of manufacturing machine for Industry 4.0 systems

Víctor Julio Ramírez-Durán *, Idoia Berges and Arantza Illarramendi
[a] *Department of Languages and Information Systems, University of the Basque Country UPV/EHU, Manuel Lardizabal Ibilbidea, 1, 20018 Donostia-San Sebastián, Spain*
E-mails: victorjulio.ramirez@ehu.eus, idoia.berges@ehu.eus, a.illarramendi@ehu.eus

Editors: Dhaval Thakker, University of Bradford, UK; Pankesh Patel, Fraunhofer USA – CESE, USA; Muhammad Intizar Ali, Insight Centre for Data Analytics, National University of Ireland, Ireland; Tejal Shah, Newcastle University, UK
Solicited reviews: Julius Mboli, University of Bradford, UK; Marwan Al-Tawil, King Abdullah II School of Information Technology, The University of Jordan, Jordan; two anonymous reviewer

Abstract. Semantically rich descriptions of manufacturing machines, offered in a machine-interpretable code, can provide interesting benefits in Industry 4.0 scenarios. However, the lack of that type of descriptions is evident. In this paper we present the development effort made to build an ontology, called *ExtruOnt*, for describing a type of manufacturing machine, more precisely, a type that performs an extrusion process (extruder). Although the scope of the ontology is restricted to a concrete domain, it could be used as a model for the development of other ontologies for describing manufacturing machines in Industry 4.0 scenarios.

The terms of the *ExtruOnt* ontology provide different types of information related with an extruder, which are reflected in distinct modules that constitute the ontology. Thus, it contains classes and properties for expressing descriptions about components of an extruder, spatial connections, features, and 3D representations of those components, and finally the sensors used to capture indicators about the performance of this type of machine. The ontology development process has been carried out in close collaboration with domain experts.

Keywords: Ontology, extruder, Industry 4.0, Smart Manufacturing

1. Introduction

Different initiatives and strategies are emerging in the 4th Industrial revolution (Industry 4.0) that is currently being experienced in the manufacturing sector. Mainly they address, on the one hand, the compilation of manufacturing records of products, with data about their history, state, quality and characteristics, and on the other hand, the application of manufacturing intelligence to those records, so that the exploitation of those data allows manufacturers to predict, plan and manage specific circumstances in order to optimize their production. Those initiatives enable important business opportunities for the manufacturers.

Moreover, the appropriate design and implementation of such initiatives requires an innovation effort by deploying, among others, mechatronics for advanced manufacturing systems, manufacturing strategies, knowledge-workers and modelling, simulation and forecasting methods and tools [8]. Concerning modeling, a lack of sound descriptions of manufacturing machines that happen to be accessible, interoperable, and reusable can be identified nowadays. Thus, in order to alleviate that existing shortage we have developed an ontology for providing detailed descriptions of a real manufacturing machine type (called ex-

*Corresponding author. E-mail: victorjulio.ramirez@ehu.eus.

truder) that performs an extrusion process.[1] We have not found any other ontology concerning extruders, however, we believe that the ontology-based description of different manufacturing machine types can contribute significantly to the development of the Industry 4.0.

The purpose of this paper is to present the *ExtruOnt* ontology. It includes terms to describe (1) the *main components* of an extruder (e.g. the drive system), (2) the *spatial connections* between the extruder components (e.g. the filter is externally connected to the barrel), (3) the *different features* of the components (e.g. the power consumption of the motor is 40.5 kWh), (4) the *3D description* of the position of the components (e.g. the feed hopper is located at point *q(0,0,-1)* in a 3D canvas), and, (5) the *sensors* that need to be used to capture indicators about the performance of that extruder (e.g the temperature sensor that captures the melting temperature of the polymer).

The *ExtruOnt* ontology has been implemented using OWL 2[2] and the Protégé[3] [23] development environment. *ExtruOnt* is in line with concepts included in an ontology-based context model for industry presented in [13] and is aligned with several ontologies: the DUL[4] ontology, which models physical contexts; the MASON ontology, an upper ontology for representing the core concepts of the manufacturing domain [20]; SAREF4INMA [6], a SAREF [9] extension for semantic interoperability in the industry and manufacturing domain; the GeoSPARQL ontology, which incorporates descriptions about Region Connection Calculus (RCC) [24]; the OM[5] ontology, the largest unit ontology [27]; the 3D Modeling Ontology (3DMO), which maps the entire XSD-based vocabulary of the industry standard X3D[6] (ISO/IEC 19775-19777) to OWL 2 [30] and with the SOSA/SSN, which defines general concepts about sensors [15].

Apart from the interest that the *ExtruOnt* ontology has in itself, the main contributions of the *ExtruOnt* ontology are the following: (1) Reusability. Its modular design facilitates the task of developing other ontologies for different types of manufacturing machines. The module that describes the components of an extruder could be replaced by another module that would describe another type of manufacturing machine, while alignments with other modules should be adapted to meet the requirements of the new type of machine. Moreover, the defined alignments of *ExtruOnt* ontology with upper ontologies such as DUL and MASON facilitate the task of modeling different manufacturing operations (e.g. customer orders, production plans); (2) Expressiveness of Spatial Connections. It incorporates a hierarchical description of possible relations in Region Connection Calculus and some custom-defined ones. Dealing with all those descriptions, more specific spatial relations can be defined and thus fine-grained results for questions can be provided.

Finally, the use of the *ExtruOnt* ontology as the core element of ontology-based systems, developed for Smart Manufacturing scenarios, can bring several benefits. For example, the development of an ontology-based Visual Query System will bring the following benefits to the different types of workers of a manufacturing plant:

- *Novice workers.* The 3D rendering of an extruder machine obtained from descriptions in the ontology will allow novice workers to familiarize themselves with the extrusion process due to its similarity to reality.
- *Product Designers.* The descriptions referring to the components of the extruder as well as the constraints regarding their spatial connections, positioning and features contained in the ontology will facilitate product designers the task of creating customized 3D images of extruder machines.
- *Domain experts.* Ontology-based annotation of data captured by sensors will allow domain experts to perform an assisted exploration of data.

In the rest of this paper, we present first, distinct approaches that have been defined in the literature, related to two aspects considered during the development process of *ExtruOnt*: existing ontologies and ontology evaluation techniques. Then, we show some methodologies that have been proposed to adequately develop ontologies. Next, we illustrate the steps that we followed to develop the *ExtruOnt* ontology using the NeOn methodology [32] and the modules that constitute *ExtruOnt*. Later, we show the results of the evaluation process carried out considering two goals: do-

[1] In which some material is forced through a series of dies in order to create a desired shape.
[2] https://www.w3.org/TR/owl2-overview/
[3] https://protege.stanford.edu/
[4] http://ontologydesignpatterns.org/wiki/Ontology:DOLCE+DnS_Ultralite
[5] https://enterpriseintegrationlab.github.io/icity/OM/doc/index-en.html
[6] http://www.web3d.org/what-x3d-graphics

main coverage and quality of modeling. We finish with some conclusions and future work.

2. Related work

In the specialized literature several ontologies related to the Smart Manufacturing area can be found. Those ontologies were defined with distinct purposes and, therefore, describe different types of information related to that area. For example, the PSL (Process Specification Language) ontology [14] includes fundamental concepts for representing manufacturing processes. The foundational elements of the core of the PSL ontology are four primitive classes (*activity*, *activity-occurrence*, *timepoint*, *object*), three primitive relations (*participates-in*, *before*, *occurrence-of*) and two primitive functions (*beginof*, *endof*). The MASON (Manufacturing's Semantics Ontology) ontology [20] is an upper ontology for representing what authors consider the core concepts of the manufacturing domain: products, processes and resources. As a result, the main classes of MASON are *Entity* (for specifying the product), *Operation* (for describing all processes linked to manufacturing) and *Resource* (for representing concepts regarding machine-tools, tools, human resources and geographic resources). The SIMPM (Semantically Integrated Manufacturing Planning Model) ontology [31] is an upper ontology that models the fundamental constraints of manufacturing process planning: manufacturing activities and resources, time and aggregation. MaRCO (Manufacturing Resource Capability Ontology) [18] defines capabilities of manufacturing resources. Its main class is *Capability*, which is specialized to cover both, simple capabilities (e.g. *Fixturing*, *SpinningTool*) and combined capabilities (those that require a combination of two or more simple capabilities, e.g. *PickAndPlace*, which requires *FingerGrasping* or *Vacuum Grasping*, *Moving* and *Releasing*). The MSDL (Manufacturing Service Description Language) ontology [1] allows to describe manufacturing services. More precisely, a *Manufacturing Service* is seen as a *Service* that is provided by a *Supplier* and that has some *Manufacturing Capability*, which is enabled by some *Manufacturing Resource* and delivered by some *Manufacturing Process*. The P-PSO (Politecnico di MilanoProduction Systems) ontology [11] considers three aspects in the manufacturing domain: the physical aspect (the material definition of the system), the technological aspect (the operational view of the system) and the control aspect (the management activities), for information exchange, design, control, simulation and other applications. Thus, its main classes are *component*, *operation* and *controller*, which model the aforementioned three aspects, as well as *part*, *operator* and *subsystem*. OntoSTEP (ONTOlogy of Standard for the Exchange of Product model data) [2] allows the description of product information mainly related to geometry. MCCO (Manufacturing Core Concepts Ontology) [34] focuses on interoperability across the production and design domains of product lifecycle. It provides some core classes in categories such as *ManufacturingProcess*, *ManufacturingFacility*, *ManufacturingResource* and *Feature*. Finally, SAREF4INMA [9] pursues favouring interoperability with industry standards. Some of its main classes are *ProductionEquipment*, *Factory*, *Item* and *MaterialCategory*.

Although some of the mentioned ontologies contain some general terms for representing the concept of industrial machine (e.g. *Machine-tool* in MASON, *Device* in MarCO, *ProductionEquipment* in SAREF4INMA), further specialization and characterization are needed for fitting our goal, that is, for describing specific industrial machine types with a fine-grained detail, and more particularly, extruder machines. The search on different ontology repositories (e.g. LOV [35], Swoogle [7], ODP [10]) for an ontology that covered this domain yielded unsuccessful, and for that reason we built the *ExtruOnt* ontology following a well-established methodology.

Furthermore, considering the relevance of evaluating the quality and correctness of an ontology once it has been built, several evaluation approaches have been proposed in the specialized literature depending on the evaluation goal. The NeOn guidelines for carrying out the ontology evaluation activity [28] identify the following goals of evaluation: *domain coverage*, *quality of modeling*, *suitability for an application/task* and *adoption and use*. Then, specific evaluation approaches need to be chosen depending on the selected goals. These approaches include, among others, comparing to a gold standard ontology [22], comparing to unstructured or informal data [3], using human assessments [21], and using reasoners to assess the logical correctness of the ontology [17]. Another relevant work in the area of ontology evaluation is the one in [36], where a common framework that considers quality criteria for aspects of ontology evaluation is presented. More precisely, it identifies the following criteria: *accuracy*, *adaptability*, *clarity*, *completeness*, *computational efficiency*, *conciseness*, *consistency* and *organizational fitness*. In the case of the proposed *ExtruOnt* ontology, some aspects considered in those works were taken into account during the evaluation process (see Section 6).

3. Design methodologies

Different methodologies such as On-To-Knowledge [33], Diligent [25] and NeOn [32] can be found in the literature to adequately develop well-founded ontologies. On-To-Knowledge proposes a knowledge meta process consisting of five steps: *feasibility study* to determine whether to begin the actual development of the ontology; *kickoff*, where the requirements are specified and a semi-formal ontology description is developed; *refinement*, where the target ontology is obtained by refining and formalizing the semi-formal one; *evaluation*, where the evaluation of the ontology is done; and *application and evolution*, where the ontology is applied in the target system and maintained. On-To-Knowledge suggests reusing ontologies in the kickoff step if available, but does not provide any guidelines for it. Moreover, it does not deal with non-ontological resources nor other ontological resources such as ontology design patterns. Diligent proposes a process for a distributed development of ontologies that comprises five main steps: *build*, where an initial version of the ontology is built by different stakeholders such as domain experts, users, and knowledge and ontology engineers; *local adaptation*, where users adapt the ontology for their own purposes; *analysis*, where a control board analyses the local versions to detect similarities and decide which changes and requests are added to the next shared version of the ontology; *revision*, where the board revises the new version of the shared ontology; and *local update*, where users can update their local ontologies with information from the new version. This methodology does not detail the series of activities that should be followed during the *build* step, and moreover, it does not include guidelines for using neither ontological nor non-ontological resources in the development process. The NeOn methodology describes a set of nine scenarios that may occur when building an ontology, along with a list of activities that should be carried out in each scenario. Tightly related to those scenarios, it presents two ontology network life cycle models (waterfall and iterative-incremental) with several versions. The basic version is the Four-phase model, which includes the following phases: *initiation*, where the requirements are specified; *design*, where both an informal and a formal model of the ontology are created; *implementation*, where the formal model is implemented in an ontology language; and *maintenance*, where the ontology is used until errors or missing knowledge are detected. The NeOn methodology places special emphasis on reusing and re-engineering both ontological and non-ontological knowledge resources. Thus, more detailed versions of the basic model (e.g Five-phase model, Six-phase + Merging model) include as well one or more of the following phases, resulting in a variety of paths to develop an ontology: *reuse*, where existing ontological or non-ontological resources are added to the model; *re-engineering*, where those resources are modified to serve to the intended purpose; and *merging*, where ontologies are merged or alignments are established among ontological resources. The methodology includes thorough guidelines on how to perform all the mentioned activities.

4. Development of the *ExtruOnt* ontology

In order to develop the *ExtruOnt* ontology we selected the NeOn methodology. In our opinion, NeOn beats the other methodologies in these two aspects: on the one hand, the variety of scenarios that it takes into account, which results in a more flexible methodology, and on the other hand, the great detail in the description of the activities that need to be carried out when building the ontology. Furthermore, due to the requirements of *ExtruOnt*, which include reuse of ontological and no-ontological resources, re-engineering, merging, aligning with domain ontologies, implementation and evaluation among others, its development process fits with the Six-Phase + Merging Phase Waterfall Ontology Network Life Cycle Model. In Fig. 1 the phases of the aforementioned life cycle model along with scenarios, activities and modules of the *ExtruOnt* ontology involved in each scenario are indicated. These modules and their purpose are explained in Section 5. The different phases of the life cycle model are explained below.

4.1. Initiation

In collaboration with the R&D director of a company that manufactures extruder machines, we created the Ontology Requirements Specification Document (ORSD) that contains among others, the purpose of the *ExtruOnt* ontology, its scope and the Compe-

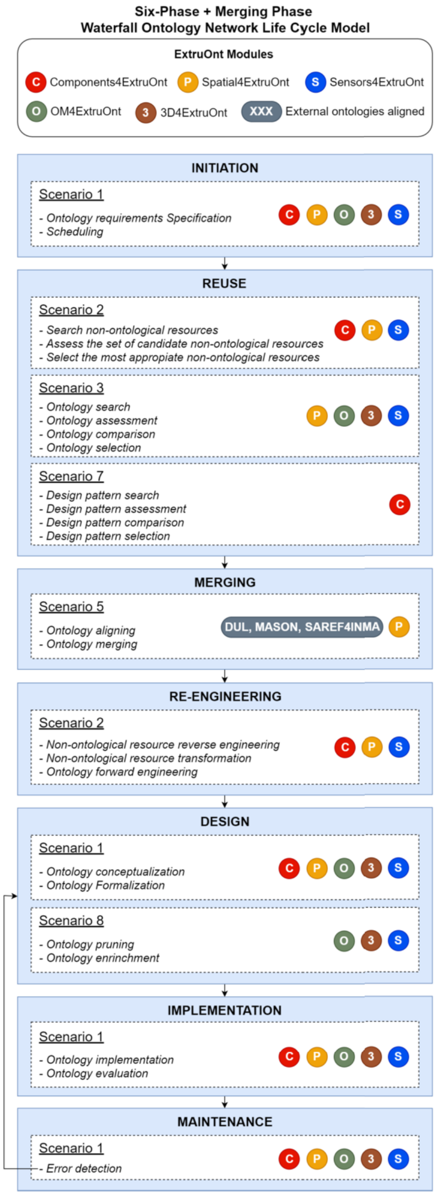


Fig. 1. The Six-Phase + Merging Phase Waterfall Ontology Network Life Cycle Model along with scenarios, activities and *ExtruOnt* modules.

tency Questions (CQs), see Table 1. After a detailed analysis of those questions, it was noticed that they referred to five different dimensions regarding information related to extruders. Thus, the questions were classified in the following five groups, one for each dimension: the components of an extruder, the spatial connections between those components, their features, their 3D description and the sensors that capture information about several indicators (Scenario 1).

4.2. Reuse

Due to the fact that the search for an ontology that covered all these dimensions was unsuccessful, we focused on searching both ontological and non-ontological resources for each dimension.

In this subsection, we present the non-ontological and ontological resources used to describe the aforementioned dimensions.

– *Components of an extruder*: In order to describe the components, we relied on the one hand, on non-ontological resources existing in the specialized literature and mainly in a full chapter dedicated to the extruder and its equipment that appears in [12]. Moreover, due to the complexity of the extrusion head, another non-ontological resource was used as a reference to represent the features of this component. In [29], a thorough explanation of the extrusion head design and applications is presented, categorizing the extrusion head depending on the position and the type of extrudate obtained (Scenario 2). On the other hand, the PartOf[7] ontology design pattern was selected in order to specify parthood between the extruder and its components, as well as between different parts that constitute each component (Scenario 7).
– *Spatial connections between components*: In the specialized literature can be found the Region Connection Calculus (RCC) [5,26], which is intended to represent the spatial relations between objects and facilitate reasoning over those relations. There are multiple representations of the RCC. The main one is RCC8, which consists of 8 basic relations that are possible between two regions. Different ontologies have tried to represent the RCC descriptions (GeoSPARQL[24], Spatial Relations Ontology,[8] NeoGeo Spatial On-

[7] http://ontologydesignpatterns.org
[8] http://data.ordnancesurvey.co.uk/ontology/spatialrelations/

Table 1

Summary of the Ontology Requirements Specification Document for *ExtruOnt*

1. **Purpose**

   The purpose of the *ExtruOnt* ontology is to provide a reference model for the physical representation of extruder machines and the time series data gathered from their sensors, allowing to describe the extruder components, their position with respect to other components and the data obtained from sensing devices.

2. **Scope**

   The ontology will focus on general purpose extruder machines.

3. **Implementation language**

   The ontology has to be implemented in a formalism that allows classification of classes and realization between instances and classes.

4. **Intended users**
   - *User 1*: Novice workers.
   - *User 2*: Product designers.
   - *User 3*: Domain Experts.

5. **Intended uses**
   - *Use 1*: To describe different models of extruders.
   - *Use 2*: To help the process of identifying the extruder components and their location.
   - *Use 3*: To help to select the optimal extruder for a specific product.
   - *Use 4*: To recognize differences between extruder models.
   - *Use 5*: To improve user interaction with the different sensing devices in the extruder and the gathered data.

6. **Ontology requirements**

   (6.a) Non-functional requirements (not applicable)

   (6.b) Functional requirements: Groups of competency questions
   - *CQG1*: Extruder components-related competency questions:
     * CQ1.1: How many heater bands does the extruder E01 have?
     * CQ1.2: What kind of extrusion head does the extruder E02 have?
     * CQ1.3: Is the machine E03 a single or double screw extruder?
     * CQ1.4: Is the extruder E04 powered by an AC motor?
     * CQ1.5: Is this extruder E05 suitable to process plastic pellets?
     * CQ1.6: Can the extruder E06 process multiple polymers?
     * …
   - *CQG2*: Spatial connections-related competency questions:
     * CQ2.1: With which components are the filters FIL01 connected?
     * CQ2.2: Which components overlap the barrel BAR01?
     * CQ2.3: Which components are disconnected from the motor M01?
     * CQ2.4: Which components are monitored in the drive system DS01?
     * CQ2.5: How many sensors does the barrel BAR02 have?
     * …
   - *CQG3*: Features-related competency questions:
     * CQ3.1: What is the diameter of the barrel BAR03?
     * CQ3.2: What are the optimal operating conditions of the screw SCR01?
     * CQ3.3: What is the maximum torque produced by the motor M02?
     * CQ3.4: Does the extruder E07 fit in a space 3 meters wide by 5 meters long?
     * CQ3.5: What is the bottles-per-hour production rate of the extruder E08?
     * …
   - *CQG4*: 3D positioning-related competency questions:
     * CQ4.1: Which components of extruder E11 can not be located in a 3D canvas?
     * CQ4.2: What are the modeling and position of the feed hopper FH01?
     * …

Table 1
(Continued)

	– *CQG5*: Sensors and observations-related competency questions: * CQ5.1: What properties are observed by the sensors located in the extrusion head EH01? * CQ5.2: What is the unit of measurement used by the motor consumption sensor MCS01? * CQ5.3: Where is the melting temperature sensor located in extruder E08? * CQ5.4: What is the identifier of the temperature sensor in extrusion head EH02? * CQ5.5: When was the first and last observation made by sensor SN01? * CQ5.6: What was the average, maximum and minimum value of the observations in a day for the sensor SN02? * CQ5.7: How many observations from torque sensor SN03 are outside the optimal values? * CQ5.8: how long was the maximum period of extruder E09 inactivity during the last week? * CQ5.9: At what times during August 21st, 2018 and August 22nd, 2018 did the melting temperature exceed the maximum optimal operational value in extruder E10? *…
7.	**Pre-glossary of terms**
	Extruder, feed system, observation, sensor, tangential proper part, measure, 3D canvas …

tology[9]). We selected the GeoSPARQL ontology, which models the RCC8 relations, because it is the base for the other spatial ontologies (Scenario 3).

- *Features of the components*: Based on a work that evaluates ontologies of measurements [19], two ontologies were considered: QUDT[10] [16] and OM[11] [27]. QUDT is the result of a NASA-sponsored initiative to formalize Quantities, Units of Measure, Dimensions and Types, and it is categorized as a medium sized ontology. OM is an ontology that allows to model concepts and relations in the context of food research and it was the largest unit ontology compared. In the aforementioned evaluation, multiple issues were found in QUDT ontology like reasoning impossibility, duplicated units, wrong specifications, typing errors, etc. Moreover, only English labels were added and, according to the article, the reported issues remain unsolved. On the other hand, OM shared some issues with QUDT like reasoning impossibility, wrong dimension values, typing errors, but the reported issues have been corrected and labelling can be found in Dutch and Chinese for a subset of individuals. Equally important, more concepts can be found in OM, so this was the selected ontology (Scenario 3).
- *3D representation of components*: We selected the 3D Modeling Ontology (3DMO) [30] because this ontology maps the entire XSD-based vocabulary of the industry standard X3D[12] (ISO/IEC 19775-19777) to OWL 2. Therefore, it can be used for the representation, annotation, and efficient indexing of 3D models (Scenario 3).
- *Sensors for capturing information about indicators*: We did not find any ontological resource that defines the specific types of sensors that are used to monitor extruders. However, the well known SOSA/SSN[15] ontology defines general concepts about sensors, which can be specialized with information obtained from non-ontological resources about extruders [12] to reflect the specificities of the extrusion domain (Scenario 3).

4.3. Merging

To guarantee semantic interoperability, the *ExtruOnt* ontology is aligned with other domain ontologies such as: (1) DUL, an upper ontology created to provide a set of concepts to facilitate interoperability among ontologies; (2) MASON, an upper ontology for representing the core concepts of the manufacturing domain and (3) SAREF4INMA, a SAREF extension for industry and manufacturing (Scenario 5). The selection of these ontologies was carried out taking into account different key factors such as domain, use, maintenance, acceptance, popularity and coverage. For example, in the selection of MASON, other different ontologies were considered: MaRCO, whose approach is oriented to machine capabilities and, thus, out of our scope; MSDL, with a large amount of concepts fo-

[9] http://geovocab.org/doc/neogeo/
[10] http://www.linkedmodel.org/catalog/qudt/1.1/index.html
[11] https://enterpriseintegrationlab.github.io/icity/OM/doc/index-en.html
[12] http://www.web3d.org/what-x3d-graphics

cused on processes and resources but leaving products aside; SIMPM, with few concepts and focused only on the processes; and finally, PSL, P-PSO, MCCO and OntoSTEP whose OWL definitions could not be found. On the contrary, MASON defines a meaningful categorization of products, processes and resources, it has been widely reviewed [4] and it is currently available. The terms used in the ontology alignment are presented in Section 5.1.

Concerning to the spatial connection between components, we realized that using only the GeoSPARQL ontology was not sufficient for answering competency question CQ2.2. Thus, a twofold approach was used: in addition to the GeoSPARQL ontology, information about other RCC spatial relations obtained from the aforementioned non-ontological RCC resources was incorporated (Scenario 5).

4.4. Re-engineering

A re-engineering process was carried out to transform the non-ontological resources mentioned previously into conceptual models, analyzing the structure of the resource (chapters, subsections, connections, order, etc.). Once the conceptual model for each resource had been created, they were used as input of the design phase. (Scenario 2).

4.5. Design

The modularization of ontologies facilitates the development, reuse and maintenance of an ontology. In addition, it conforms to the dimensionality approach obtained from the ORSD analysis. Therefore, each of the five dimensions was represented through a module: the components of an extruder (*components4ExtruOnt*), the spatial connections between those components (*spatial4ExtruOnt*), their features (*OM4ExtruOnt*), their 3D description (*3D4ExtruOnt*) and the sensors that capture information about several indicators (*sensors4ExtruOnt*), which altogether form the *ExtruOnt*[13] ontology (Scenario 1). The key features of each module are presented in depth in Section 5.

OM, SOSA/SSN and 3DMO ontologies contain a wide range of concepts that belong to the domains they represent, however, due to the specific domain we wanted to model, a pruning process was carried out for these ontologies keeping only those concepts and descriptions that are relevant, favoring lightness, cleanliness and maintenance of the ontology (Scenario 8). Additionally, the pruned SOSA/SSN ontology was enriched with specialized concepts drawn from the conceptual model (see Section 5.5).

[13] http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/ExtruOnt.owl

4.6. Implementation

A formal model expressed in a Description Logic was generated and implemented in OWL 2 DL Web Ontology Language using Protégé [23] (Scenario 1). Later, a wide evaluation of the ontology was done which is presented in Section 6, describing the different considered approaches.

4.7. Maintenance

The maintenance phase is currently undergoing. Once an error is detected, the ontology will be taken to the design phase to be corrected, as stipulated in the Waterfall ontology network life cycle model.

5. Ontology modules

As said before, *ExtruOnt* is divided in five modules aiming to describe several characteristics of an extruder machine (see Fig. 2).

In the following, the key features of each module are presented.

5.1. components4ExtruOnt

The *components4ExtruOnt*[14] module is the main module of the *ExtruOnt* ontology and is intended to describe the components of an extruder. According to [12], five major systems can be distinguished in an extruder:

- Drive system.
- Feed system.
- Screw, barrel and heating system.
- Head and die assembly.
- Control system.

Moreover, the components of each one of these systems are explained. For instance, the drive system is composed of motor, gear box, bull gear, and thrust

[14] http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/components4ExtruOnt.owl

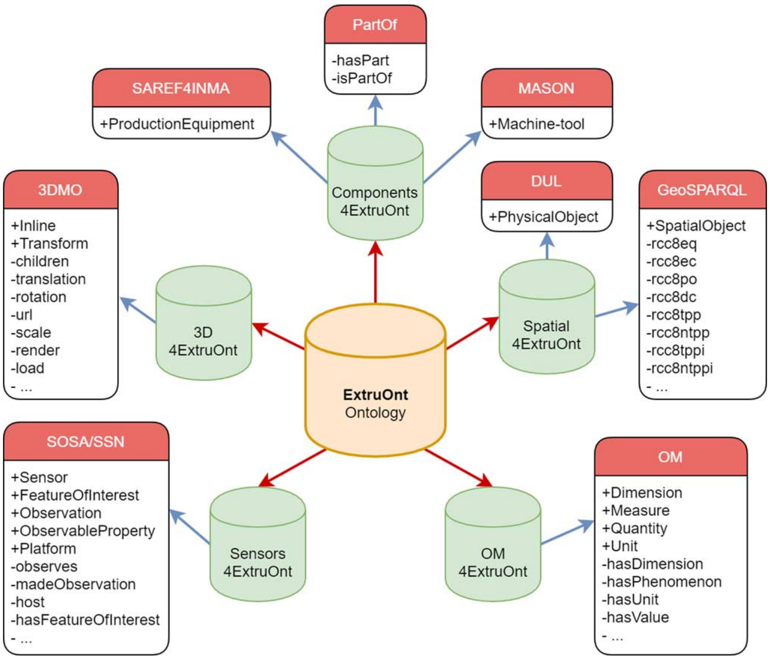


Fig. 2. *ExtruOnt* ontology diagram showing the reuse of terms from other domain ontologies.

bearing; and the head and die assembly contains the head, die/nozzle, breaker plate and filters/screens. This analysis of the components of the extruder was used as base to create the *components4ExtruOnt* module.

A new main class called `Extruder` was created for representing the extrusion machine, while the connections between the extruder and its systems and components were made using the `hasPart` object property of the `PartOf`[15] ontology design pattern. Moreover, custom-made specializations of `hasPart` were created to relate specific components, e.g., `hasBarrel`, `hasScrew` and `hasHeaterBand`. The parthood relations of the extruder and its components are shown in Fig. 3. To facilitate integration with other domain ontologies, the terms `saref4inma:ProductionEquipment`[16] and `MASON:Machine-tool`[17] were included as superclasses of `Extruder`.

[15] http://www.ontologydesignpatterns.org/cp/owl/partof.owl
[16] https://w3id.org/def/saref4inma
[17] https://sourceforge.net/projects/mason-onto/

Moreover, the specialization of each component was represented using `rdfs:subClassOf` relations. An example is illustrated in Fig. 4.

With respect to the extrusion head, the classification that can be found in [29] was used to provide a detailed representation of this component. Figures 5 and 6 exemplify this representation.

Among others, the following competency questions are resolved with the *components4ExtruOnt* module:

- CQ1.1: How many heater bands does the extruder E01 have?
- CQ1.2: What kind of extrusion head does the extruder E02 have?
- CQ1.3: Is the machine E03 a single or double screw extruder?
- CQ1.4: Is the extruder E04 powered by an AC motor?
- CQ1.5: Is this extruder E05 suitable to process plastic pellets?
- CQ1.6: Can the extruder E06 process multiple polymers?

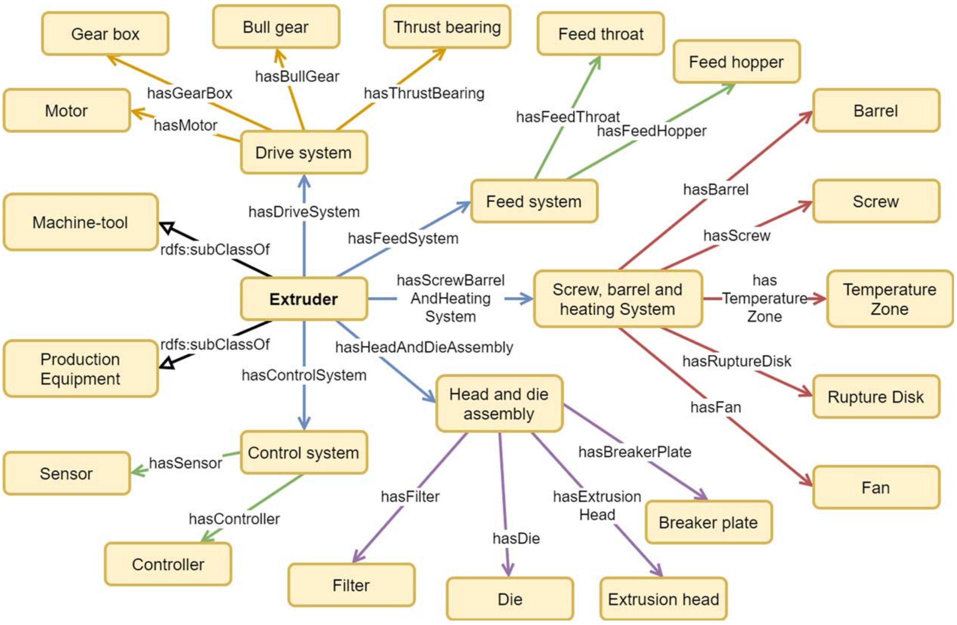


Fig. 3. Some components of an extruder.

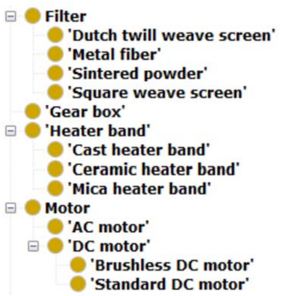


Fig. 4. Excerpt of the class hierarchy of the components.

A SPARQL query to answer the competency question CQ1.4 is as follows:[18]

```
PREFIX : <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/Extruder01#>
PREFIX rdf: <http://www.w3.org/1999/02/
    22-rdf-syntax-ns#>
PREFIX c4e: <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/components4ExtruOnt#>
PREFIX p: <http://www.ontologydesignpatterns.org/
    cp/owl/partof.owl#>
ASK { :E04 p:hasPart ?motor01.
    ?motor01 a c4e:AC_motor
        }
```

[18]We assume that the query is executed after inferences are provided by a reasoner. (This applies for all the examples in this paper.)

As a result, the description of the extruder in the *components4ExtruOnt* module will help novice workers to recognize its different sections and components. Moreover, it will help domain experts to formulate queries, according to their needs, related to the amount of components and their types.

5.2. *spatial4ExtruOnt*

The main representation of RCC is RCC8, which consists of 8 basic relations that are possible between two regions: Equal (EQ), Disconnected (DC), Externally Connected (EC), Partially Overlapping (PO), Tangential Proper Part (TPP), Non-Tangential Proper Part (NTPP), Tangential Proper Part inverse (TPPi) and Non-Tangential Proper Part inverse (NTTPi). A stripped down version of RCC8 is RCC5, which consists of 5 relations: Equal (EQ), Discrete (DR), Par-

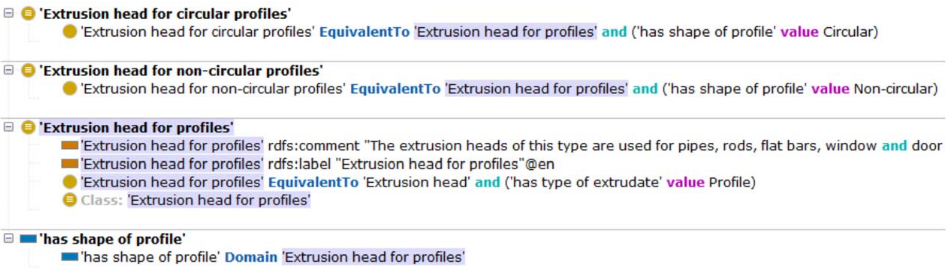


Fig. 5. Definition of the Extrusion head for profiles.

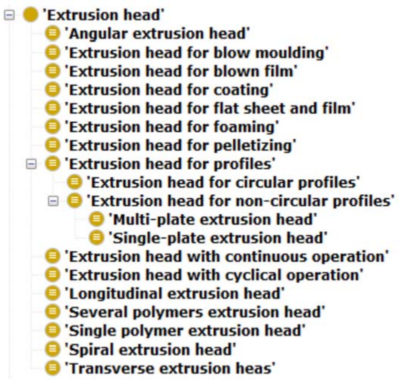


Fig. 6. Subclasses of Extrusion head.

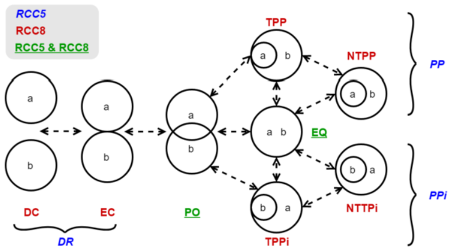


Fig. 7. RCC5 and RCC8 relations.

tially Overlapping (PO), Proper Part (PP) and Proper Part inverse (PPi). The graphical representation of RCC5 and RCC8 relations with their mappings are shown in Fig. 7.

For the *spatial4ExtruOnt*[19] module, a submodule of the GeoSPARQL ontology was used, which contains the `SpatialObject` main class and the object properties referencing to the RCC8 relations. To encourage semantic interoperability, the term `Physical-Object` from DUL ontology[20] was included as a superclass of `SpatialObject`. Moreover, a hierarchical object property representation was made including RCC8 relations connected to RCC5 ones, and some more general custom-defined properties. For example, `rcc8tpp` (tangential proper part) is a subproperty of `rcc5pp` (proper part) and, in the same way, `rcc5pp` is a subproperty of the custom-made `overlapsNotEquals` object property. Another example is the following: when two objects overlap, three possible situations can occur: (1) `A` is equal to `B`, (2) `A` partially overlaps `B` and (3) `A` overlaps but is not equal to `B`. This is represented with the `overlaps` object property and three subproperties: `rcc8eq` (equals), `rcc8po` (partially overlapping) and `overlapsNotEquals` (overlaps but not equal). This hierarchy allows a fine-grained classification of spatial relations and can provide detailed results to general questions, e.g., the answer to the question about the objects that overlaps object X will return those objects that are equals, partially overlapping and proper part of object X. The object property hierarchy is shown in Fig. 8.

RCC8 also defines a composition table where the possible relations between an object A and an object C are indicated based on the relation between object A and B, and the relation between object B and C. However, the OWL 2 DL expressivity level is not sufficient to represent the full table, and for that reason, in *spatial4ExtruOnt* only compositions that yield a single re-

[19] http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/spatial4ExtruOnt.owl

[20] http://ontologydesignpatterns.org/wiki/Ontology:DOLCE+DnS_Ultralite

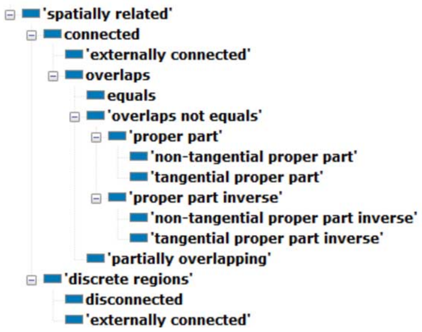


Fig. 8. Object property hierarchy in *spatial4ExtruOnt*.

sult for the type of relation between objects A and C have been defined in the ontology, more precisely by means of property chains (see Fig. 9).

Once the *spatial4ExtruOnt* module was added to *ExtruOnt*, it was possible to describe the spatial connections between the components of the extruder. The classes that describe single components were declared as subclasses of the `SpatialObject` class and the relations between components were made. For example: the filter is externally connected to the barrel and the breaker plate, and it is a tangential proper part of the extrusion head (Fig. 10).

With the *spatial4ExtruOnt* module, it is possible to answer several competency questions. These are some of them:

- CQ2.1: With which components are the filters FIL01 connected?
- CQ2.2: Which components overlap the barrel BAR01?
- CQ2.3: Which components are disconnected with the motor M01?
- CQ2.4: Which components are monitored in the drive system DS01?
- CQ2.5: How many sensors does the barrel BAR02 have?

The CQ2.2 competency question is resolved with the following SPARQL query:

```
PREFIX : <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/Extruder01#>
PREFIX s4e: <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/spatial4ExtruOnt#>
SELECT DISTINCT ?component
WHERE {
  {?component s4e:overlaps :BAR01}
    UNION
  {:BAR01 s4e:overlaps ?component}
}
```

The *spatial4ExtruOnt* module will allow novice workers to understand the spatial connections between the different components of an extruder. Furthermore, it will help product designers and domain experts to define the distribution of the components, e.g., the position of the sensors in the head and die assembly.

5.3. *OM4ExtruOnt*

The objective of the *OM4ExtruOnt*[21] module is to provide the terms that are necessary to describe the features of the components. This is an important step in the representation of the extruder, as single components could have different characteristics: a barrel could have different dimensions and manufacturing materials.

A submodule of the OM ontology was used to create *OM4ExtruOnt*, where only the concepts useful for characterizing the components of the extruder and process were taken into account. As stated before, due to the fact that OM is an ontology in the context of food research, it is common to find concepts like `NumberColor1` and `NumberRottenFlowers` to refer to the avocado color and flower status respectively. Consequently, these concepts were removed keeping only concepts like temperature, speed, size, etc.

The elements of the *OM4ExtruOnt* module can be connected to the elements of the *components4ExtruOnt* module by means of the object property `hasPhenomenon`, which links a measure made for a feature with the object to which the measure applies. For example, in Fig. 11 a measure (`ex:VoltageMeasure01`) of the motor voltage (`ex:MotorVoltage01`) of a specific motor (`ex:Motor01`) is represented, which in this case takes the value of 220 volts.

Once the features of the components are defined using the *OM4ExtruOnt* module, it is possible to answer more competency questions, such as:

- CQ3.1: What is the diameter of the barrel BAR03?
- CQ3.2: What are the optimal operating conditions of the screw SCR01?
- CQ3.3: What is the maximum torque produced by the motor M02?
- CQ3.4: Does the extruder E07 fit in a space 3 meters wide by 5 meters long?

[21] http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/OM4ExtruOnt.owl

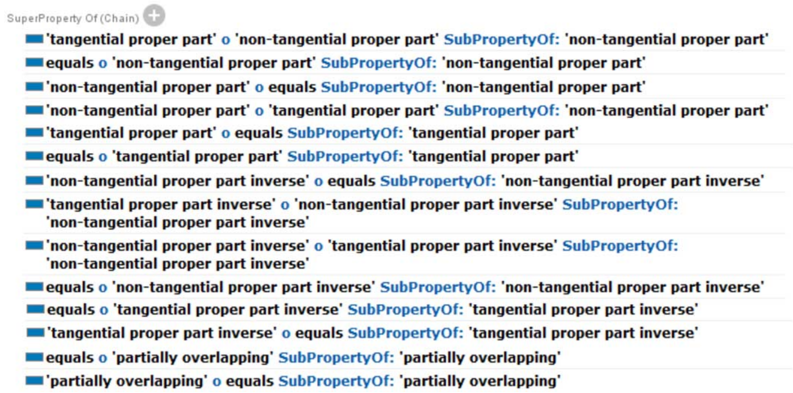


Fig. 9. Property chains defined in *spatial4ExtruOnt*.

Fig. 10. Excerpt of the Filter class description.

– CQ3.5: What is the bottles-per-hour production rate of the extruder E08?

To solve the CQ3.3 competency question a SPARQL query was designed:

```
PREFIX : <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/Extruder01#>
PREFIX rdf: <http://www.w3.org/1999/02/
    22-rdf-syntax-ns#>
PREFIX om: <http://www.ontology-of-units-of-
    measure.org/resource/om-2/>
SELECT ?motorTorque01 ?torqueMeasure ?value ?unit
WHERE { ?motorTorque01 a om:Torque.
  ?motorTorque01 om:hasPhenomenon :M02.
  ?motorTorque01 om:hasValue ?torqueMeasure.
  ?torqueMeasure om:hasUnit ?unit;
  om:hasNumericalValue ?value.
}
```

On the one hand, the definition of the features of the components made on the *OM4ExtruOnt* module will contribute to the novice workers' awareness of the maximum operating condition of the components. On the other hand, it provides a tool for domain experts to annotate the features of the components, gathered from the design process facilitating the preparation of their specification.

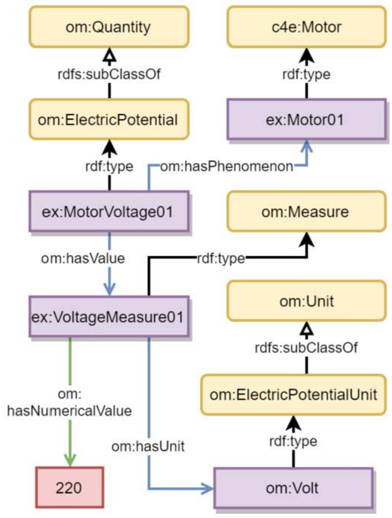


Fig. 11. Example of definition of a measure for the feature Motor voltage.

5.4. *3D4ExtruOnt*

The graphic representation of an extruder permits to visually understand/observe the positioning of each component that is part of it. Many images of extruders can be found in books, articles, brochures and websites. However, the limitations of a 2D environment makes it difficult to visualize the exact position of the

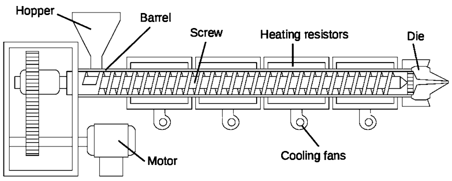


Fig. 12. 2D representation of the components of an extruder.

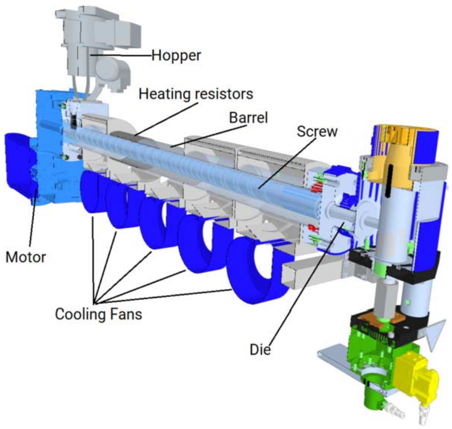


Fig. 13. 3D representation of the components of an extruder.

components. Thus, the understanding of an extruder is limited due to the lack of interaction, and the viewer is restricted to the bi-dimensional expressiveness of the author (Fig. 12). On the contrary, a 3D representation of an extruder allows to improve the viewer's interaction, facilitating to move, rotate, zoom in and zoom out. This advantage provides each user with a personalized experience (Fig. 13).

The purpose of the *3D4ExtruOnt*[22] module is to provide terms for describing the position of each single component in the extruder, in a way that each single component model can be located in a 3D canvas.

X3D is a royalty-free open standards file format and run-time architecture to represent and communicate 3D scenes and objects, which is approved for the International Standards Organization (ISO). With a set of rich features, X3D can be used in scientific visualization, CAD and architecture, training and simulation, etc. and supports:

[22] http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/3D4ExtruOnt.owl

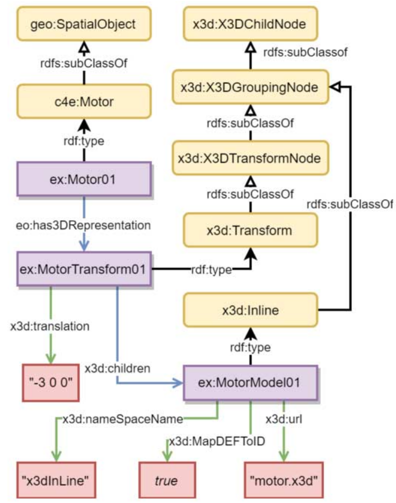


Fig. 14. Definition of motor model location in a 3D canvas.

- 3D graphics and programmable shaders.
- 2D graphics.
- CAD data.
- Animation.
- User interaction.
- Navigation.

The selected 3DMO ontology contains a complete X3D definition. To build the *3D4ExtruOnt* module, only the section referring to the 3D object positioning was selected. To connect the elements of the *3D4ExtruOnt* module with the elements of the *components4ExtruOnt* module, a new `has3DRepresentation` object property was included, whose range is the X3D `Transform` class and the domain is the `SpatialObject` class, previously mentioned. `Transform` class provides the `translation` property where the x, y and z coordinates, referring to the position of a 3D model in a canvas, can be specified. The `Inline` class allows to load different external 3D file formats (obj, stl, collada, fbx, etc.) by using the `url` property to specify the path to the resource location. An example of the 3D positioning of the motor is shown in Fig. 14.

Now, it is possible to answer competency questions referring to 3D object positioning, for example:

- CQ4.1: Which components of extruder E11 can not be located in a 3D canvas?

– CQ4.2: What are the modeling and position of the feed hopper FH01?

The following SPARQL query can be used to answer the competency question CQ4.2:

```
PREFIX : <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/Extruder01#>
PREFIX rdf: <http://www.w3.org/1999/02/
    22-rdf-syntax-ns#>
PREFIX e: <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/ExtruOnt#>
PREFIX x3d: <http://purl.org/ontology/x3d/>
SELECT ?position ?nameSpace ?id ?url
WHERE { :FH01 e:has3DRepresentation ?hopper3d.
  ?hopper3d a x3d:Transform;
    x3d:translation ?position;
    x3d:children ?model3d.
  ?model3d a x3d:Inline;
    x3d:nameSpaceName ?nameSpace;
    x3d:MapDEFToID ?id;
    x3d:url ?url.
}
```

The *3D4ExtruOnt* module will help domain experts in the design process of components, by providing the required information to position 3D models of components in a scene. Moreover, the detection of faults or collisions will be facilitated. Furthermore, it will help novice workers to understand the physical appearance of single components and recognize them in real-world scenarios.

5.5. *sensors4ExtruOnt*

This module is intended to enable domain experts to gain a greater value and insights out of the captured data from the sensors of the extruders, in order to keep trace of the performance of the extruder and allowing to detect possible future faults.

The *sensors4ExtruOnt*[23] module imports the *SOSA/SSN* [15] and *OM4ExtruOnt* ontologies. The class `Sensor` was created as a specialization of `sosa:Sensor`. Two properties were added to this class: `indicatorId` (the identifier of the sensor) and `sensorName` (the name of the sensor). Moreover, two main subclasses of Sensor were defined: `BooleanSensor` and `DoubleValueSensor` to represent sensors that capture true/false data and numerical data respectively. Finally, these two subclasses were specialized for describing more specific type of sensors, more precisely sensors for observing: whether a resistor is on or off, whether a fan is on or off, the level and composition of the additive, the number of bottles made in a shift, the feed rate of the polymer, the melting temperature of the polymer, the power consumption of the motor, the pressure in the pressurized zones of the extruder, the speed of the rotational components, the temperature, the thickness of the extrudate and the viscosity of the extrudate.

The observable property for each sensor type is indicated by `sosa:observes`. For example, the observable property of a `MotorConsumptionSensor` is `Power` (imported from *OM4ExtruOnt*) and its unit is `Watt`, an individual of `PowerUnit`. Each sensor type is related to the type of observation that it makes through the `sosa:madeObservation` property. For each observation, its value and timestamp are indicated by properties `sosa:hasSimpleResult` and `sosa:ResultTime` respectively. The annotations made in the data and the descriptions in the module can be used to generate a customized and semantically enriched chart to visualize the data. For example, when a sensor is defined as an individual of `MotorConsumptionSensor` class, it can be inferred that it captures values in Watts, its symbol is W and its optimal operational values are between 15,600 and 20,000 units. This information can be used to select the most convenient visual representation of the data, improving the analysis and user experience. An excerpt of the module can be found in Fig. 15.

In order to indicate the spatial location of a sensor in the extruder the terms described in the module *spatial4ExtruOnt* can be used. In addition, the parts of the extruder (described in the module *components4ExtruOnt*) that host sensors can be seen as `sosa:Platforms`, and linked to them via the object property `sosa:hosts`. Finally, the feature of interest of the observations of each type of sensors has been indicated using the property `sosa:hasFeatureOfInterest`. For example, in the case of a `MotorConsumptionSensor` the motor of the extruder is both its platform and its feature of interest, while in the case of a `MeltingTemperatureSensor` the platform is the barrel of the extruder and its feature of interest is the polymer used in that extrusion process (see Fig. 16).

With the addition of this module, a selection of competency questions can be solved, among others:

– CQ5.1: What properties are observed by the sensors located in the extrusion head EH01?
– CQ5.2: What is the unit of measurement used by the motor consumption sensor MCS01?

[23] http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/sensors4ExtruOnt.owl

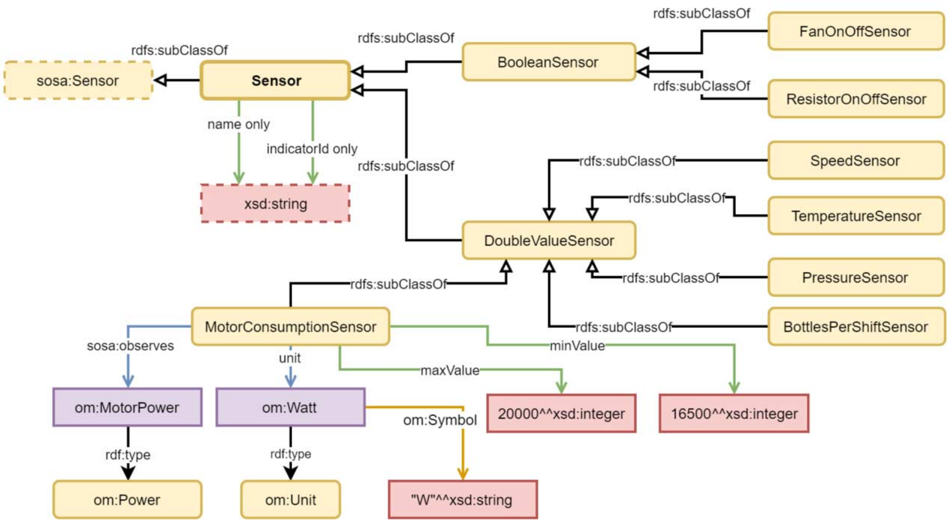


Fig. 15. Excerpt of the *sensors4ExtruOnt* module showing some classes and properties related to sensors.

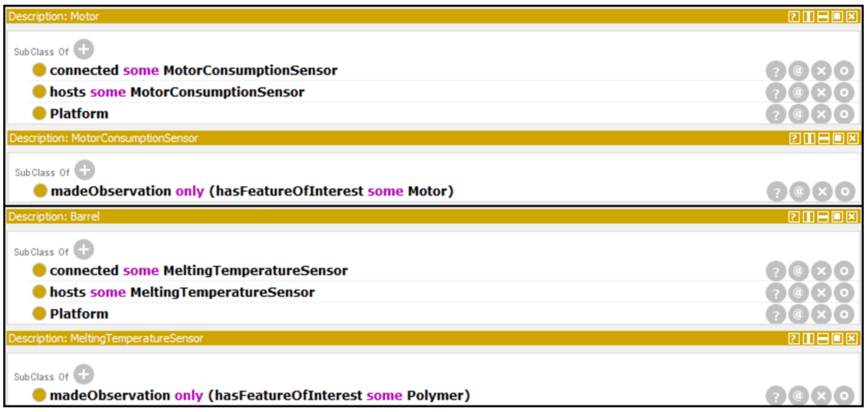


Fig. 16. Excerpt of the descriptions of classes Motor, MotorConsumptionSensor, Barrel and MeltingTemperatureSensor.

– CQ5.3: Where is the melting temperature sensor located in extruder E08?
– CQ5.4: What is the identifier of the temperature sensor in extrusion head EH02?
– CQ5.5: When was the first and last observation made by sensor SN01?
– CQ5.6: What was the average, maximum and minimum value of the observations in a day for the sensor SN02?
– CQ5.7: How many observations from torque sensor SN03 are outside the optimal values?
– CQ5.8: How long was the maximum period of extruder E09 inactivity during the last week?

– CQ5.9: At what times during August 21st, 2018 and August 22nd, 2018 did the melting temperature exceed the maximum optimal operational value in extruder E10?

A SPARQL query to answer the CQ5.9 competency question is presented as follows:

```
prefix :<http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/Extruder01#>
prefix sosa: <http://www.w3.org/ns/sosa/>
prefix xsd: <http://www.w3.org/2001/XMLSchema#>
prefix sn4e: <http://bdi.si.ehu.es/bdi/ontologies/
    ExtruOnt/sensors4ExtruOnt#>
PREFIX p: <http://www.ontologydesignpatterns.org/
    cp/owl/partof.owl#>
select ?resultValue ?resultTime
where {
  :E10 p:hasPart ?barrel01.
  ?barrel a c4e:Barrel .
  ?barrel sosa:hosts ?meltingTempSn01 .
  ?meltingTempSn01 a sn4e:MeltingTemperatureSensor;
    sosa:madeObservation ?obs;
    sn4e:maxValue ?maxValue.
  ?obs sosa:hasSimpleResult ?resultValue ;
    sosa:resultTime ?resultTime .
  filter(?resultValue > ?maxValue) .
  filter((xsd:dateTime(?resultTime) >=
    "2018-08-21T00:00:00.000Z"^^xsd:dateTime) &&
    (xsd:dateTime(?resultTime) <=
    "2018-08-22T23:59:59.999Z"^^xsd:dateTime))
}
order by asc(?resultTime)
```

The *sensors4ExtruOnt* module allows domain experts to analyze and keep trace of sensors data in a structured way, retaining important relations and properties between the data, sensors and components of an extruder, which can be valuable in a future failure prediction process.

6. Evaluation

Once the *ExtruOnt* ontology was developed, in order to check its quality, two evaluation goals were considered: *Domain coverage* to see in which extent it covered the considered extrusion domain, and *Quality of the modeling* in terms of the design and development process and in terms of the final result. The third goal identified by NeOn (*Suitability for an application/task*) will be considered once software artifacts whose core element is the *ExtruOnt* ontology (see Section 7) are built. Moreover, the passing of time will allow to evaluate the ontology regarding the goal of *Adoption and use*. During the evaluation process, the ontology was also assessed by three types of persons: (1) A R&D director of a company that develops machines that produce bottles based on an extrusion process, who we work closely with. This person also provides us real data captured from the machines developed by his company. (2) A director of an IBDS (Industrial Big Data Services) Provider company. IBDS is an ITS (Information Technology Supplier) company that supplies manufacturers with the required technology and services to smartize their manufacturing businesses. Thus, IBDS Providers constitute a fundamental agent in industrial scenarios where there is an interest in adopting Smart Manufacturing approaches. (3) An expert in developing and managing ontologies who works in a technology center specialized in the industrial domain.

6.1. *Domain coverage*

Using the non-ontological resources and reusing some other existing ontologies related to the dimensions considered in the ontology, a first version of *ExtruOnt* ontology was built. Then, after a rigorous discussion process with the three experts, who evaluated the correctness and usefulness of the described information in the ontology, it was redefined and some new terms were incorporated and some others were eliminated. Thus, R&D director of the company, based on his knowledge about the extrusion process, evaluated the semantic quality of the ontology. For example, he suggested to eliminate the three types of categories that we defined related to type of heads (that appear in the non-ontological resource regarding extrusion heads) and refer them through the definition of new features in the existing extrusion head term (for example, shape of profile and quantity of plates) in order to avoid some ambiguities in the representation.The director of an IBDS, based on his acquired knowledge by providing smart manufacturing services to different types of manufacturing companies, evaluated to what extent the ontology could be adapted and used in other manufacturing scenarios. Considering his comments we saw interesting to deal with two upper ontologies: DUL and MASON (the last one focused on the manufacturing domain), because they contain terms that could be relevant in other scenarios, for example *process* and *operation* terms to describe the logistics, schedule and maintenance operations in a factory. Finally, the expert on ontologies evaluated the quality of the alignments with existing ontologies. In this sense, he suggested the alignments with SAREF4INMA instead of SAREF, as was our first approach. In the final version of the *ExtruOnt* ontology, regarding the main concepts described in the non-ontological resources, 125 terms were included, and regarding those related to the extrusion head, 32 were included; covering the 95% of the vocabulary. The remaining 5% corresponds to terms out of the ontology scope or without significant value (e.g., parts of obsolete extruder models). Evaluation against a gold standard was not possible because after performing a thorough search we could not find a gold standard source to compare. Nevertheless, we will continue with the search process and, as soon as we find it, an additional evaluation step will be performed to reinforce the adaptability and reuse tests made to the ontology.

6.2. *Quality of the modeling*

This evaluation goal focuses on the quality of the ontology and can be assessed using a wide range of approaches. In this section we focus on ontology metrics, in common pitfalls in the ontology development process and in the contrast of some defined criteria used for the evaluation of the ontology during the development process. We selected these approaches because, using all three, a fairly accurate picture of the ontology quality can be obtained.

6.2.1. *Ontology metrics*

The basic ontology metrics, including amount of axioms, classes, properties and individuals in the ontology, were extracted from Protégé. They are listed in Table 2. A schema and graph metrics comparison with other ontologies of the manufacturing domain is listed in Table 3. The data was extracted using OntoMetrics.[24] As it can be seen, the metrics for *ExtruOnt* remain in the range of values of other well-known manufacturing domain ontologies. Some metrics like *Inheritance Richness* and *Equivalence Ratio* present a moderate high value due to the semantic interoperability level achieved, i.e., the amount of reused ontologies. However, comparing specific metrics like *tCardinality*, *Depth* and *xtBreadth* would be unfair since the level of abstraction of the compared ontologies differs.

6.2.2. *OOPS! evaluation*

The Ontology Pitfall scanner (OOPS!) evaluates an ontology by searching for design pitfalls considered from a catalogue of 41 common pitfalls in the ontology development process, classified in a three level scale: critical, important and minor. Most of them (33 out of 41 pitfalls) can be identified semi-automatically by OOPS!. The initial evaluation of *ExtruOnt* yielded some flaws that were corrected, nonetheless, 2 minor pitfalls remain due to external ontology imports. Table 4 presents the evaluation summary made by OOPS!.

6.2.3. *Evaluation criteria during the development process*

Criteria defined in [36] were used for the evaluation of the ontology during the development process. These criteria are listed below with an explanation of their application in *ExtruOnt*.

- *Accuracy*: The ontology development process was assisted by three experts. Moreover, the modules of *ExtruOnt* were designed using well supported ontological and non-ontological resources. As evidence, *components4ExtruOnt* was created using two non-ontological resources [12,29], *spatial4ExtruOnt* is based in the Region Connection Calculus relations, *OM4ExtruOnt* uses a submodule of the well known OM ontology, *3D4ExtruOnt* uses concepts from the 3DMO ontology, which follows an ISO open standard (X3D) and finally, *sensors4ExtruOnt* imports definitions from SOSA/SSN ontology.
- *Adaptability*: Each module of *ExtruOnt* can be used individually. Thus, it provides reusability and extensibility, making the ontology easily adaptable to describe other different industrial machines. For example, to describe a wire drawing machine,[25] a new main ontology should be created (e.g. *WidraOnt*), importing on it four modules from *ExtruOnt*, more precisely, the *spatial4ExtruOnt*, *OM4ExtruOnt*, *sensors4ExtruOnt* and *3D4ExtruOnt* modules, which do not have to be modified since the terms in these modules describe information related to general manufacturing machines. Therefore, only the *components4ExtruOnt* module should be redefined (e.g. *components4WidraOnt*), incorporating to it terms referring to the new components (such as puller, coiling roller, capstan, wire, etc.) that belong to the new machine, importing some terms from *components4ExtruOnt* (such as motor, gearbox, etc.) that are shared between both machines and leaving out some other terms (such as extrusion head, barrel, hopper, etc.) that do

[24] https://ontometrics.informatik.uni-rostock.de/ontologymetrics/index.jsp

[25] A machine that reduces the diameter of a wire by pulling it through a single or a series of drawing dies.

Table 2

Ontology metrics

	Components	Spatial	OM	Sensors	3D	ExtruOnt
Metrics						
Axiom	1010	378	3740	775	111	6021
Logical axiom count	506	88	1946	199	36	2779
Declaration axioms count	167	40	477	113	25	822
Class count	80	1	107	52	8	248
Object property count	60	15	17	38	1	131
Data property count	0	0	11	9	13	33
Individual count	17	0	308	7	0	332
Annotation Property count	19	28	39	21	8	115
DL expressivity	SHOIQ	ALRI+	ALCHON(D)	ALCROIN(D)	ALC(D)	SROIQ(D)
Class axioms						
SubClassOf	302	0	148	146	6	602
EquivalentClasses	25	0	47	0	0	72
DisjointClasses	11	0	0	3	2	16
GCI count	0	0	0	0	0	0
Hidden GCI Count	1	0	47	0	0	48
Object property axioms						
SubObjectPropertyOf	52	15	1	1	0	69
EquivalentObjectProperties	0	0	0	0	0	0
InverseObjectProperties	25	3	0	14	0	42
DisjointObjectProperties	0	0	0	0	0	0
FunctionalObjectProperty	0	0	1	2	0	3
InverseFunctionalObjectProperty	0	0	0	1	0	1
TransitiveObjectProperty	2	3	0	0	0	5
SymmetricObjectProperty	0	9	0	0	0	9
AsymmetricObjectProperty	0	0	0	0	0	0
ReflexiveObjectProperty	0	1	0	0	0	1
IrrefexiveObjectProperty	0	0	0	0	0	0
ObjectPropertyDomain	35	15	15	2	1	68
ObjectPropertyRange	36	15	16	2	1	70
SubPropertyChainOf	0	27	0	4	0	31
Data property axioms						
SubDataPropertyOf	0	0	0	0	0	0
EquivalentDataProperties	0	0	0	0	0	0
DisjointDataProperties	0	0	0	0	0	0
FunctionalDataProperty	0	0	1	0	0	1
DataPropertyDomain	0	0	11	7	13	31
DataPropertyRange	0	0	10	8	13	31
Individual axioms						
ClassAssertion	21	0	407	7	0	435
ObjectPropertyAssertion	0	0	1007	0	0	1007
DataPropertyAssertion	0	0	282	2	0	284
NegativeObjectPropertyAssertion	0	0	0	0	0	0
NegativeDataPropertyAssertion	0	0	0	0	0	0
SameIndividual	0	0	0	0	0	0
DifferentIndividuals	1	0	0	0	0	1
Annotation axioms						
AnnotationAssertion	319	229	1315	410	50	2323
AnnotationPropertyDomain	0	0	0	0	0	0
AnnotationPropertyRangeOf	0	0	0	0	0	0

Table 3

Schema and Graph metrics comparison

	ExtruOnt	MaRCO	MASON	MSDL	SAREF4INMA
Schema metric					
Attribute richness	0.129921	0.535484	0.073171	0.007418	0.297297
Inheritance richness	2.531496	3.312903	1.199187	1.135015	1.810811
Relationship richness	0.255787	0.529115	0.111446	0.477816	0.309278
Attribute class ratio	0	0	0	0	0
Equivalence ratio	0.291339	0.009677	0	0.010386	0
Axiom/class ratio	24.192913	12.43871	5.926829	30.317507	9.081081
Inverse relations ratio	0.325758	0.011494	0.212766	0.152411	0.178571
Class/relation ratio	0.293981	0.142137	0.740964	0.460068	0.381443
Graph metric					
Absolute root cardinality	45	8	15	3	7
Absolute leaf cardinality	148	219	166	472	15
Absolute sibling cardinality	186	310	244	666	25
Absolute depth	478	1520	1385	5766	58
Average depth	2.489583	4.367816	5.54	8.479412	2.230769
Maximal depth	6	8	8	15	4
Absolute breadth	192	348	250	680	26
Average breadth	4.682927	3.702128	3.164557	3.4	2.363636
Maximal breadth	45	38	15	36	7
Ratio of leaf fan-outness	0.582677	0.706452	0.674797	0.700297	0.405405
Ratio of sibling fan-outness	0.732283	1	0.99187	0.988131	0.675676
Tangledness	0.153543	0.403226	0.113821	0.106825	0.216216
Total number of paths	192	348	250	680	26
Average number of paths	32.0	43.5	31.25	45.333333	6.5

Table 4

Summary of the OOPS! minor pitfalls for ExtruOnt

Code	**P02: Creating synonyms as classes.**
Description	Several classes whose identifiers are synonyms are created and defined as equivalent (owl:equivalentClass) in the same namespace.
Appears in	`http://www.ontology-of-units-of-measure.org/resource/om-2/CelsiusScale` `http://www.ontology-of-units-of-measure.org/resource/om-2/FahrenheitScale`
Code	**P04: Creating unconnected ontology elements.**
Description	Ontology elements (classes, object properties and datatype properties) are created isolated, with no relation to the rest of the ontology.

Appears in	`https://w3id.org/def/saref4inma#ProductionEquipment`
	`http://xmlns.com/foaf/0.1/Agent`
	`http://www.owl-ontologies.com/mason.owl#Machine-tool`
	`http://www.w3.org/2006/time#TemporalEntity`
	`http://purl.org/vocommons/voaf#Vocabulary`

not belong to the new machine. The main class `WireDrawingMachine`, which represents the new machine that we want to describe, should be defined as a subclass of `MASON:Machinetool` and `SAREF4INMA:ProductEquipment` to favour interoperability. The new components should be incorporated under the `owl:Thing` class in the *components4WidraOnt* module and linked to the *spatial4ExtruOnt* module as subclasses of `SpatialObject`. Moreover, the connections between the machine and its components should be made using the `hasPart` object property or new custom-made subproperties of `hasPart`. In this way, it is possible to describe the spatial and parthood relations between components of the new machine, for example:

```
prefix C4W: <http://bdi.si.ehu.es/bdi/
    ontologies/ExtruOnt/components4WidraOnt#>
prefix C4E: <http://bdi.si.ehu.es/bdi/
    ontologies/ExtruOnt/components4ExtruOnt#>
prefix po: <http://www.ontologydesignpatterns.
    org/cp/owl/partof.owl#>
prefix geo: <http://www.opengis.net/ont/
    geosparql#>

C4W:WireDrawingMachine po:hasPart C4W:Casptan,
                                  C4E:Motor.
C4W:Casptan geo:rcc8ec some C4E:Motor.
```

Which means that the wire drawing machine has the capstan and the motor as parts, and the capstan is externally connected to the motor. Finally, some other minor adaptations should be carried out regarding the linking of the new terms to the concepts defined in the other imported modules, as it was explained for *ExtruOnt*.

- *Clarity*: The custom terms defined in all modules of *ExtruOnt* contain non-ambiguous names, labels and comments facilitating the human readability and avoiding confusions and difficulty when the creation of individuals is carried out.
- *Completeness*: The *ExtruOnt* Ontology can answer all the competency questions specified in the ORSD document, representing correctly the domain for which it was created.
- *Efficiency*: Although the submodule extraction process from extensive ontologies such as OM and the utilization of specific terms in the context reduce the size of *ExtruOnt*, the reasoner execution time keeps too long when multiple extruders are described containing several data from sensors. However, the annotation and querying process can be carried out seamless.
  Conciseness: The knowledge contained in the modules *components4ExtruOnt* and *spatial4ExtruOnt* was retrieved from sources that are specific to the domains of extrusion and spatial relations respectively, thus avoiding irrelevant information. Moreover, for the remaining modules, submodules from OM, SOSA/SSN and 3DMO were extracted in the *Design* phase so that *ExtruOnt* incorporates only the concepts and descriptions from those ontologies that are relevant for our domain.
- *Consistency*: No inconsistencies were found in *ExtruOnt* when reasoning was performed. The reasoner used was Fact++.[26]

We did not evaluate the criterion of *Organizational fitness* because the ontology has not been deployed yet.

7. Conclusion and future work

The purpose of this paper is to present the *ExtruOnt* ontology, which contains terms to describe a type of manufacturing machine for performing extrusion processes (extruder). It is constituted by five modules: *components4ExtruOnt* for representing the components of an extruder, *spatial4ExtruOnt* for representing spatial relationships among those components, *OM4ExtruOnt* for representing the features of those components, *3D4ExtruOnt* for representing 3D models of the components, and *sensors4ExtruOnt* for representing the data captured by sensors. Although the *ExtruOnt* ontology is focused on extruders, it has been defined in such a way that it can be used as a model for describing other types of manufacturing machines by customizing or replacing some of its modules.

The descriptions contained in the *ExtruOnt* ontology will allow different types of users to familiarize themselves with the extrusion process, to interoperate with other manufacturing companies in an easy way, to create customized 3D images of extruder machines and an assisted exploration of data captured by sensors.

The *ExtruOnt* ontology has been documented and is available online. It has been evaluated according to two evaluation goals: domain coverage and quality of modeling, and has been assessed by humans and software artifacts. The evaluation shows that *ExtruOnt* can provide the answers to the competency questions defined, satisfying the proposed requirements and, therefore, proving that its modules are correctly developed.

[26] http://owl.man.ac.uk/factplusplus/

Furthermore, it is aligned with related ontologies, facilitating interoperability.

Finally, in addition to the necessary task of maintenance, we will mainly focus the future work on the development of two software artifacts whose core element will be the *ExtruOnt* ontology, in order to measure its performance in practical scenarios. The first artifact will be a Visual Query System, that will provide those advantages that we have mentioned through the paper to distinct types of users that work in the considered smart manufacturing scenario. The second artifact will be a recommender system that taking into account, on the one hand, the requirements of clients interested in buying an extruder machine and, on the other hand, the information described in the *ExtruOnt* ontology, will propose the most suitable extruder and the possible customizations that can be incorporated into it.

Acknowledgements

The authors would like to thank Urola Solutions for their help with information about the extrusion process and for providing real data. This research was funded by the Spanish Ministry of Economy and Competitiveness, grant number FEDER/TIN2016-78011-C4-2R. The work of Víctor Julio Ramírez-Durán is funded by the contract with reference BES-2017-081193.

References

[1] F. Ameri and D. Dutta, An upper ontology for manufacturing service description, in: *ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, 2006, pp. 651–661. doi:10.1115/DETC2006-99600.

[2] R. Barbau, S. Krima, R. Sudarsan, A. Narayanan, X. Fiorentini, S. Foufou and R.D. Sriram, OntoSTEP: Enriching product model data using ontologies, *Computer-Aided Design* **44**(6) (2012), 575–590. doi:10.1016/j.cad.2012.01.008.

[3] C. Brewster, H. Alani, S. Dasmahapatra and Y. Wilks, Data driven ontology evaluation, in: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, European Language Resources Association (ELRA), Lisbon, 2004, http://www.lrec-conf.org/proceedings/lrec2004/pdf/737.pdf.

[4] Q. Cao, C. Zanni-Merk and C. Reich, Ontologies for manufacturing process modeling: A survey, in: *International Conference on Sustainable Design and Manufacturing*, Springer, 2018, pp. 61–70. doi:10.1007/978-3-030-04290-5_7.

[5] A.G. Cohn, B. Bennett, J. Gooday and N.M. Gotts, Qualitative spatial representation and reasoning with the region connection calculus, *GeoInformatica* **1**(3) (1997), 275–316. doi:10.1023/A:1009712514511.

[6] L.M. Daniele, M. Punter, C. Brewster, R. García Castro, M. Poveda and A. Fernández, A SAREF Extension for Semantic Interoperability in the Industry and Manufacturing Domain, in: *Enterprise Interoperability: Smart Services and Business Impact of Enterprise Interoperability*, Wiley, 2018, pp. 201–207, Chapter 25. ISBN 9781119564034. doi:10.1002/9781119564034.ch25.

[7] L. Ding, T. Finin, A. Joshi, R. Pan, R.S. Cost, Y. Peng, P. Reddivari, V. Doshi and J. Sachs, Swoogle: A search and metadata engine for the Semantic Web, in: *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, CIKM '04*, ACM, New York, NY, 2004, pp. 652–659, http://swoogle.umbc.edu. ISBN 1-58113-874-1. doi:10.1145/1031171.1031289.

[8] EFFRA, Factories of the future roadmap, Technical report, EFFRA, 2019.

[9] ETSI, SmartM2M; smart applicances; reference ontology and oneM2M mapping, Technical specification, TS 103 264 V2.1.1, ETSI, 2017.

[10] A. Gangemi and V. Presutti, Ontology design patterns, in: *Handbook on Ontologies*, S. Staab and R. Studer, eds, Springer, Berlin, 2009, pp. 221–243. ISBN 978-3-540-92673-3. doi:10.1007/978-3-540-92673-3_10.

[11] M. Garetti and L. Fumagalli, P-PSO ontology for manufacturing systems, *IFAC Proceedings Volumes* **45**(6) (2012), 449–456. doi:10.3182/20120523-3-RO-2023.00222.

[12] H.F. Giles, E.M. Mount and J.R. Wagner, *Extrusion: The Definitive Processing Guide and Handbook*, Plastics Design Library, Elsevier, 2004. ISBN 9780815517115.

[13] F. Giustozzi, J. Saunier and C. Zanni-Merk, Context modeling for Industry 4.0: An ontology-based proposal, *Procedia Computer Science* **126** (2018), 675–684. doi:10.1016/j.procs.2018.08.001.

[14] M. Grüninger, Using the PSL Ontology, in: *Handbook on Ontologies*, S. Staab and R. Studer, eds, Springer, Berlin, 2009, pp. 423–443. ISBN 978-3-540-92673-3. doi:10.1007/978-3-540-92673-3_19.

[15] A. Haller, K. Janowicz, S.J.D. Cox, M. Lefrançois, K. Taylor, D. Le Phuoc, J. Lieberman, R. García-Castro, R. Atkinson and C. Stadler, The modular SSN ontology: A joint W3C and OGC standard specifying the semantics of sensors, observations, sampling, and actuation, *Semantic Web* **10**(1) (2018), 9–32. doi:10.3233/sw-180320.

[16] R. Hodgson and P.J. Keller, QUDT – Quantities, units, dimensions and data types in OWL and XML, 2011, http://www.qudt.org.

[17] M. Horridge, B. Parsia and U. Sattler, Explaining inconsistencies in OWL ontologies, in: *Scalable Uncertainty Management*, L. Godo and A. Pugliese, eds, Springer, Berlin, 2009, pp. 124–137. ISBN 978-3-642-04388-8. doi:10.1007/978-3-642-04388-8_11.

[18] E. Järvenpää, N. Siltala, O. Hylli and M. Lanz, The development of an ontology for describing the capabilities of manufacturing resources, *Journal of Intelligent Manufacturing* **30**(2) (2019), 959–978. doi:10.1007/s10845-018-1427-6.

[19] J.M. Keil and S. Schindler, Comparison and evaluation of ontologies for units of measurement, *Semantic Web* **10**(1) (2018), 33–51. doi:10.3233/sw-180310.

[20] S. Lemaignan, A. Siadat, J.-Y. Dantan and A. Semenenko, MASON: A proposal for an ontology of manufacturing domain, in: *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, 2006, pp. 195–200. doi:10.1109/DIS.2006.48.

[21] A. Lozano-Tello and A. Gomez-Perez, ONTOMETRIC: A method to choose the appropriate ontology, *Journal of Database Management* **15**(2) (2004), 1–18. doi:10.4018/jdm.2004040101.

[22] A. Maedche and S. Staab, Measuring similarity between ontologies, in: *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, A. Gómez-Pérez and V.R. Benjamins, eds, Springer, Berlin, 2002, pp. 251–263. ISBN 978-3-540-45810-4. doi:10.1007/3-540-45810-7_24.

[23] M.A. Musen, The Protégé project: A look back and a look forward, *AI Matters* **1**(4) (2015), 4–12. doi:10.1145/2757001.2757003.

[24] M. Perry and J. Herring, OGC GeoSPARQL – A geographic query language for RDF data, OGC implementation standard, 2012.

[25] H.S. Pinto, S. Staab and C. Tempich, DILIGENT: Towards a fine-grained methodology for distributed, loosely-controlled and evolving engineering of ontologies, in: *Proceedings of the 16th European Conference on Artificial Intelligence, ECAI'04*, IOS Press, Amsterdam, 2004, pp. 393–397. ISBN 978-1-58603-452-8.

[26] D.A. Randell, Z. Cui and A.G. Cohn, A spatial logic based on regions and connection, in: *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning, KR'92*, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1992, pp. 165–176. ISBN 1558602623.

[27] H. Rijgersberg, M. van Assem and J. Top, Ontology of units of measure and related concepts, *Semantic Web* **4**(1) (2013), 3–13. doi:10.3233/sw-2012-0069.

[28] M. Sabou and M. Fernandez, Ontology (network) evaluation, in: *Ontology Engineering in a Networked World*, M.C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta and A. Gangemi, eds, Springer, Berlin, 2012, pp. 193–212. ISBN 978-3-642-24794-1. doi:10.1007/978-3-642-24794-1_9.

[29] J. Sikora, Design of Extrusion heads, Department of Polymer Processing, Lublin University of Technology, 2008.

[30] L.F. Sikos, A novel ontology for 3D semantics: Ontology-based 3D model indexing and content-based video retrieval applied to the medical domain, *International Journal of Metadata, Semantics and Ontologies* **12**(1) (2017), 59–70. doi:10.1504/ijmso.2017.10008658.

[31] D. Šormaz and A. Sarkar, SIMPM – Upper-level ontology for manufacturing process plan network generation, *Robotics and Computer-Integrated Manufacturing* **55** (2019), 183–198. doi:10.1016/j.rcim.2018.04.002.

[32] M.C. Suárez-Figueroa, A. Gómez-Pérez and M. Fernández-López, The NeOn methodology for ontology engineering, in: *Ontology Engineering in a Networked World*, M.C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta and A. Gangemi, eds, Springer, Berlin, 2012, pp. 9–34. ISBN 978-3-642-24794-1. doi:10.1007/978-3-642-24794-1_2.

[33] Y. Sure, S. Staab and R. Studer, On-To-Knowledge Methodology (OTKM), in: *Handbook on Ontologies*, S. Staab and R. Studer, eds, Springer, Berlin, 2004, pp. 117–132. ISBN 978-3-540-24750-0. doi:10.1007/978-3-540-24750-0_6.

[34] Z. Usman, R.I.M. Young, N. Chungoora, C. Palmer, K. Case and J. Harding, A manufacturing core concepts ontology for product lifecycle interoperability, in: *Enterprise Interoperability*, M. van Sinderen and P. Johnson, eds, Springer, Berlin, 2011, pp. 5–18. ISBN 978-3-642-19680-5. doi:10.1007/978-3-642-19680-5_3.

[35] P.-Y. Vandenbussche, G.A. Atemezing, M. Poveda-Villalón and B. Vatant, Linked Open Vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web, *Semantic Web* **8**(3) (2016), 437–452. doi:10.3233/sw-160213.

[36] D. Vrandečić, Ontology evaluation, in: *Handbook on Ontologies*, S. Staab and R. Studer, eds, Springer, Berlin, 2009, pp. 293–313. ISBN 978-3-540-92673-3. doi:10.1007/978-3-540-92673-3_13.

7.2 A Semantic Approach for Big Data Exploration in Industry 4.0

Title: A Semantic Approach for Big Data Exploration in Industry 4.0

Authors: Idoia Berges, Víctor Julio Ramírez-Durán, Arantza Illarramendi

Journal: Big Data Research

Impact factor (2019): 2.673

Rank in category COMPUTER SCIENCE, THEORY & METHODS: 27/108 (Q1)

Publisher: Elsevier

Year: 2021

DOI: https://doi.org/10.1016/j.bdr.2021.100222

Big Data Research 25 (2021) 100222

Contents lists available at ScienceDirect

Big Data Research

www.elsevier.com/locate/bdr

A Semantic Approach for Big Data Exploration in Industry 4.0

Idoia Berges *, Víctor Julio Ramírez-Durán, Arantza Illarramendi

University of the Basque Country UPV/EHU, Donostia - San Sebastián, 20018, Spain

ARTICLE INFO

Article history:
Received 7 May 2020
Received in revised form 1 December 2020
Accepted 13 March 2021
Available online 15 April 2021

Keywords:
Data exploration
Industry 4.0
Ontologies

ABSTRACT

The growing trends in automation, Internet of Things, big data and cloud computing technologies have led to the fourth industrial revolution (Industry 4.0), where it is possible to visualize and identify patterns and insights, which results in a better understanding of the data and can improve the manufacturing process. However, many times, the task of data exploration results difficult for manufacturing experts because they might be interested in analyzing also data that does not appear in pre-designed visualizations and therefore they must be assisted by Information Technology experts.

In this paper, we present a proposal materialized in a semantic-based visual query system developed for a real Industry 4.0 scenario that allows domain experts to explore and visualize data in a friendly way. The main novelty of the system is the combined use that it makes of captured data that are semantically annotated first, and a 2D customized digital representation of a machine that is also linked with semantic descriptions. Those descriptions are expressed using terms of an ontology, where, among others, the sensors that are used to capture indicators about the performance of a machine that belongs to a Industry 4.0 scenario have been modeled. Moreover, this semantic description allows to: formulate queries at a higher level of abstraction, provide customized graphical visualizations of the results based on the format and nature of the data, and download enriched data enabling further types of analysis.

1. Introduction

We are witnessing a digital transformation era in which ubiquitous sensors and Internet of Things (IoT) devices enable the *datafication* of virtually any aspect of digitally connected individuals and machines [1]. Thus a new kind of economy that is based on data has emerged, coining the concept of data-driven economy [2]. One sector where data-driven economy is being implanted all over the world is the manufacturing industry. The fourth industrial revolution has given rise to what is called Smart Manufacturing, addressing the use of modern Information Technologies (IT) to transform the acquired data into manufacturing intelligence in order to achieve meaningful improvements in all aspects of manufacturing. The term Smart Manufacturing includes different initiatives, among which we can find “Smart Manufacturing” in USA, “Made in CHINA 2025”, “Future Manufacturing” in UK and “Industry 4.0” in Europe [3]. These initiatives enable important business opportunities for the manufacturers.

In general, the deployment of Smart Manufacturing approaches demands the introduction of data-related Information Technologies and digital platforms supporting it. Moreover, the design and implementation of such technologies and platforms faces diverse research and innovation challenges. These include, among others, improved methods of gathering valuable machine data and data integration across different sources of heterogeneous nature, implementation of advanced data analytics technologies and methods, and visualizing data to provide the right information, to the right person, at the right time.

While it is known that the data pre-processing and analytics phases consume most of the effort to be made in the whole data process, there is a remaining effort that must be devoted to the visualization and explanation of the results. One way to tackle this effort is to rely on efficient visualization metaphors and in smart visual interaction paradigms [4].

Regarding visualization, in [5] a survey of visualization technologies tailored for Smart Manufacturing scenarios can be found. Focusing on visual surveillance of all the captured raw data, many proposals rely on the use of a technology to compose observability dashboards[1] that allow experts in the manufacturing process to find relevant information in the data. Those types of technologies also support different data stores that provide SQL-like query language for interacting with stored data. However, many times,

* Corresponding author.
E-mail addresses: idoia.berges@ehu.eus (I. Berges), victorjulio.ramirez@ehu.eus (V.J. Ramírez-Durán), a.illarramendi@ehu.eus (A. Illarramendi).

https://doi.org/10.1016/j.bdr.2021.100222
2214-5796/

[1] https://logz.io/blog/grafana-vs-kibana/.

domain experts of those Smart Manufacturing scenarios, interested in analyzing data belonging to particular domains, which have specific meanings, have limited programming skills. As a result, they might find it difficult to analyze data that is not visualized in the dashboards.

Considering that limitation, in this paper we present an alternative that can be complementary to the use of dashboards, and which is based on the use of a semantic-based Visual Query System (VQS) tailored to Industry 4.0 scenarios.

In general, VQSs use a visual representation to depict the domain of interest and provide interaction mechanisms in order to formulate requests to the data stores by means of visual expressions [6].

The VQS that we propose provides a 2D digital representation of a manufacturing machine and dynamically customized forms to formulate queries. In this way domain experts can gain value and insights out of the captured data as rapidly as possible, minimizing the need to contact Information Technology experts. Both the digital representation and the forms are linked with semantic descriptions. The handling of semantic descriptions for data exploration and visualization tasks is what constitutes a novel technical contribution of the proposal. Those semantic descriptions are expressed using terms of different ontologies[2] such as the Semantic Sensor Network Ontology SOSA/SSN [7], the Ontology of units of Measure OM [8], the Semanticscience Integrated Ontology SIO[3] and of one ontology that we have built for an Industry 4.0 scenario, called ExtruOnt [9]: an ontology for describing a type of manufacturing machine. ExtruOnt imports terms from already developed ontologies and incorporates specific terms regarding the components of an extruder machine, their characteristics and spatial connections, and those related to sensors (e.g., `MotorConsumptionSensor`) used to capture observations about the performance of a machine (e.g., `DoubleValueObservation`) in order to favor interoperability issues. Although ExtruOnt focuses on extruders, it is straightforward to consider other types of machines due to the nature of the ontology (see section 3). The semantic descriptions incorporated in the ontology can be stored in a knowledge platform (e.g., Virtuoso). The proposed system provides the following benefits in the data exploration and visualization process:

1. **Possibility of querying the monitored data at a higher level of abstraction**. The system facilitates domain experts to formulate queries by operating on a 2D digital representation of a machine and then customizing them through forms. Those forms are dynamically generated by making use of stored semantic descriptions.
2. **Possibility of downloading semantically enriched data**. Domain experts of smart manufacturing scenarios can download, in an easy way, specific semantic descriptions related to domain data through a form. Using tools they already know (e.g., Excel) they have the possibility to perform new types of analyses with those semantic descriptions that they can not do dealing only with raw data.
3. **Possibility of incorporating on-the-fly semantic annotations in the visualization of results.** The results obtained for the queries are shown using tailored graphical representations customized according to the nature of the data domain. Moreover, they can be enriched with semantic description such as information related to outliers, incorporated to the raw data captured by sensors.
4. **Possibility of providing a customized visualization of a manufacturing machine**. Not all the machines of the same type (e.g., extruders) incorporate the same type of sensors. Visualizing a machine with its specific sensors, and thus, providing customized representations of machines is possible by consulting the semantic descriptions contained in the ontology

The system has been materialized for a real smart manufacturing scenario (in particular, a plastic bottle production factory that follows an extrusion process) and has been tested by domain experts. Thus, as an additional contribution we show the previously mentioned advantages located in this case study together with an empirical evaluation of the usability and of the performance in terms of required space to store the semantic descriptions and queries execution time when dealing with them.

A preliminary version of this work that outlines some basic concepts of our system was presented at the Eighth International Workshop on Modeling and Management of Big Data (MoBiD 2019) [10]. Here, the original workshop paper has been extended to include, among others, a new type of query of greater complexity, a download functionality, two additional visualization options and an empirical evaluation of the performance of the proposal in different storage solutions.

In the rest of this paper we present first some related works. Then, we introduce a background on the semantic descriptions considered in the paper. Next, we present the different resources for data exploration incorporated in the implemented system. Then, we perform an empirical evaluation about the implemented system and its behavior. Finally, we end with some conclusions.

2. Related work

In general, visual data explorations allow users to interactively explore the content of the data and identify interesting patterns that may be of their interest, in an autonomous way, without requiring assistance from Information Technology experts. Thus, Key Performance Indicator (KPI) dashboards have been actively used for this purpose in several domains, including manufacturing. For example, in [11] a digital control room that integrates multitouch and multiuser-based annotation dashboards for analyzing manufacturing data is presented. In [12] a calendar view is used to visualize and identify the issues and outliers that occur during the manufacturing process. In [13] a machine learning-based approach is used to support real-time analysis and visualization of sensor and ERP data. What these works have in common is that they directly show the information of a set of previously defined KPIs.

However, visualization systems must also offer customization capabilities to different user-defined exploration scenarios and preferences according to the analysis needs [4]. In this sense, Visual Query Systems already have a track record. They have been used for querying databases [6], for retrieving data from the Web [14] and also for visual exploration of time series. In this last case, there are approaches that advocate for the use of example-based methods such as [15], and [16], which proposes a multilevel map-based visualizations of geolocated time series. Different proposals can also be found among systems that deal with semantic data, such as SparqlFilterFlow [17], which employs a diagram-based approach to represent the queries, and Rhizomer [18], which employs a form based approach. Moreover, OptiqueVQS [19] is a semantic-based visual query system that exploits ontology projection techniques to enable graph-based navigation over an ontology during query construction and sampled data to enhance selection of data values for some data attributes. It shows all the classes defined in the loaded domain ontology to the users as a starting point for queries formulation. This forces domain experts to gather experience in the ontology before using the system. Other works such as

[2] Here the term *ontology* refers to the knowledge base composed of the conceptual level (i.e., axioms for classes and properties) and the instance level (i.e., assertions about individuals).

[3] https://bioportal.bioontology.org/ontologies/SIO.

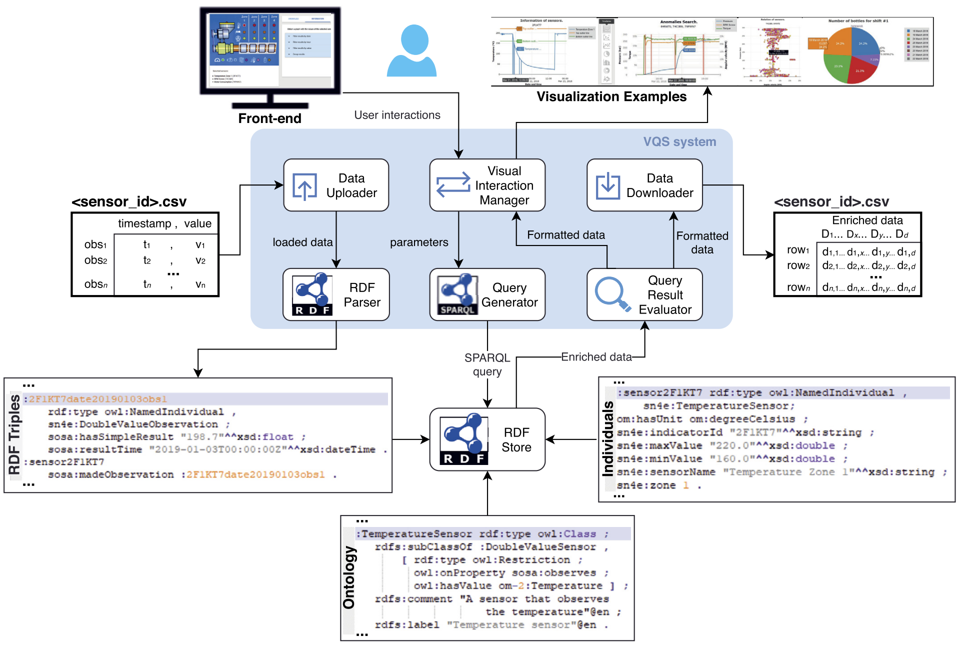


Fig. 1. Proposed VQS system overview.

[20] focus on tools such as OWLViz[4] or LODLive [21] for visualizing the content of ontologies and knowledge bases in the form of a graph but not for querying them, or on tools that allow querying but require a good knowledge of the underlying ontology (e.g., [22]).

Few proposals can be found in the literature that provide visual query systems for industrial scenarios and even less that provide semantic-based visual query systems. However, the incorporation of semantic techniques such as ontologies in the manufacturing field is being considered in different proposals due to the benefits that they bring to shift towards a Smart Manufacturing-oriented operation. Thus, ontologies are being defined and used in different manufacturing scenarios ([23,24]). For example, The P-PSO ontology [25] considers four aspects in the manufacturing domain: Product, Physical Aspect, Technological Aspect and Control Aspect. The ontology for Prognostics and Health Management of Machines [26] provides standardization of concepts and terms that are relevant to failure analysis in mechanical components. OntoSTEP [27] allows the description of product information mainly related to geometry.

The semantic based visual query system proposed in this paper (Fig. 1) considers the two aspects mentioned above: lack of visual query systems for smart manufacturing scenarios and interest of incorporating semantics techniques in those scenarios. For this reason, it makes a combined use of a manufacturing-oriented ontology and an interactive visual 2D digital representation of a machine to allow domain experts a smart data exploration process. The system manages a specific ontology called *ExtruOnt*[5] [9]. Due to a lack of sound descriptions of manufacturing machines that happen to be accessible, interoperable, and reusable we had to develop that ontology for providing detailed descriptions of a real manufacturing machine, in particular, an extruder.

3. Background on semantic descriptions

The I4TSMS [28] framework that we are developing aims to provide multiple services in Smart Manufacturing scenarios. Four main software components can be distinguished in its architecture (Fig. 2): The WebApp component, which allocates the Web Application front-end and provides users with a data exploration and visualization artifact; the Pre-Processing component, which is in charge of the captured time-series data preprocessing tasks; the Data and Knowledge Manager, which manages the data and knowledge storage requirements; and finally, the Web Service component, which manages the interconnection between the WebApp component and the Data and Knowledge Manager for providing the answers to the queries formulated by the users.

3.1. Ontology description

The Data and Knowledge Manager relies on a knowledge system to manage the semantic descriptions of the main components of the machine and its sensors. Regarding those semantic descriptions, *ExtruOnt* is composed of different modules which contain

4 https://github.com/protegeproject/owlviz.

5 http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/ExtruOnt.

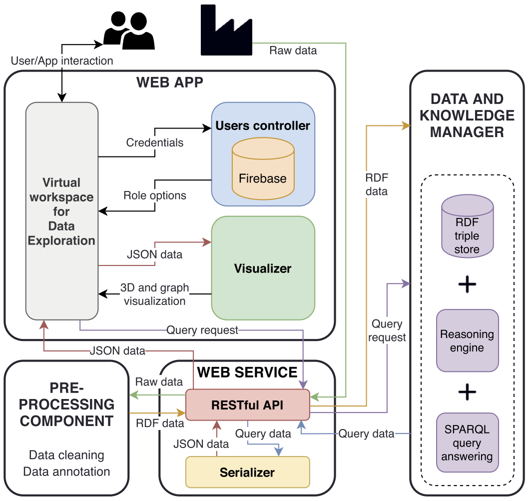


Fig. 2. Architecture of the I4TSMS web platform.

classes and properties for expressing descriptions about the features of the extruder, such as its components (*components4ExtruOnt*), characteristics (*OM4ExtruOnt*), spatial connections (*spatial4ExtruOnt*) and finally the sensors (*sensors4ExtruOnt*) used to capture indicators about the performance of that type of machine. Of those modules, only the module components4ExtruOnt is dependent on the type of machine. Thus, in order to adapt the approach to other machines, let's say a wire drawing machine, a new main ontology should be created (e.g., WidraOnt), without making major changes on the *OM4ExtruOnt*, *spatial4ExtruOnt*, *sensors4ExtruOnt* and *3D4ExtruOnt* modules (perhaps a new type of sensor, if needed), and only the *components4ExtruOnt* module would need to redefine.

In this paper, we mainly focus only on the *sensors4ExtruOnt* module, which contains both specific terms regarding sensors (e.g., `Pressure Sensor`) used to capture observations (e.g., `DoubleValueObservation`), and terms imported from other well-known ontologies, such as SOSA/SSN [7], OM [8] and SIO in order to favor its interoperability. We extend the classes `Sensor`, `Observation` and `ObservableProperty` of the SOSA/SSN ontology to describe more accurately the nuances of the sensors, observations and properties of our domain, and therefore allowing to provide the most suitable graphical representation.

The main class `Sensor` is a specialization of `sosa:Sensor` where two new properties have been included: `indicatorId` (to represent the identifier of the sensor) and `sensorName` (to indicate the name of the sensor). In our scenario, sensors capture either true/false data or numerical data. Thus, two subclasses of `Sensor` have been defined to represent this type of information: `BooleanSensor` and `DoubleValueSensor` respectively. Moreover, these subclasses have been further specialized in order to represent the specific sensors needed to monitor an extruder (see Table 1). Classes `ResistorOnOffSensor` and `FanOnOffSensor` are subclasses of `BooleanSensor`, while the remaining classes are subclasses of `DoubleValueSensor`.

For each type of sensor its observable property is indicated by the property `sosa:observes`. For example, the observable property of a `FeedRateSensor` is the `Speed` individual imported from OM. Moreover, for each subclass of `BooleanSensor` the true/false meaning of the observed property can be indicated by properties `meaningState1` and `meaningState0` respectively. For instance, the values of properties `meaningState1` and `meaningState0` in a `ResistorOnOffSensor` are "`On`" and "`Off`", which indicates that when the sensor registers the value *true* it means that the sensor is on and when the value is *false* the sensor is off. For each subclass of `DoubleValueSensor`, the expected range of the values captured by these sensors under normal conditions has been specified by properties `minValue` and `maxValue`. This range will help to identify possible outliers in the captured data. Other properties regarding capabilities of the sensors, such as the feasible measurement range captured by the sensor, are defined using properties from SOSA/SSN. The units of measure have been imported from OM and the graph and chart types from the SIO. Finally, for each type of sensor an axiom has been added which constrains the type of observations made by the sensor through property `sosa:madeObservation`. For each observation its value and timestamp are indicated by properties `sosa:hasSimpleResult` and `sosa:resultTime`. Two subclasses of `Observation` have been defined, depending on the type of value of the observation: `BooleanObservation` and `DoubleValueObservation`. In Fig. 3 an excerpt of the module can be found. For legibility reasons, only the most important properties and classes, and two specific types of sensors (`ResistorOnOffSensor` and `MotorConsumptionSensor`) are fully pictured.

3.2. Semantically enriched data

The real data used for testing the performance of the system have been provided by a Capital Equipment Manufacturer (CEM). This company has installed several sensors in the extruders that it manufactures. Those sensors register time-series data with a continuous measurement at 1 Hz (i.e., one measurement per second) of a variety of equipment setting parameters and physical magnitudes (temperatures, pressures, etc.) related to the raw materials,

Table 1
Example of the amount of data collected by the sensors of an average extruder machine in one year under normal conditions.

Sensor type/class	Description	Count	records[a]	Raw[b]	Virtuoso[b]	Stardog[b]	RDFox[c]	*Neo4j[b]*
ResistorOnOffSensor	Observes whether a resistor is on or off	4	126,144,048	3.41	14.79	32.87	52.89	*56.32*
FanOnOffSensor	Observes whether a fan is on or off	4	126,144,048	3.51	11.86	20.72	63.02	*57.15*
TemperatureSensor	Captures the temperature	10	315,360,120	9.06	32.15	51.79	157.55	*145.54*
MotorConsumptionSensor	Captures consumption of the motor	2	63,072,024	1.73	6.25	10.36	31.51	*30.17*
SpeedSensor	Captures the speed of the rotational parts	4	126,144,048	3.51	16.33	20.72	63.02	*54.13*
PressureSensor	Captures the pressure in the extruder	2	63,072,024	1.74	5.10	10.36	31.51	*28.22*
MeltingTemperatureSensor	Captures the melting temperature	2	63,072,024	1.65	7.74	10.36	31.51	*27.76*
BottlesPerShiftSensor	Captures the number of bottles in a shift	4	126,144,048	3.51	11.08	20.72	63.02	*54.70*
Others	Other sensors in the extruder	19	598,774,188	16.76	66.05	98.41	299.34	*270.33*
	Total	51	1,607,926,572	44.88	171.35	276.31	793.37	*724.32*

[a] One record per second.
[b] Disk size in gigabytes.
[c] Memory usage in gigabytes.

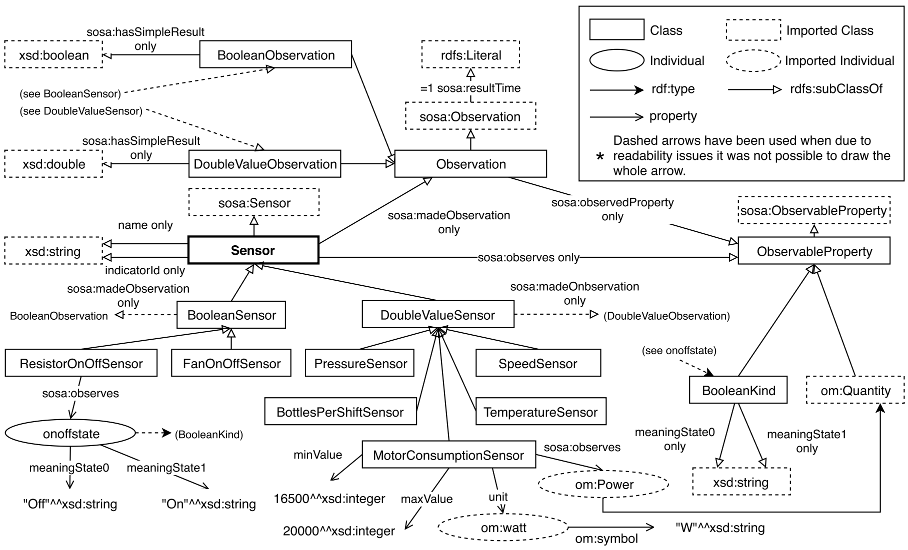


Fig. 3. Excerpt of the ontology showing the main classes and properties.

production processes and industrial equipment. In particular, the data came from an extruder machine from a plastic bottles production plant based on an extrusion process, on which the CEM has installed 51 sensors that generate time series of different types (see Table 1 in section 5.2.1). For the tests, a year's worth of data captured by those sensors has been gathered.

In order to provide an enhanced analysis of the data, the raw sensor data is semantically enriched with additional knowledge expressed in an ontology by means of RDF[6] annotations. Although semantic technologies were not created for Big Data, we have opted to use them in our proposal, as they provide several advantages such as their ability to describe and integrate heterogeneous data, and to infer new knowledge [29]. The captured data is annotated using the terms in the *ExtruOnt* ontology, converted to a set of RDF triples and stored in the RDF store by means of SPARQL insert queries. The original data contains a pair [value, timestamp] for each of the observations and, for each pair, an instance of `Observation` is created, indicating the value and its timestamp, along with the type of data, and the sensor. For example let us assume that sensor `sensor79PWN7` is an instance of class `MotorConsumptionSensor` (Fig. 4a) that has made the observa-

[6] https://www.w3.org/RDF/.

```
(a) :sensor79PWN7 rdf:type :MotorConsumptionSensor .

(b) :sensor79PWN7 sosa:madeObservation :obs1 .
    :obs1 sosa:hasSimpleResult "18710"^^xsd:double ;
          sosa:resultTime "2018-03-22T19:21:33.559Z"^^xsd:dateTime .

(c) :obs1 rdf:type :DoubleValueObservation .
    :sensor79PWN7 sosa:observes om:Power ;
                  :minValue "16500"^^xsd:double ;
                  :maxValue "20000"^^xsd:double ;
                  :unit om:watt .
    om:watt om:symbol "W"^^xsd:string .
```

Fig. 4. (a) Declaration of `sensor79PWN7` as an instance of `MotorConsumptionSensor`. (b) Example of the triples generated when annotating an observation. (c) Some of the triples that can be inferred.

tion [18710, 2018-03-22T19:21:33.559Z]. Then, the annotations in Fig. 4b are generated, where `obs1` refers to the newly created observation.

Moreover, due to the knowledge available in the ontology, additional information is now related to the observation (Fig. 4c). Since `obs1` was made by `MotorConsumptionSensor sensor79PWN7`, which is a `DoubleValueSensor` that makes `DoubleValueObservation`s, `obs1` can be classified as a `DoubleValueObservation`. Furthermore, due to the description of the sensor, it is possible to identify that: `obs1` is an observation of `om:Power`, the value of the observation complies with the expected range of values (between 16500 and 20000) for the observations of this type of sensor, the unit of the observation is `om:watt` and the symbol to represent watts is `"W"`.

4. Data exploration

VQSs allow users to analyze data by using a visual interface even if they only have basic technical skills. Thus, VQSs must be both expressive and usable [6].

We have developed an easy-to-use interface but which still allows for performing several types of queries. The user is presented with a dynamically generated picture of the extruder, which consists on a background image of the machine and a top layer where its sensors are placed. The background image is selected depending on the number of zones of the extruder (e.g., 4-zone-extruder, 5-zone-extruder), which can be obtained from the annotations of the machine in the ontology using a SPARQL query. For creating the top layer, another SPARQL query is asked to obtain information about the sensors that are deployed in the machine, along with their type and deployment data. Both SPARQL queries have been implemented within the system and are transparent to the user. Then, clickable bullets representing the sensors, as well as icons that specify their type, are placed in the aforementioned top layer. By using this approach it is possible to provide customized visualizations of multiple extruders. For example, in Fig. 5a the representation of a specific 4-zone-extruder is shown. In this case, 17 sensors for different indicators have been placed dynamically: four `TemperatureSensor`s, four `ResistorOnOffSensor`s, four `FanOnOffSensor`s, a `MotorRPMSensor` (a type of `SpeedSensor`), a `MotorConsumptionSensor`, a `PressureSensor`, a `MeltingTem-peratureSensor` and a `BottlesPerShiftSensor`.

Moreover, at the moment, the system allows for three different types of queries that have been selected in collaboration with domain experts and a download facility that allows one to obtain enriched data enabling thus additional types of analyses. Moreover, due to the modularized nature of the implementation, it could be easily extended to cover other kinds of queries in the future.

4.1. Information queries

Information queries are the most simple queries, used to ask for information about the observations of specific sensors. The user selects the sensors by clicking on them and inputs the desired constraints (e.g., date, hour, limits of values, aggregation functions) in a form that is dynamically generated depending on the characteristics of the sensors that have been selected. For example, if sensor 2F1KT7 is selected, the annotations made about it indicate that it is a `TemperatureSensor`, and due to a reasoning process that is also a `DoubleValueSensor`, meaning that it records numerical values. Thus, a slider is shown which allows to restrict the values of the retrieved information to the user's desired range. Moreover, since properties `minValue` and `maxValue` indicate that the usual range for that type of sensor is [160.0, 220.0] and that the unit is `om:degreeCelsius`, the slider has been customized so that values 160.0 °C and 220.0 °C are highlighted (see Fig. 5b), and its limits have been set to the feasible measurement range of the sensor, which in this case is [0.0, 250.0].

Likewise, if sensor URS001 is selected, the annotations indicate that it is a `ResistorOnOffSensor` (and therefore a `BooleanSensor`) and that the true/false values indicate whether the sensor is activated or not. This information is reflected by using an on/off switch. The simplicity of the used design helps users to formulate queries with a high level of abstraction. Once the selection has been made, a SPARQL query is generated and executed against the stored data. Fig. 6(a) shows an example of such a query.

4.2. Relation queries

Relation queries are used to ask for the observations made by some specific sensors when certain values hold in the observations made at the same timestamp by some other sensors. First, the user selects all the sensors that take part in the relation. Then they specify which are the sensors whose values they want to ask for, meaning that the remaining selected sensors are the ones whose values are fixed. The user indicates the fixed value for these sensors, which can be a numerical or boolean value (depending on the type of sensor), or the minimum, maximum or average value registered by the sensor in the specified time range. Once again, the form to create the queries is generated dynamically, based on the selected sensors and the information available about them in the ontology.

4.3. Anomalies queries

Anomalies queries indicate certain correlations between the values of different sensors that are supposed to hold under normal conditions. The system allows users to run customized or predefined anomalies queries. In order to create a customized anomaly query, after selecting the corresponding sensors the user must establish the correlations between those sensors. For example, one could establish that when the screw rotation speed increases, the pressure and the torque (which is an indicator related to the values of the `MotorConsumptionSensor`) increase as well (see Fig. 5c). Then, this information can be used to locate anomalies in the data (i.e., timestamps where the defined correlations did not

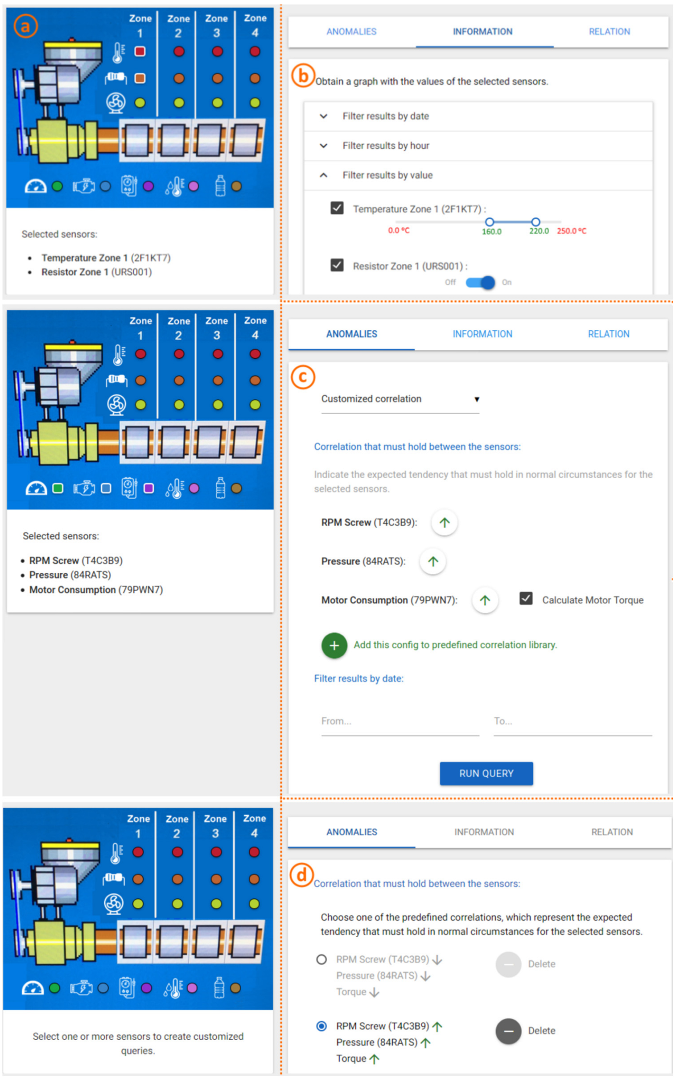


Fig. 5. User interface for querying about the data stored in the quad store.

hold). The system provides also support for saving the customized anomalies queries as predefined queries so that they can be used in future occasions (see Fig. 5d).

4.4. Data visualization

Once a query has been generated, it is executed against the repository of annotated data and the results are shown in form of a chart. In order to select the most suitable representation depending on the nature of the query and the sensors involved, a visualization module has been developed and imported in the sensors module of the ontology, where several recommendations for visualization have been described. As an alternative, an approach to automate data visualization in Big Data analytics but following a model-driven approach can be seen in [30].

```
prefix :<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Instances#>
prefix sosa: <http://www.w3.org/ns/sosa/>
prefix xsd: http://www.w3.org/2001/XMLSchema#
select ?resultValue ?resultTime
where {                                                    (a)
  :sensor2F1KT7 sosa:madeObservation ?obs .
  ?obs sosa:hasSimpleResult ?resultValue ;
    sosa:resultTime ?resultTime .
  filter(?resultValue >= "170"^^xsd:double && ?resultValue <= "200"^^xsd:double) .
  filter((xsd:dateTime(?resultTime) >= "2018-03-20T00:00:00.000Z"^^xsd:dateTime) &&
    (xsd:dateTime(?resultTime) <= "2018-03-22T23:59:59.999Z"^^xsd:dateTime))
} order by asc(?resultTime)
```

```
prefix :<http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Instances#>
prefix rdfs: < http://www.w3.org/2000/01/rdf-schema#>
prefix owl: < http://www.w3.org/2002/07/owl#>
select ?sensorType ?minValue
where {
  :sensor2F1KT7 rdf:type ?sensorType .
  ?sensorType rdfs:subClassOf                              (b)
    [rdf:type owl:Restriction;
              owl:onProperty :minValue ;
              owl:hasValue ?minValue] .
}
```

Fig. 6. SPARQL query examples: (a) SPARQL query that is generated when asking for the observations made by sensor 2F1KT7 between 20th and 22nd March 2018 within range 170 and 200. (b) SPARQL query to ask for the minimum value expected for the observations of sensor 2F1KT7.

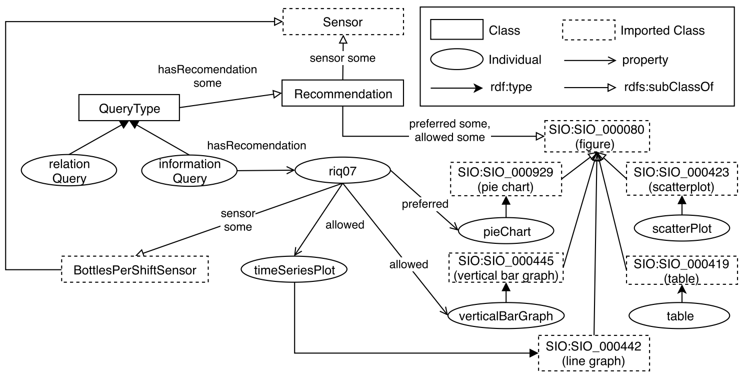


Fig. 7. Excerpt of the visualization module of the ontology.

The visualization module uses terms to describe the graph and chart types (e.g., pie chart, bar graph, scatter plot, line graph) from the Semanticscience Integrated Ontology (SIO). This ontology is a simple, integrated upper level ontology for consistent knowledge representation across physical, processual and informational entities. Moreover, we have incorporated additional chart types that were not found in SIO (e.g., gauge chart).

In Fig. 7 an excerpt of the module can be found, with one example of recommendation. More precisely, recommendation `riq07` indicates that when the query is an `information-Query` and the selected sensor is a `BottlesPerShift Sensor`, then the preferred chart is a pie chart (an instance of class `SIO:SIO_000929`), but vertical bar graphs and time-series plots (instances of classes `SIO:SIO_000445` and `SIO:SIO_000442`, respectively) are also allowed.

In Fig. 8a a time-series visualization is used for an information query about temperature sensor 2F1KT7. Top and bottom outlier lines indicate the expected maximum and minimum values for that sensor. The annotations made in the data and the descriptions in the ontology have been used to generate a semantically enriched customization of the chart, as noted in the second benefit explained in section 1. Since sensor 2F1KT7 is a `TemperatureSensor` that captures values in Celsius degrees, symbol °C is indicated. Moreover, the values for the outlier lines are obtained through SPARQL queries, such as the one in Fig. 6(b) that is used to query about the minimum expected value for that sensor. Fig. 8(b) shows the aforementioned pie chart for a sensor of type `BottlesPerShiftSensor`, indicating the number of bottles and percentage of the total made each day. In Fig. 8(c) a scatterplot has been used to show the relation between the values of two sensors, which measure the screw rotation speed and the pressure. Finally, in Fig. 8(d), a customized time series chart has been used to visualize the results of an `anomaliesQuery` that takes into account the torque, the screw rotation speed and the pressure. A reddish stripe is pictured wherever an anomaly has occurred in the expected trends of the recorded measures. It is also important to say that although our application proposes a visualization of results based on the preferred recommendations in the ontology, then the analyst can select other visualization mode among the allowed ones for that type of query and sensor if it better suits their interest (see Fig. 8(a)).

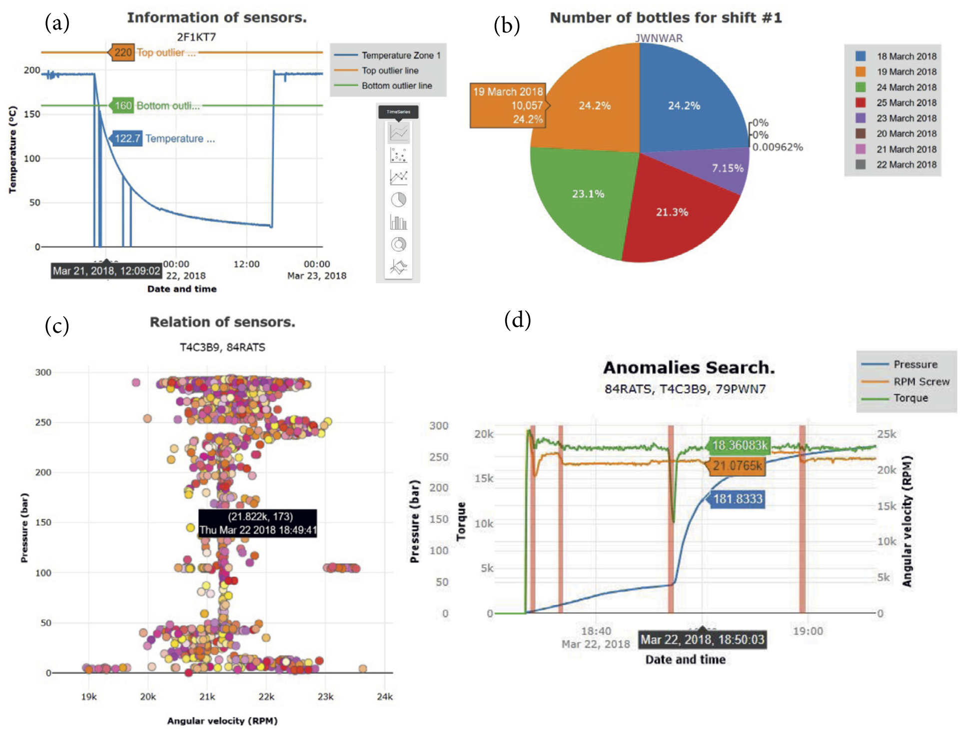


Fig. 8. Examples of visualizations of results. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

4.5. Download enriched data

The data download option is divided into several dynamically built sections with which the user is provided not only with the basic data of the sensors (timestamp, value, sensor id, etc.) but also with enriched data (sensor min/max operational values, machine info, data value conversions, etc.) thanks to the semantic descriptions annotated in the underlying ontology. For example, to convert the values of the zone 1 temperature sensor (2F1KT7) from degrees Celsius to Kelvin, the `om:hasFactor` and `om:hasOff-Set` properties of the individual `om:CelsiusScale` belonging to the class `om:IntervalScale` are used, which define the conversion factor (1.0) and the measurement offset (-273.15). In the same way, if a conversion from Kelvin to Fahrenheit is needed, the same properties defined in the individual `om:FahrenheitScale` (`om:hasFactor`=1.8 and `om:hasOff-Set`=-459.67) can be used. The different available scales are related by the dimension to which they correspond (property `om:hasDimension`) and the factor and offset values are based in a specific scale of that dimension. In this way, all the available scales for temperature are grouped in the dimension `om:thermodynamicTemperature-Dimension` and the factor and offset values are based on the scale `om:KelvinScale`. Moreover, those scales are obtained taking into account the unit of measurement set for zone 1 temperature sensor (`om:degreeCelsius`). The definitions of the conversions between measures are annotated by default in the ontology. Fig. 10 shows the annotations referring to the example described above.

Furthermore, the connections between the different modules of the ontology allow to link the sensors with the components of the extruder where they are located. Likewise, they allow to extract the characteristics related to these components, expanding the amount of enriched data that can be requested by the user. For example, the useful life and performance of the extruder motor in different shifts could be analyzed by comparing its technical specifications (power, speed, torque, etc.) with the values of the motor consumption sensor for the same shifts. All this information can be downloaded directly as shown in the *"Extruder component data related to selected sensors"* section of Fig. 9.

The data download option also includes aggregation functions (average, median, mode, count, sum, etc.) and a filtering by date range. The interface is presented in a simple way where the user selects the types of enriched data that they wish to obtain from a dynamically generated list depending on the sensors selected in the section on the left. Pressing the download button generates a CSV file with the requested information.

5. An empirical evaluation

In this section we present an empirical evaluation made in the context of the case study from the points of view of the usability and the behavior.

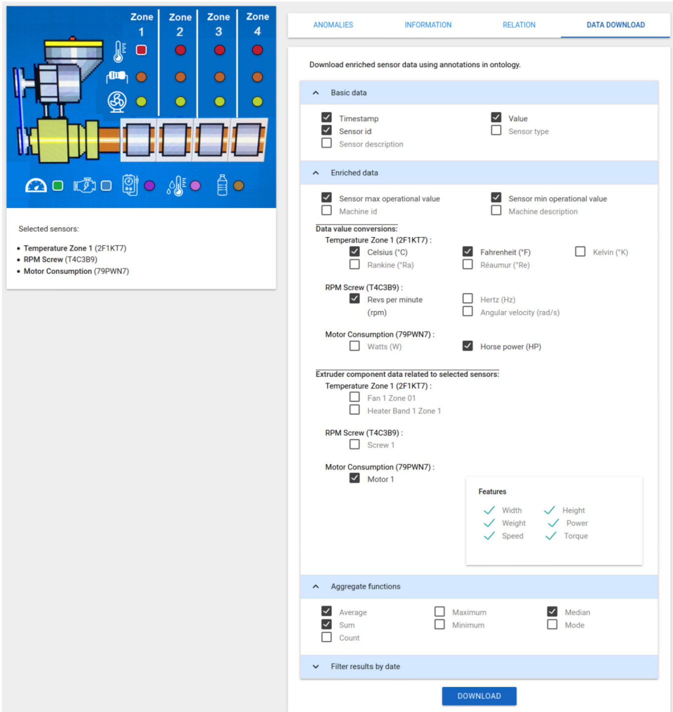


Fig. 9. User interface for downloading enriched data.

5.1. *Evaluation of the usability*

A qualitative usability evaluation of the system was performed. It involved two types of persons: 1) A R&D director of a company that develops machines that produce bottles based on an extrusion process, who we work closely with. This person also provides us real data captured from the machines developed by his company. 2) A director of an IBDS (Industrial Big Data Services) Provider company. An IBDS is an ITS (Information Technology Supplier) company that supplies manufacturers with the required technology and services to smartize their manufacturing businesses. Thus, IBDS Providers constitute a fundamental agent in industrial scenarios where there is an interest in adopting Smart Manufacturing approaches.

The users were introduced to the system and we asked them about the following main aspects: how intuitive was the system for performing the queries that they wanted, the grade of customization offered for each of the query types, the elapsed time between running the query and the presentation of the results, and the suitability of the visualization of the results. Moreover, we asked for any other additional suggestions.

They provided positive responses with regard to the intuitiveness of the system for performing queries, the customization of queries that can be achieved through the application, and speed and suitability of the visualization of the results. In addition to some minor changes that are already incorporated in the solution presented in this paper, they made the suggestion that providing the opportunity of choosing between a 3D representation of the extruder and the current 2D representation could enhance the solution, since some users might want to work with a more realistic representation of the machine. We are currently working on that 3D representation.

5.2. *Evaluation of the behavior*

As stated in section 3 the Data and Knowledge Manager must rely on a knowledge system to provide its functionalities. We evaluated the behavior of our solution in three different RDF stores and a NoSQL database. The RDF stores were selected by taking into account the performance results shown in [31] and [32]: Virtuoso[7] (version 08.03.3314), Stardog[8] (Version 7.1.1) and RDFox[9] (Version 2.1.1). The selected NoSQL database was Neo4j[10] (Version 4.1.3), due to its first position in the ranking presented in the DB-Engines website.[11] The hardware setup, where the RDF stores and the NoSQL database were evaluated, consisted of individual instances of m1-ultramem-40 virtual machines (40 Intel Skylake

[7] https://virtuoso.openlinksw.com/.
[8] https://www.stardog.com/.
[9] https://www.oxfordsemantic.tech/.
[10] https://neo4j.com/.
[11] https://db-engines.com/en/ranking/graph+dbms.

```
:sensor2F1KT7 rdf:type owl:NamedIndividual ,
            sn4e:TemperatureSensor;
      om:hasUnit om:degreeCelsius ;
      sn4e:indicatorId "2F1KT7"^^xsd:string ;
      sn4e:maxValue "220.0"^^xsd:double ;
      sn4e:minValue "160.0"^^xsd:double ;
      sn4e:sensorName "Temperature Zone 1"^^xsd:string ;
      sn4e:zone 1 .

om:CelsiusScale rdf:type owl:NamedIndividual ,
                         om:IntervalScale ;
                om:hasDimension om:thermodynamicTemperature-Dimension ;
                om:hasScale om:KelvinScale ;
                om:hasUnit om:degreeCelsius ;
                om:hasFactor "1.0"^^xsd:float ;
                om:hasOff-Set "-273.15"^^xsd:float .

om:FahrenheitScale rdf:type owl:NamedIndividual ,
                            om:IntervalScale ;
                   om:hasDimension om:thermodynamicTemperature-Dimension ;
                   om:hasScale om:KelvinScale ;
                   om:hasUnit om:degreeFahrenheit ;
                   om:hasFactor "1.8"^^xsd:float ;
                   om:hasOff-Set "-459.67"^^xsd:float .

om:RankineScale rdf:type owl:NamedIndividual ,
                         om:RatioScale ;
                om:hasDimension om:thermodynamicTemperature-Dimension ;
                om:hasScale om:KelvinScale ;
                om:hasUnit om:degreeRankine ;
                om:hasFactor "1.8"^^xsd:float ;
                om:hasOff-Set "0.0"^^xsd:float .

om:ReaumurScale rdf:type owl:NamedIndividual ,
                         om:IntervalScale ;
                om:hasDimension om:thermodynamicTemperature-Dimension ;
                om:hasScale om:KelvinScale ;
                om:hasUnit om:degreeReaumur ;
                om:hasFactor "1.0"^^xsd:float ;
                om:hasOff-Set "-218.52"^^xsd:float .
```

Fig. 10. Excerpt of the annotations for data value conversions.

virtual CPUs, 961 GB of RAM and SSD with 1200 GB of storage capacity) from the Google Cloud Compute Engine platform. Next some of the evaluation results are presented.

5.2.1. Data storage space

Table 1 presents the results of the required storage space for the time-series data captured during one year. With regard to the RDF stores, Virtuoso makes a better space management, decreasing the space needed to store the series by 37.99% and 78.4% compared to Stardog and RDFox, respectively. It is worth mentioning that RDFox is an in-memory RDF triple store, therefore, it uses the Random Access Memory to store the data instead of disk space. This feature makes the data loading faster than the other tested RDF stores but penalizes the amount of memory used due to poor compression.

In the case of the NoSQL database Neo4j, the storage space management does not outperform Virtuoso either. It had been selected for the empirical evaluation because it also supports the concept of relationship. In this way, it is possible to represent an RDF triple as the relationship between two nodes. However, the way in which the RDF triples were loaded into Neo4j differs from the way used for RDF stores. While SPARQL inserts are used for the latter, the Neo4j RDF & Semantics toolkit (n10s[12]) is used for the former. n10s is a plugin that enables the use of RDF in Neo4j and can be used to import existing RDF datasets, build integrations with RDF generating endpoints or easily construct RDF endpoints on Neo4j.

[12] https://neo4j.com/labs/neosemantics/.

The main drawback in Neo4j is that there not exists an underlying ontology and after the transformation of the RDF data into graph data, all the RDF, RDFS, and OWL tags lose their semantic meaning becoming just simple labels, preventing the use of reasoning with them.

5.2.2. Query response times

Fig. 11 shows the response time for the three types of queries presented in section 4.

Information Queries. The query response time evaluation for information queries is shown in Fig. 11(a) using the three different RDF stores named previously and three different time windows. The queries were executed 10 times each and the average value was calculated. As it can be seen, on the one hand, Virtuoso is the fastest RDF store for the three different time windows, even faster than the in-memory RDFox store. On the other hand, Stardog presents a low performance solving this type of queries with a considerable distance from the others.

Relation Queries. In Fig. 11(b) an evaluation of the query response time for this type of queries is presented. RDFox is the fastest RDF store for those queries with a time window of a day and a week. However, Virtuoso presents a better performance for those relation queries with a longer time window. Also, the query response time in Virtuoso for shorter time windows remains constant (4.9 seconds for a day and a week time window). Stardog continues to show poor performance with very high query response times.

Anomalies Queries. This type of query demands a different approach with respect to the other ones. In this case, it is necessary to ask about the behavior of the values over time,

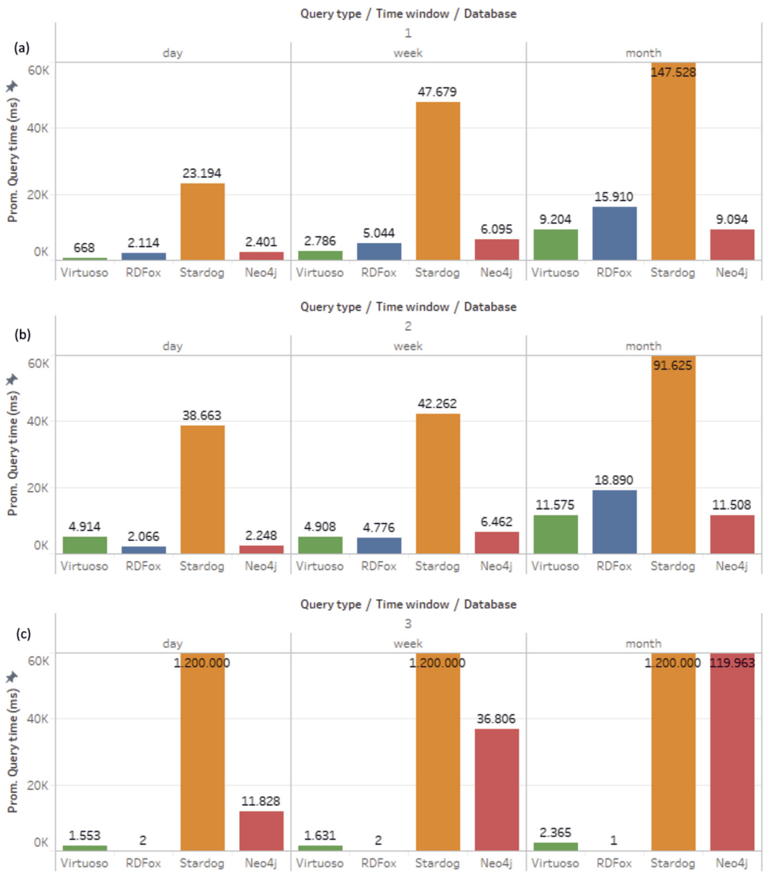


Fig. 11. Query response time for: (a) information queries; (b) relation queries; (c) anomalies queries.

detecting increases and decreases in those values. In time series, the observations are taken with a certain periodicity, and the values between one observation and the next may fluctuate depending on the sensitivity, precision and tolerance of the sensor, in addition to the noise generated by external factors. Thus, to correctly detect increases and decreases, it is necessary to establish a minimum variation in the values of the observations for a period of time, so that these increases and decreases are not the product of errors generated by the nature of the sensor or external factors. These minimal variations depend on the kind of observation to be evaluated. For example, the power consumption of a motor can vary between 16,500 and 20,000 watts in contrast with the temperature of an extruder band heater, which can vary between 160 and 220 degrees Celsius. This means that oscillations are greater in the observations of the power consumption than in the temperature, needing to define independent minimum variations for each type of observation.

The detection of increases and decreases in values can be done by applying rules, which is one of the advantages provided by RDF stores, creating an additional RDF triple to indicate whether in a period of time there is an increase or decrease in value and limiting the query to ask only for the behavior of those triples. However, each RDF store supports a specific rule language and evaluation method, i.e., RDFox supports the use of Datalog rules with the materialization of inferred triples; Virtuoso supports the creation of custom inference rules using the SPIN vocabulary; and Stardog supports two different syntaxes for defining rules, the first is native Stardog Rules syntax (based on SPARQL) and the second is the standard RDF/XML syntax for SWRL. Virtuoso and Stardog use the query rewriting method to evaluate the rules. Fig. 12 shows an example of the syntax used for a rule in RDFox.

In the query response time evaluation presented in Fig. 11(c), it can be seen that the materialization method for rule application used in RDFox provides a remarkable performance compared to Virtuoso and Stardog. Nevertheless, Virtuoso shows a notewor-

```
demo:valueIncrement[?sensor,?actDate] :-
    eo:MotorRPMSensor[?sensor], sosa:madeObservation[?sensor,?obs1],
    sosa:resultTime[?obs1,?actDate], demo:hasAVGResult[?obs1,?actVal],
    BIND (?actDate + "-PT15M"^^xsd:duration AS ?pasDate),
    sosa:madeObservation[?sensor,?obs2],sosa:resultTime[?obs2,?pasDate],
    demo:hasAVGResult[?obs2,?pasVal], FILTER(?actVal > ?pasVal+500) .
```

Fig. 12. Datalog rule to materialize an increase in the value of observations for all motor RPM sensors when the increase is greater than 500 units in a time span of 15 minutes.

thy performance even using the query rewriting method. Finally, queries made in Stardog exceeded the maximum waiting time of two minutes.

Regarding the performance of Neo4j to solve the three types of defined queries, it can be observed that the response times for information and relation queries are similar to those obtained with the best performing RDF stores (i.e., Virtuoso and RDFox). However, for anomaly queries, the query response time increases to about 2 minutes as, to the best of our knowledge, it is not possible to use rules in Neo4j. Therefore, it is necessary to first query the basic data and then apply some post-processing using an external framework (e.g., Spark,[13] Hadoop[14]) to detect anomalies.

6. Conclusion

The development of software tools that support customization capabilities that facilitate data exploration and visualization, to different users according to their analysis needs, is a challenge that is being considered in manufacturing scenarios. Exploration and visualization of captured time-series data provides increasing knowledge about the indicators used in the monitoring of machines. In this paper we have presented a semantic-based visual query system that enables domain experts to formulate queries dealing with a customized digital representation of the machine and on-the-fly generated forms. The system also offers the capability of downloading enriched data. Moreover, it provides a tailor-made visualization of the results depending on their nature. The whole process is supported by an underlying ontology where the main components of the machine and its sensors have been described.

Although disk space is a crucial concern when referring to big data scenarios, the benefits of semantic data annotation for data analysis purposes are not comparable with the limited knowledge extracted directly from raw data. Therefore, the increase in storage due to semantic annotation is acceptable and moreover, the query response times obtained are manageable.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors would like to thank Urola Solutions for their help with information about the extrusion process and for providing real data. This research was funded by the Spanish Ministry of Economy and Competitiveness under Grant No. FEDER/TIN2016-78011-C4-2-R and the Basque Government under Grant No. IT1330-19. The work of Víctor Julio Ramírez is funded by the Spanish Ministry of Economy and Competitiveness under contract with reference BES-2017-081193.

[13] https://spark.apache.org/.
[14] https://hadoop.apache.org/.

References

[1] D. Ciuriak, The economics of data: implications for the data-driven economy, in: Data Governance in the Digital Age, Centre for International Governance Innovation, 2018, Ch. 8.

[2] European Commission, Towards a thriving data-driven economy, Tech. rep., 2014, https://ec.europa.eu/digital-singlemarket/news/communication-data-driven-economy.

[3] A. Kusiak, Smart manufacturing, Int. J. Prod. Res. 56 (1–2) (2018) 508–517, https://doi.org/10.1080/00207543.2017.1351644.

[4] N. Bikakis, G. Papastefanatos, O. Papaemmanouil, Big data exploration, visualization and analytics, Big Data Res. 18 (2019) 100123, https://doi.org/10.1016/j.bdr.2019.100123.

[5] F. Zhou, X. Lin, C. Liu, Y. Zhao, P. Xu, L. Ren, T. Xue, L. Ren, A survey of visualization for smart manufacturing, J. Vis. 22 (2) (2019) 419–435, https://doi.org/10.1007/s12650-018-0530-2.

[6] T. Catarci, M.F. Costabile, S. Levialdi, C. Batini, Visual query systems for databases: a survey, J. Vis. Lang. Comput. 8 (2) (1997) 215–260, https://doi.org/10.1006/jvlc.1997.0037.

[7] A. Haller, K. Janowicz, S.J. Cox, M. Lefrançois, K. Taylor, D. Le Phuoc, J. Lieberman, R. García-Castro, R. Atkinson, C. Stadler, The modular SSN ontology: a joint W3C and OGC standard specifying the semantics of sensors, observations, sampling, and actuation, Semant. Web 10 (1) (2019) 9–32, https://doi.org/10.3233/SW-180320.

[8] H. Rijgersberg, M. van Assem, J. Top, Ontology of units of measure and related concepts, Semant. Web 4 (1) (2013) 3–13, https://doi.org/10.3233/SW-2012-0069.

[9] V.J. Ramírez-Durán, I. Berges, A. Illarramendi, ExtruOnt: an ontology for describing a type of manufacturing machine for Industry 4.0 systems, Semant. Web 11 (6) (2020) 887–909, https://doi.org/10.3233/SW-200376.

[10] I. Berges, V.J. Ramírez-Durán, A. Illarramendi, Facilitating data exploration in Industry 4.0, in: G. Guizzardi, F. Gailly, R.S.P. Maciel (Eds.), Advances in Conceptual Modeling – Proceedings of ER 2019 Workshops FAIR, MREBA, EmpER, MoBiD, OntoCom, and ER Doctoral Symposium Papers, Salvador, Brazil, November 4-7, 2019, in: Lecture Notes in Computer Science, vol. 11787, Springer, 2019, pp. 125–134.

[11] J. Rubart, B. Lietzau, P. Söhlke, Analyzing manufacturing data in a digital control room making use of semantic annotations, in: IEEE 14th International Conference on Semantic Computing, IEEE, ICSC 2020, San Diego, CA, USA, February 3-5, 2020, pp. 434–438.

[12] D. Chankhihort, S.S. Choi, G.J. Lee, B.M. Im, D. Ahn, E. Choi, A. Nasridinov, S. Kwon, S. Lee, J. Kang, K. Park, K. Yoo, Integrative manufacturing data visualization using calendar view map, in: Eighth International Conference on Ubiquitous and Future Networks, ICUFN 2016, Vienna, Austria, July 5-8, 2016, IEEE, 2016, pp. 114–116.

[13] N. Iftikhar, B.P. Lachowicz, A. Madarasz, F.E. Nordbjerg, T. Baattrup-Andersen, K. Jeppesen, Real-time visualization of sensor data in smart manufacturing using lambda architecture, in: S. Hammoudi, C. Quix, J. Bernardino (Eds.), Proceedings of the 9th International Conference on Data Science, Technology and Applications, DATA 2020, Lieusaint, Paris, France, July 7-9, 2020, SciTePress, 2020, pp. 215–222.

[14] J. Lloret-Gazo, A survey on visual query systems in the web era (extended version), CoRR, arXiv:1708.00192 [abs].

[15] B. Eravci, H. Ferhatosmanoglu, Diversity based relevance feedback for time series search, Proc. VLDB Endow. 7 (2013) 109–120, https://doi.org/10.14778/2732228.2732230.

[16] G. Chatzigeorgakidis, K. Patroumpas, D. Skoutas, S. Athanasiou, S. Skiadopoulos, Visual exploration of geolocated time series with hybrid indexing, Big Data Res. 15 (2019), https://doi.org/10.1016/j.bdr.2019.02.001.

[17] F. Haag, S. Lohmann, S. Bold, T. Ertl, Visual SPARQL querying based on extended filter/flow graphs, in: Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces, ACM, 2014, pp. 305–312.

[18] J.M. Brunetti, R. García, S. Auer, From overview to facets and pivoting for interactive exploration of semantic web data, Int. J. Semantic Web Inf. Syst. 9 (1) (2013) 1–20, https://doi.org/10.4018/jswis.2013010101.

[19] A. Soylu, E. Kharlamov, D. Zheleznyakov, E. Jiménez-Ruiz, M. Giese, M.G. Skjæveland, D. Hovland, R. Schlatte, S. Brandt, H. Lie, I. Horrocks, OptiqueVQS: a visual query system over ontologies for industry, Semant. Web 9 (5) (2018) 627–660, https://doi.org/10.3233/SW-180293.

[20] F. Antoniazzi, F. Viola, RDF graph visualization tools: a survey, in: 23rd Conference of Open Innovations Association, FRUCT 2018, Bologna, Italy, November 13-16, 2018, IEEE, 2018, pp. 25–36.

[21] D.V. Camarda, S. Mazzini, A. Antonuccio, LodLive, exploring the Web of data, in: V. Presutti, H.S. Pinto (Eds.), I-SEMANTICS 2012 – 8th International Conference on Semantic Systems, I-SEMANTICS '12, Graz, Austria, September 5-7, 2012, ACM, 2012, pp. 197–200.

[22] S. Lohmann, S. Negru, F. Haag, T. Ertl, Visualizing ontologies with VOWL, Semant. Web 7 (2016) 399–419, https://doi.org/10.3233/SW-150200.

[23] E. Kharlamov, B.C. Grau, E. Jiménez-Ruiz, S. Lamparter, G. Mehdi, M. Ringsquandl, Y. Nenov, S. Grimm, M. Roshchin, I. Horrocks, Capturing industrial information models with ontologies and constraints, in: The Semantic Web - ISWC 2016 – Proceedings of 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Part II, 2016, pp. 325–343.

[24] E. Negri, L. Fumagalli, M. Garetti, L. Tanca, Requirements and languages for the semantic representation of manufacturing systems, Comput. Ind. 81 (C) (2016) 55–66, https://doi.org/10.1016/j.compind.2015.10.009.

[25] M. Garetti, L. Fumagalli, P-PSO ontology for manufacturing systems, in: 14th IFAC Symposium on Information Control Problems in Manufacturing, IFAC Proc. Vol. 45 (6) (2012) 449–456, https://doi.org/10.3182/20120523-3-RO-2023.00222.

[26] D.L. Nuñez, M. Borsato, An ontology-based model for prognostics and health management of machines, J. Ind. Inf. Integr. 6 (2017) 33–46, https://doi.org/10.1016/j.jii.2017.02.006.

[27] R. Barbau, S. Krima, R. Sudarsan, A. Narayanan, X. Fiorentini, S. Foufou, R.D. Sriram, OntoSTEP: enriching product model data using ontologies, Comput. Aided Des. 44 (6) (2012) 575–590, https://doi.org/10.1016/j.cad.2012.01.008.

[28] K. Villalobos, I. Berges, B. Diez, A. Goñi, A. Illarramendi, A multi-services architecture for smart manufacturing scenarios, in: International Conference on Industrial Internet of Things and Smart Manufacturing, Imperial College London, London, United Kingdom, 2018.

[29] K. Thirunarayan, A.P. Sheth, Semantics-empowered big data processing with applications, AI Mag. 36 (1) (2015) 39–54, https://doi.org/10.1609/aimag.v36i1.2566.

[30] M. Golfarelli, S. Rizzi, A model-driven approach to automate data visualization in big data analytics, Inf. Vis. 19 (1) (2019) 24–47, https://doi.org/10.1177/1473871619858933.

[31] A. Addlesee, Comparison of linked data triplestores: developing the methodology, https://medium.com/wallscope/comparison-of-linked-data-triplestores-developing-the-methodology-e87771cb3011, 2019.

[32] A. Addlesee, Comparison of linked data triplestores: a new contender, https://medium.com/wallscope/comparison-of-linked-data-triplestores-a-new-contender-c62ae04901d3, 2019.

7.3 Towards the implementation of Industry 4.0: a methodology based approach oriented to the customer life cycle

Title: Towards the implementation of Industry 4.0: a methodology-based approach oriented to the customer life cycle

Authors: Víctor Julio Ramírez-Durán, Idoia Berges, Arantza Illarramendi

Journal: Computers in Industry

Impact factor (2019): 3.954

Rank in category COMPUTER SCIENCE, INTERDISCIPLINARY APPLICATIONS: 21/109 (Q1)

Publisher: Elsevier

Year: 2021

DOI: https://doi.org/10.1016/j.compind.2021.103403

Computers in Industry 126 (2021) 103403

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.elsevier.com/locate/compind

Towards the implementation of Industry 4.0: A methodology-based approach oriented to the customer life cycle

Víctor Julio Ramírez-Durán*, Idoia Berges, Arantza Illarramendi

Department of Languages and Information Systems, University of the Basque Country UPV/EHU, Donostia - San Sebastián 20018, Spain

ARTICLE INFO

Article history:
Received 30 May 2020
Received in revised form
10 November 2020
Accepted 15 January 2021
Available online 29 January 2021

Keywords:
Industry 4.0
Customer life cycle
CAD
Ontology

ABSTRACT

Many different worldwide initiatives are promoting the transformation from machine dominant manufacturing to digital manufacturing. Thus, to achieve a successful transformation to Industry 4.0 standard, manufacturing enterprises are required to implement a clear roadmap. However, Small and Medium Manufacturing Enterprises (SMEs) encounter many barriers and difficulties (economical, technical, cultural, etc.) in the implementation of Industry 4.0. Although several works deal with the incorporation of Industry 4.0 technologies in the area of the product and supply chain life cycles, which SMEs could use as reference, this is not the case for the customer life cycle. Thus, we present two contributions that can help the software engineers of those SMEs to incorporate Industry 4.0 technologies in the context of the customer life cycle. The first contribution is a methodology that can help those software engineers in the task of creating new software services, aligned with Industry 4.0, that allow to change how customers interact with enterprises and the experiences they have while interacting with them. The methodology details a set of stages that are divided into phases which in turn are made up of activities. It places special emphasis on the incorporation of semantics descriptions and 3D visualization in the implementation of those new services. The second contribution is a system developed for a real manufacturing scenario, using the proposed methodology, which allows to observe the possibilities that this kind of systems can offer to SMEs in two phases of the customer life cycle: Discover & Shop, and Use & Service.

1. Introduction

The instantiation of the data-driven economy in the manufacturing industry has led to the development of different initiatives and strategies addressing the use of data exploitation to optimize and transform the manufacturing business. Among those initiatives we can find “Smart Manufacturing” in USA, “Made in CHINA 2025″, “Future Manufacturing” in UK and “Industry 4.0” in Europe (Kusiak, 2018), which enable important business opportunities for the manufacturers.

Considering the Industry 4.0 in particular (Öztemel and Gursev, 2020), this initiative can have the greatest impact in three key areas regardless of the sector (Cotteleer and Sniderman, 2017): Products, supply chain, and customers. Regarding products, technologies such as sensors, machine learning or robotics can transform the way products are designed and developed, and the exploitation of data about the products can allow manufacturers to predict, plan and manage specific circumstances in order to optimize their production. Furthermore, new business models can be contemplated by selling data and services in addition to physical products. In the case of the supply chain, advanced forecasting techniques relying on internal (e.g. demand) and external (e.g. market trends) data can allow for a faster delivery time (Alicke et al., 2016). Moreover, real-time information about the supply network and the logistics capabilities can allow for a more flexible planning and inventory processes, which can react to changing demand or supply situations. Finally, regarding customers, Industry 4.0 technologies can help to gain a better understanding of the customers, enhance their experience when interacting with the products and enable better post-sale support. The proposal presented in this paper considers this last area.

However, what is observed is that most of research addressing implementation techniques in Industry 4.0 scenarios is created for large organizations and the majority of Small and Medium Enterprises (SMEs) are overwhelmed by the large amount of existing 4.0 technologies, the time and effort needed to learn them, and the costs of their implementation (Masood and Sonntag, 2020). Moreover, while several works deal with the incorporation of Industry 4.0 in the area of the products (e.g. Tomic et al., 2020; Mourtzis et al., 2018) and the area of the supply chain (e.g. the survey in Winkelhaus and Grosse (2020)), literature on how to introduce

* Corresponding author.
E-mail addresses: victorjulio.ramirez@ehu.eus (V.J. Ramírez-Durán), idoia.berges@ehu.eus (I. Berges), a.illarramendi@ehu.eus (A. Illarramendi).

https://doi.org/10.1016/j.compind.2021.103403

Industry 4.0 technologies in the area of the customers is rather scarce. The proposal shown in this paper considers those two drawbacks: the difficulties of SMEs to achieve a successful transformation towards Industry 4.0 standard and the scare literature in the area of customer life cycle, and thus it presents a new methodology which can help software engineers of SME manufacturing scenarios to incorporate Industry 4.0 technologies in the context of the customer life cycle. The goal pursued with this incorporation is to improve efficiency and enhance customer experiences, thus helping manufactures to attract and retain customers.

The methodology consists of stages that are divided into phases which in turn are made up of activities. It uses similar phases of those considered by a typical life cycle for the development of an information system (Elmasri and Navathe, 2010), which are also used in well-known development models such as Waterfall, Iterative or Agile. But what makes it different and can be considered the main of our contributions are the detailed descriptions of the activities that must be carried out in each phase. Furthermore, one technical novelty of the proposed methodology is the commitment made in it to connect semantic based technology, used to represent knowledge related to manufactured products, with 3D digital technology, used to visualize those products. Within semantic technologies, ontologies allow the representation of knowledge of a particular domain through the definition of categories, properties and relationships between concepts and entities in a way that is understandable by both machines and humans. The use of ontologies has been increasing over the years, making the knowledge represented through them much broader, facilitating the definition of new concepts through reuse and improving interoperability between heterogeneous information systems. Furthermore, 3D rendering technologies allow detailed visualization of objects by adding navigation components that enhance the user experience. These technologies have evolved in such a way that those processes that some years ago took several minutes and needed a powerful system to be executed, nowadays can be carried out in milliseconds from the web browser of a conventional computer or from a mobile device (e.g. smartphones, tablets, laptops, etc.). We believe that an accurate and detailed description of the products or services offered by the manufacturing industry together with an improved visual presentation, both enhanced by the use of these technologies, will enrich the experience offered to customers, positively affecting decision-making.

Although there exist other proposals in the smart manufacturing scenarios that advocate for the use of those technologies separately (e.g. in (Kharlamov et al., 2019) authors propose a semantic rule language for industrial internet of things and in (van Lopik et al., 2020) augmented reality technology for industry 4.0), we have not found any other one that proposes the combined use of them.

In order to show the feasibility of the methodology, its application has been deployed in a system for a real manufacturing scenario in Urola Solutions,[1] a medium-sized enterprise located in the Basque Country and which belongs to the MONDRAGON Corporation.[2] Thus, the contribution of the paper is twofold: On the one hand, the methodology that guides software engineers of SMEs in the task of creating software services that allow to improve the relationship between customers and a company; and on the other hand, a system as a proof of concept, which allows to observe the possibilities of that type of solutions.

In the rest of the paper, a brief overview of the customer life cycle phases is presented first. Then, some related works to the use of semantic descriptions and 3D technologies in the manufacturing scenario are shown. Next, the proposed methodology with its stages, phases and associated activities is described. Later, an implementation of a case study for a real manufacturing scenario that considers two phases of the customer life cycle and which has been developed using the proposed methodology is introduced. Finally, some conclusions and future work are discussed.

2. Brief overview of the customer life cycle

The traditional customer life cycle consists of a series of five phases that the customer goes through on his way to acquire a good or service, with the objective of turning people into paying customers and achieving a loyalty relationship between the customer and the brand. These phases, which range from capturing the attention of a potential customer to achieving the aforementioned loyalty, are: Reach, Acquisition, Conversion, Retention and Loyalty. In the specific case of manufacturing, a framework for the customer life cycle oriented to Industry 4.0 is presented in Hood et al. (2016), which condenses the five general phases into three:

3 Discover & shop: This phase has been greatly explored by e-commerce platforms such as Amazon,[3] AliExpress[4] or eBay,[5] mainly supported by the Business-to-Costumer (B2C) model, whose intention is to improve the end customer shopping experience by offering an intuitive experience and a recommendation system based on previous search and purchase preferences. However, in the manufacturing environment, and because of the complexity of their products, companies using the B2C model usually provide their customers with a generic physical or digital brochure with a lack of depth description of the offered goods or services. This approach may end up generating more doubts in the customer than it intends to solve, thus requiring the intervention of a representative of the sales department in an additional process different from that which constitutes their task, which is to close the sale. Furthermore, these companies not only provide goods and services to end customers, but also other businesses, i.e. Business-to-Business (B2B) model, which means that, if the offer is not well detailed, the purchasing processes are carried out in extensive meetings between the purchasing and sales departments of each company, taking more time than actually necessary.

4 Buy & install: This phase covers the actual buying of the product by the end customer and its installation. Many manufacturers rely on channel partners, such as dealers or distributors, for managing these tasks, so it is important to guarantee a seamless and timely information sharing between the manufacturers and their channel partners. However, this is usually done through transactional and milestone-based systems intended to synchronize activities that often rely on manual inputs and updates or outdated mechanisms that run once a day (Hood et al., 2016).

5 Use & service: This phase is oriented to customer retention and loyalty generation, and it is mainly applied in the B2B model, since the customer is tied to the use of products or services to manage their business. The most common way of providing a post-purchase service that includes repair, maintenance and support is through the customer service lines. However, this solution presents different problems such as fixed service hours, congestion on the lines, relatively long waiting times and the high possibility of not finding an immediate solution. These problems generate high customer discomfort creating an effect contrary to the desired loyalty. The effect is much more damaging in the manufacturing industry, where enormous economic loss can be

[1] https://www.urolasolutions.com/en/.
[2] https://www.mondragon-corporation.com/en/.
[3] https://www.amazon.com.
[4] https://www.aliexpress.com.
[5] https://www.ebay.com.

generated by not having the requested spare part or not having carried out preventive and corrective maintenance on time.

In the case study presented in this paper two of the three phases presented above are considered: Discover & Shop, and Use & Service. Regarding Discover & Shop two services have been implemented. The first one, allows customers to approach to the products in which they are interested by filling out a simple questionnaire and the second one allows customers to navigate through those products. The service implemented in relation to Use & Service is a kind of virtual technician that tries to solve the needs in terms of requesting spare parts.

3. Related work

There exist several technologies that enable the implementation of Industry 4.0, such as Internet of Things (IoT), Cloud Computing, Cybersecurity, Big Data and Analytics, Augmented/Virtual Reality, Additive Manufacturing, Simulation, and Robotics. In the specialized literature, several projects and case studies using these technologies in the context of manufacturing SMEs can be found. For example, project ESMERA (European SMEs Robotic Applications (Icer et al., 2018)) of the European Commission's Horizon 2020 Research and Innovation Programme aims to boost robotics innovation for European SMEs by funding projects such as REFLECT (CASP, 2020), which tackles the assembly of deformable parts in dishwashers by using a robotic system. Also under the Horizon 2020 Programme, project CloudiFacturing (CloudiFacturing, 2020) aims to optimize production processes and producibility in SMEs using Cloud/HPC-based modeling and simulation. More precisely, it supports projects such as 3D-CPAM (3D Clothing Production by Additive Manufacturing (Tomic et al., 2020)), which uses advanced HPC/Cloud services and modern 3D printing technologies to optimize the 3D fashion design manufacturing process, or D2Twin (D2LAB, 2020), which uses big data analytics to improve quality control and maintenance.

As mentioned in the previous section, our proposal is supported by two pillars: semantic-based technology to represent knowledge related to manufactured products, and 3D digital technology to visualize those products. These pillars are highly related to two of the aforementioned Industry 4.0 enablers: IoT and Augmented/Virtual reality.

From the IoT perspective where equipment and products communicate and are connected to each other, interoperating and integrating data and information is a demanding task that can be facilitated by semantic technologies such as ontologies. Ontologies allow to represent the semantics of knowledge and data in a formal, comprehensive and reusable way. Thus, several ontologies with different purposes have been defined in manufacturing scenarios, such as: the PSL ontology (Grüninger, 2009), which includes fundamental concepts for representing manufacturing processes; the MASON ontology (Lemaignan et al., 2006), an upper ontology that represents what authors consider the core concepts of the manufacturing domain: products, processes and resources; the SIMPM ontology (Šormaz and Sarkar, 2019), an upper ontology that models the fundamental constraints of manufacturing process planning: manufacturing activities and resources, time and aggregation; the MaRCO ontology (Järvenpää et al., 2019), which defines capabilities of manufacturing resources; the MSDL ontology (Ameri and Dutta, 2006), which allows to describe manufacturing services; the P-PSO ontology (Garetti and Fumagalli, 2012), which considers three aspects in the manufacturing domain: the physical aspect (the material definition of the system), the technological aspect (the operational view of the system) and the control aspect (the management activities), for information exchange, design, control, simulation and other applications; OntoSTEP (Barbau et al., 2012), which allows the description of product information mainly related to geometry; MCCO (Usman et al., 2011), which focuses on interoperability across the production and design domains of product lifecycle; ExtruOnt(Ramírez-Durán et al., 2020), which describes different aspects of extrusion machines such as their components and the 3D position, the spatial connections and the features of those components; CMO (Talhi et al., 2019), which represents the cloud manufacturing domain to support information exchange between cloud manufacturing resources; and SAREF4INMA (de Roode et al., 2020), which pursues favouring interoperability with industry standards. Finally, a literature review of papers related to ontologies in the area of product lifecycle management is presented in (Fortineau et al., 2013).

Regarding 3D digital technology, Virtual Reality (VR), Augmented Reality (AR) and Mixed Reality (MR) are considered relevant technologies for the new generation of intelligent manufacturing (Zhang et al., 2019). Virtual reality is a high-end human-computer interface that allows interaction with simulated environments in real time and through multiple sensorial channels (Liagkou et al., 2019). The users believe to be inside a reality that does not exist in truth, but they act like in the real world (Slater, 2009). Augmented Reality has been defined as a system which supplements the real world with virtual objects (computer generated) that appear to coexist in the same space as the real world (Bottani and Vignali, 2019; Masood and Egger, 2019). It provides benefits especially in designing products and production systems. While VR requires inhabiting an entirely virtual environment, AR uses existing natural environment and overlays virtual information on top of it. Finally, Mixed Reality like augmented reality, places digital or virtual objects in the real world. However, with mixed reality, users can quickly and easily interact with those digital objects to enhance their experience of reality or improve efficiency with certain tasks (Gonzalez-Franco et al., 2017). These technologies can been used for several purposes in the industrial and manufacturing environment, for example in the process of product design (Mourtzis et al., 2018; Guo et al., 2018), for assembly simulations (Tao et al., 2021; Al-Ahmari et al., 2016), for training purposes (Ordaz et al., 2015; Tao et al., 2019), for factory layout planning (Gong et al., 2019; Herr et al., 2018) or for improving maintenance services (Riboldi et al., 2021; Ababsa, 2020). Moreover, since 3D modeling has shown a realistic description of manufactured products by generating high-quality textures and proper lighting, it can be used for showcasing purposes. In this sense, it can be seen how some companies such as Schneider Electric[6] or Reid Supply[7] show their technical products in a interactive 3D product catalogue.

Nevertheless, to the best of our knowledge, no proposal has been made which combines the use of 3D and semantic technologies for a customer life cycle framework in a smart manufacturing scenario. Thus, in this paper, an initiative of this type is proposed.

4. Methodology

The approach introduced in this paper is intended to illustrate how SMEs could achieve closer and long-lasting relationships with customers through the implementation of new services aligned with the four industrial revolution (Industry 4.0). In this line, we present a step-by-step methodology that can help technicians of those SMEs in the task of deploying those services.

The proposed methodology considers the phases of a typical life cycle of an information system (Elmasri and Navathe, 2010) but what makes it different are the detailed guides that it provides

[6] https://sketchfab.com/blogs/enterprise/customer-stories/electronics/3d-viewer-schneider-electric.
[7] https://www.reidsupply.com/.

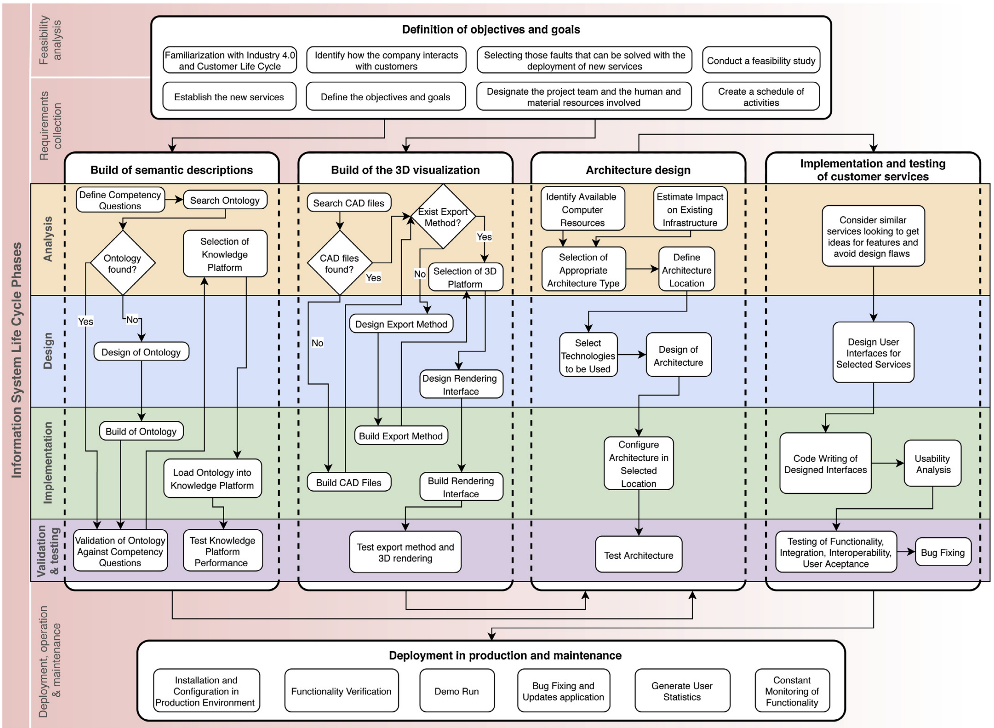


Fig. 1. Methodology diagram.

Table 1
Inputs and outputs of methodology stages.

Stage	Input	Output
Definition of objectives and goals	Interest of adopting Industry 4.0 for improving relationships with customers	List of new services, objectives, goals and roadmap for the deployment of those services
Build of semantic descriptions	Specifications of the manufactured products	Selected ontology and knowledge platform
Build of the 3D visualization	List of manufactured products	Export/synchronization method and rendering interface
Architecture design	Quality attribute requirements, knowledge platform, export method and rendering interface	An architecture
Implementation and testing of customer services	Definition of customer services	Implemented customer services
Deployment in production and maintenance	Implemented customer services	Customer services deployed in production and maintenance period started

for the incorporation of semantics descriptions and 3D visualization in the implementation of services associated to the customer life cycle. The methodology consists of stages that are divided into phases which in turn are made up of activities (see Fig. 1). In Table 1, a summary of the inputs and outputs of the methodology stages is presented. It represents the necessary income to obtain the desired result through the execution of the activities described in each stage of the proposed methodology. The different stages and activities that make up the proposed methodology are presented below.

4.1. Definition of objectives and goals

Based on the interest of the company in adopting Industry 4.0 for improving relationships with customers, the aim of this stage is to define a roadmap for the deployment of new services that can overcome some of identified weaknesses. Thus, the first activity is to carry out a *familiarization with Industry 4.0 and Customer Life Cycle*, where Industry 4.0 benefits and customer life cycle phases are presented in such a way that the concepts are clear and familiar to stakeholders.

The next activity is to *identify how the company interacts with customers* with the purpose of detecting problems and limitations. It is important to analyze the strengths and disadvantages of the business model in relation to competitors, and also the statistics of the customer support department, for example, to see which are the most frequent customer complaints.

Once the problems and limitations have been identified, it is necessary to narrow the scope by *selecting those faults that can be solved* *with the deployment of services aligned with the Industry 4.0*. Those services could include product exploration, training, maintenance, etc. in such a way that they allow to improve the relationships between customers and the company. This selection will be much more accurate thanks to the familiarization with Industry 4.0 and customer life cycle made previously.

The next activity is to *establish those new services* to be implemented throughout the customer life cycle. The *objectives and goals* (along with their priorities), that have to be achieved with the implementation of those services, must also be defined followed by a *feasibility study* that demonstrates that the project is profitable and that the company has the technical and organizational capacities to carry it out. In order to complete the roadmap, it is also important to *designate the project team and the human and material resources involved* as well as to *create a schedule of the activities* necessary to implement the project, including those responsible of each of them.

4.2. Build of semantic descriptions

One of the two pillars on which the proposed methodology is based on is the use of semantic descriptions for representing knowledge related to manufactured products. The detailed semantic description of the products and services will guarantee the creation of, for example, a robust catalogue, an accurate search engine and an improved technical support. To accomplish this, the first activity is to *define the ontology competency questions*. Competency Questions (CQs) are natural language questions outlining and constraining the scope of knowledge represented in an ontology (Wisniewski et al., 2019). Within the framework of this methodology, the competency questions should be aimed at describing the company's products and services, in addition to answering those questions that the customer may ask about their properties or characteristics.

Next, it is required to *search an ontology* that correctly answers those defined competency questions and, if necessary, make the pertinent modifications to adapt it correctly to what it has been looking for. There are different repositories where ontologies that span multiple domains can be found, such as LOV (Vandenbussche et al., 2016), Swoogle (Ding et al., 2004), ODP (Gangemi and Presutti, 2009) and Ontohub (Codescu et al., 2017), however, there are also ontologies created as result of scientific research and that are not found in those repositories, hence a search using the most popular search engines is also recommended. In the event that it is not possible to find an ontology that meets all the requirements, it is necessary to *design the ontology*. In the literature different methodologies such as On-To-Knowledge (Sure et al., 2004), Diligent (Pinto et al., 2004) and NeOn (Suárez-Figueroa et al., 2012) can be found to adequately guide engineers through a step-by-step ontology development process. Once the ontology is designed, the *ontology must be built*. This activity can be done making use of free tools[8] such as Protegé (Musen, 2015), NeOn Toolkit (Erdmann and Waterfeld, 2012) and SWOOP (Kalyanpur et al., 2006), which facilitate creation and editing, as well as verification of its structure and integrity. Finally, an *evaluation of the selected or built ontology against the competency questions* must be carried out in order to validate that they can all be answered. If this is not the case, it is necessary to go back to the search/design activity.

One might argue that an underlying ontology could be disregarded in favour of a database. However, the degree of flexibility that ontologies offer for representing the hierarchy and properties of individuals, as well as querying about them, cannot be achieved by databases. Moreover, ontologies are better suited for dealing with inheritance and are prepared for reasoning and inference purposes. In addition, there is usually a looser coupling between ontologies and applications that use them than between database schemas and applications. Finally, ontologies preserve semantics and provide a shared understanding of a domain, favouring interoperability (Uschold, 2015).

With the ontology finally defined, it is time to *select the knowledge platform* that will storage and sustain the ontology. Ontology storage models are classified into native stores and database stores. Native stores are directly built on the file system, whereas database-based repositories use relational or object relational databases as backend store. Storing ontologies using a native storage solution is straightforward compared to storing ontologies in databases, as relational databases do not support hierarchical relations directly (Abburu and Golla, 2016). Although the selection of the most suitable knowledge platform to store the ontology depends on several performance factors such as the response times of the queries and the speed of data loading, there are evaluations such as those presented in (Addlesee, 2019a,b) which help facilitate the process of selection. There are multiple ontology stores based on different technologies, some of which offer a free version with most of their functionalities. Examples of these are Virtuoso,[9] Stardog,[10] RDFox,[11] GraphDB,[12] Blazegraph[13] and AnzoGraph.[14]

The last two activities related to the build of semantic descriptions are the *loading of the ontology in the selected knowledge platform* and the *performance test of the knowledge platform*. In these activities, it is important to verify that there are no incompatibilities between the ontology and the selected knowledge platform, the data can be loaded and queried without problems, the necessary endpoints are provided and the performance is adequate. This can be done by uploading synthetic product and service data to the knowledge platform, converting the competency questions to the required query language, and executing those queries.

4.3. Build of the 3D visualization

The second pillar on which the proposed methodology is based on is the use of 3D rendering technologies for the visualization of products in order to provide an enhanced experience to customers. For this, it is necessary at this stage to verify the existence of the 3D models of the products and define how those models will be visualized by the customers. Furthermore, it is also necessary to establish a synchronization method between the modifications made to the 3D models of products and their final presentation in such a way that the changes made in the design are reflected in the customer's visualization.

The first activity consists of *verifying if there is a 3D representation (CAD files)* of the products offered by the company. Many manufacturing enterprises have those representations in the form of CAD files, which features can vary depending on the software used to create them. In the event that the company does not have 3D representations of its products, it is possible to search for them in free repositories like 3D Warehouse,[15] GrabCAD[16] and traceparts,[17] and other paid ones such as Sketchfab[18] where it can be found thousand of 3D models divided by categories.

[8] A complete listing of these tools can be found at https://www.w3.org/wiki/Ontology_editors.

[9] http://vos.openlinksw.com/owiki/wiki/VOS.
[10] https://www.stardog.com/.
[11] https://www.oxfordsemantic.tech/product.
[12] http://graphdb.ontotext.com/.
[13] https://blazegraph.com/.
[14] https://www.cambridgesemantics.com/anzograph/.
[15] https://3dwarehouse.sketchup.com/.
[16] https://grabcad.com/.
[17] https://www.traceparts.com/.
[18] https://sketchfab.com/.

If product CAD files are not available, *they must be built* using a CAD software. For this purpose, there exist free alternatives for desktop (e.g. FreeCAD[19]) or web environments (e.g. Onshape,[20] Autodesk Fusion 360,[21] CMS IntelliCAD PE[22]). Some of the latter are based on cloud processing, which introduces some key advantages such as that there is no need to invest in computer resources for processing, the latest version is always available, new features and bug fixes are added automatically, can be accessed from anywhere and can cost up to a third of the value of traditional software.[23]

The next activity is to *verify if an export/synchronization method exists*, in such a way that the changes made in the design are visualized by the customer. Most CAD software provide a manual option to export 3D models to local disk in different formats such as X3D, STL, GLTF or OBJ. However, it must be established if there is an automated option in which 3D models can be transferred to the server from where they will be rendered. If this option is not available, it is necessary to*design and build a synchronization method*. There are manual and automated methods to carry out the synchronization. The most basic one suggests that the designer export the modification and load it into the repository used by the web application. This method adds an extra burden to the designer's work and is prone to desynchronizations. Another approach suggests using CRON tasks that run the synchronization process on a set schedule or at times of low network traffic. It is also possible to synchronize using triggers at the time the designer makes a modification or the client visualizes a product, for which it is necessary to develop a query method to ask for recent modifications, either through an API in the client application or in the selected design software.

Once the export/synchronization method has been defined, *the 3D platform must be selected*, that is, the technology in charge of rendering the CAD models in the web environment. There are several free libraries and frameworks based on WebGL (an OpenGL based javascript library) for 3D rendering in web browsers[24] such as: Three.js,[25] Babylon.js[26] and X3DOM.[27] Selection of the most appropriate should be carried out taking into account previous experience, visualization needs and learning curve.

Next, the *design and build of the rendering interface* must be carried out. This consists of not only being able to render 3D models on screen, but also to offer the necessary navigation and exploration tools to improve the customer experience. For this purpose, it is necessary that the designed rendering interface allows, in addition to rendering the 3D models optimally and appropriately, modifying the visualization to add relevant information as a result of the annotations created in the ontology (name, model and characteristics of parts, etc.). The final activity consists of *performing communication tests between the CAD software and the rendering interface* using the export/synchronization method in order to check the compatibility, synchronization and visualization of the exported CAD models.

4.4. Architecture design

At this stage, the architecture that should support the knowledge platform, the export/synchronization method and the rendering interface must be defined. Moreover, as it is a solution oriented to the customer life cycle, it is necessary to define an architecture capable of supporting all the necessary services (customer management, catalogue, search engine, technical support, etc.).

Before selecting the type of architecture to use, it is necessary to *identify the available computing resources* as well as to *estimate the impact that the implementation of the new architecture* will have on the existing infrastructure. That is, to verify that the requirements for the inclusion of new technologies do not conflict with the requirements for the correct execution of existing applications (i.e. operating system and libraries versions) and that the current computing resources are sufficient or it is necessary to incur in an economic investment.

The next activity is to *select the appropriate architecture type*. Among the architectures for distributed systems (Puder et al., 2011) the best known are client/server, where clients contact the server, which is responsible for handling requests; n-tier, where (Web) clients interact with front-end services that then delegate requests to their (database) back ends; peer-to-peer, where each and every node can do both request and respond for the services; and Service-Oriented (SOA), where services are provided to the other components by application components, through a communication protocol over a network. In the framework of the Industry 4.0, several service-oriented architectures has been proposed such as RAMI4.0 (Schweichhart, 2016), ARUM (Leitão et al., 2013), SOCRADES (Karnouskos et al., 2009), PERFoRM, IMPROVE and BaSys4.0 (Trunzer et al., 2019). The selection of the architecture type should be based on the previous two activities and on the projections that the company has for its current infrastructure. The *definition of the architecture location* also depends on the latter, that is, if the architecture will be implemented in a client's own infrastructure or in the cloud. Currently, there are different cloud service providers (e.g. Amazon AWS,[28] Google Cloud,[29] Microsoft Azure[30]) which allow the implementation of infrastructure in an agile, configurable and secure way for a monthly cost, avoiding the need to incur in a high immediate economic expense.

The next activity is to *select the technologies that will be used in the design of the architecture*. Different technologies can be used for communication from the client application to the database server, such as web services or RESTful API's. The use of these technologies is recommended since it facilitates scalability and maintenance in the case of multiple client applications. Next, the *architecture design* must be carried out, which is essential to develop and maintain large-scale, long-living software systems. As stated in (Hasselbring, 2018), the architecture defines the system in terms of components and connections among those components. Moreover, the architecture shows the correspondence between the requirements and the constructed system, thereby providing some rationale for the design decisions. The architecture must be designed keeping in mind all the quality attribute requirements specified for the project, such as performance and reliability.

The last two activities consist of *configuring the architecture in the selected location* and *testing the architecture*. This last activity can be performed by defining use cases that describe specific interactions between the user of the system and the system itself, and testing those use cases against the implemented architecture checking its stability, performance, security and reliability.

4.5. Implementation and testing of customer services

In this stage, the implementation and testing of the services defined in the *Definition of objectives and goals* stage is carried out.

[19] https://www.freecadweb.org/.
[20] https://www.onshape.com/.
[21] https://www.autodesk.es/products/fusion-360/subscribe.
[22] https://www.intellicadms.com/.
[23] https://3dstartpoint.com/cloud-based-cad-software-101-how-it-works-and-top-5-picks/.
[24] A listing of the available frameworks (free and paid) for creating 3D content can be found at https://en.wikipedia.org/wiki/List_of_WebGL_frameworks.
[25] https://threejs.org/.
[26] https://www.babylonjs.com/.
[27] https://www.x3dom.org/.
[28] https://aws.amazon.com/.
[29] https://cloud.google.com/.
[30] https://azure.microsoft.com/.

These services must be aimed at guaranteeing an improvement in the customer life cycle through the incorporation of the technologies selected in the previous stages.

At first, it is important to *consider similar services* to those to be implemented, so that ideas can be obtained on how to add features and avoid design problems. That is, if one of the services to be created is a product catalogue, it is advisable to review industrial catalogues, such as Schneider EZList[31] and Siemens Industry Mall,[32] and popular online stores such as Amazon, AliExpress and eBay, looking for insights about the distribution of the sections on screen or, if a search service is included, services like Google or Yahoo can be checked to identify important functionalities (e.g. autocomplete, suggestions).

Next, the *design of user interfaces for the selected services* must be done. This process is not trivial as, in multiple times, users have to deal with frustration, fear and failure when faced with overly complex menus, incomprehensible terminology or chaotic navigation routes. To address this, interfaces should reduce anxiety and fear of use (embarrassing mistakes, privacy breaches, fear of scams), allow a smooth evolution (transition from novice to expert), allow compatibility with different input devices (keyboard, mouse, multi-touch displays, gestural input, haptic devices, VR devices), provide online help (text, video tutorials), improve the exploration of information (filter, select, navigate with minimum effort and without fear of getting lost). Designers should start by: (1) determining user needs: a thoroughly documented set of user needs clarifies the design process; (2) generating multiple design alternatives: rethinking interface designs for different situations often results in a better product for all users; and (3) carrying out extensive evaluations: which can be done before fully programming the functionality of the interfaces using sketches. Low-fidelity paper sketches are helpful, but online high-fidelity prototypes create a more realistic environment for expert review (Shneiderman et al., 2016).

The next activity consists of *code writing the services using the designed interfaces* followed by an *usability analysis*. This activity must be carried out using the software tools and programming languages preferred by each company according to its infrastructure and experience. This task should be modest if the interface design is complete and accurate. The usability analysis must be designed to find flaws in the developed services taking into account various evaluation criteria such as: time to learn, speed of performance, rate of errors by users, retention over time and subjective satisfaction. The language and expressions used in the developed services must also be taken into account. If the company operates internationally, translation into multiple languages is necessary, otherwise it must be adapted to the local culture. In (Bevan et al., 2016) several ISO standards related to usability are presented that serve as a guide to carry out this analysis: definition and concepts, evaluation reports, quality metrics, etc.

Finally, it must proceed with the *functionality, integration, interoperability and acceptance tests*. Those tests certify that the developed services meets the goals of designers and customers, moreover, a carefully tested prototype generates little change during deployment, avoiding costly upgrades. In those tests, a set of use cases for the services must be specified, with defined requirements such as the minimum response time for the combination of software and hardware or the degree of user acceptance measured through satisfaction surveys. If the services fail to meet these criteria, the services must be reworked. This testing activity usually results in a large number of bugs to be fixed which demands the use of many human resources, making the *bug fixing* a disorganized, chaotic and time-consuming process. The use of bug-tracking systems (e.g. Monday,[33] Airbrake,[34] Backlog,[35] Bugzilla,[36] Mantis[37]) facilitates this activity, assigning a state to each detected bug, managing the available resources and maintaining a traceability over the process.

4.6. Deployment in production and maintenance

In this last stage, the *developed services are installed and configured in the production environment*. It is recommended to use a version control in such a way that from now on all the changes made have a traceability. Version control helps teams keep track of all individual changes and prevent concurrent work from conflicting. In (Rao and Sekharaiah, 2016), a comparison of different version control systems can be found, which can help to select the one that best suits the needs of the company.

Once the services have been installed and configured, a *verification of their functionality* must be carried out to ensure that there are no incompatibilities with the infrastructure available in production. Furthermore, a *demo with real users must be carried out* in order to certify that the services behave the way they were designed. If there are errors detected in the production environment, *they must be corrected and updates to the libraries and packages* that are out of date in this environment, *must be performed*. Once the operation of the services is validated and approved, the maintenance period begins. During this period, the *functionality of the services must be constantly monitored* and *statistics on their use must be generated*, in order to assess whether the services positively affected customers throughout their life cycle.

5. Case study

In this section we present a case study in which the methodology described in the previous section has been applied to develop three services that allow to improve the customer life cycle in a real manufacturing company: Urola Solutions. That company develops advanced solutions for the packaging manufacture using blow moulding technology. Among the products they offer, there are different types of extruders depending on their production capacity and the types of plastic they use for packaging.

As stated in the first stage of the proposed methodology, first of all an analysis of the type of interaction between the company and its customers must be carried out in order to detect those faults that can be solved with the deployment of new services.

5.1. Definition of objectives and goals

Urola Solutions has a fairly solid range of clients and the acquisition of new clients is mainly based on the reputation achieved over time and the good references obtained as a result of these consolidated relationships. However, regarding potential new customers who browse through its website, currently they only see a general description of the products and services offered, in addition to the form and the telephone numbers of the contact section. As a consequence, requests for information are very common, and thus a personalized attention at the first contact between the future customer and the sales representative is an excessive burden for the company. A similar situation occurs with the customer service provided, where for a remote assistance an initial survey must

[31] https://ezlist.schneider-electric.com/.
[32] https://mall.industry.siemens.com/.
[33] https://monday.com/s/bug-tracking-software/.
[34] https://airbrake.io/.
[35] https://backlog.com/bug-tracking-software/.
[36] https://www.bugzilla.org/about/.
[37] https://www.mantisbt.org/.

Table 2
Duration of stages for the proposed scenario.

Stage	# of Weeks
Build of semantic descriptions	4
Build of the 3D visualization	6
Architecture design	2
Implementation of customer services	10
Deployment in production	2

be carried out before being redirected to the suitable specialist, increasing occupation times of customer service representatives and congestion on the lines.

Taking into account the aforementioned faults and a previous meeting where the phases of the customer life cycle and Industry 4.0 implementation advantages were exposed, the main goal of this case study was oriented to improve customers experience in two phases of the customer life cycle, Discover & Shop and Use & Service, through the development of three services: catalogue, searching module and virtual technician. For this purpose, the following general objectives were defined:

- Discover & Shop
  –Creation of a searching module with questions in natural language that allows customers to find the right product according to their production needs in a simple and efficient way.
  –Creation of an improved catalogue of the company's products, with 3D visualization and advanced navigation options that allows customers to see those products in detail.
- Use & Service
  –Creation of a virtual technician, which will try to solve the needs in terms of requesting spare parts and will only redirect the most complex cases to the appropriate specialist.

A feasibility study was conducted to determine if the company had the necessary staff, technical resources and capital to develop the project[38], and as a result some guidelines were established: the project duration could not be longer that 8 months and no more than 3 company employees should be involved with a weekly dedication of 32 h each as a maximum. Considering those guidelines, a team of 6 members was assembled as follow: two software engineers, one for the backend (API, Database) and another for the frontend (Application, Users management), with previous experience for the necessary programming and the R&D manager (with direction role) regarding company employees; and the three authors of this paper with guidance and support roles. The design of service interfaces was supported by a company designer with no direct commitment to the project.

The duration of stages in Table 2 was estimated for a period of 6 months taking into account the next stages of the proposed methodology. Moreover, in Fig. 2 an excerpt of the project roadmap defined at that time using the activities indicated in the proposed methodology is shown.

5.2. Build of semantic descriptions

Given that the main products marketed by Urola Solutions are extruders, an ontology should be found that correctly describes this type of industrial machines. For this, it is necessary to define competency questions that describe this type of machines and represent the questions that customers can ask about them when using the new services. Below, some of the competency questions defined and categorized according to the service to which they correspond are presented.

1. Catalogue
   (a) How many models of extruders does the company offer?
   (b) What kind of product do these extruders make?
   (c) What are the characteristics of the products manufactured by the extruders?
   (d) Is there a 3D model of the extruder that can be visualized?
   (e) What is the production (batch size) of a specific extruder model?
2. Searching module
   (a) What is the volume of the bottle that an extruder produces?
   (b) What is the size of the bottle?
   (c) How many bottles per hour does an extruder produce?
   (d) What is the necessary space to house an extruder?
3. Virtual technician
   (a) What are the possible solutions for a problem with the motor?
   (b) Where is the screw located?
   (c) Which supplier has a compatible replacement fan?
   (d) The extruder has stopped suddenly, what are the steps to follow?

After defining the competency questions, it proceeded to look for an ontology that would describe this type of industrial machines (extruders) in the repositories indicated in the methodology. However, the search in these repositories did not give a favorable result, so a search was made with the words "Extruder ontology" using google scholar. This search returned our scientific paper (Ramírez-Durán et al., 2020) with the description of an ontology named ExtruOnt.

As mentioned in Ramírez-Durán et al. (2020), the ExtruOnt ontology[39] is the one that describes extruders most thoroughly. The ExtruOnt ontology represents different aspects related to extrusion machines. It includes terms to describe (1) the *main components* of an extruder (e.g. the drive system), (2) the *spatial connections* between the extruder components (e.g. the filter is externally connected to the barrel), (3) the *different features* of the components (e.g. the power consumption of the motor is 40.5 kWh), (4) the *3D description* of the position of the components (e.g. the feed hopper is located at point $q(0,0,-1)$ in a 3D canvas), and, (5) the *sensors* that need to be used to capture indicators about the performance of that extruder (e.g. the temperature sensor that captures the melting temperature of the polymer). This ontology has been implemented using OWL 2[40] and the Protégé[41] development environment. Moreover, if an adequate ontology for another manufacturing scenario is not found, the detailed semantic description of the extruder that is made in the ExtruOnt ontology can serve as a model when making semantic descriptions of other products.

Next, the ExtruOnt ontology was evaluated against the competency questions. Based on the classes (concepts) and relationships described in the ontology, the competency questions were represented using the SPARQL language, dummy individuals were created and the questions were executed, verifying that the obtained result was equal to the expected result. Thus, it was possible to validate that the extruder models offered by Urola Solutions could be correctly described using the ExtruOnt ontology and the competency questions could be fully answered. As an example, the SPARQL query for the competency question 1(e) is presented as follows:

[38] The details of the feasibility study are not available due to the company's data privacy policies.

[39] http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt.
[40] https://www.w3.org/TR/owl2-overview/.
[41] https://protege.stanford.edu/.

ROADMAP

PROJECT TITLE	I4.0 CLC SERVICES	COMPANY NAME	UROLA SOLUTIONS
PROJECT MANAGER	R&D Manager	DATE	2019/11/26

WBS NUMBER	ACTIVITY TITLE	ACTIVITY OWNER	START DATE	DUE DATE	DURATION	PCT OF ACTIVITY COMPLETE
1	Definition of objectives and goals					
1.1	Familiarization with Industry 4.0 and Customer Life Cycle	R&D Manager	2019-11-18	2019-11-19	1	100%
1.2	Identify how the company interacts with customers	R&D Manager	2019-11-18	2019-11-19	1	100%
1.3	Selecting those faults that can be solved with the deployment of new services	R&D Manager	2019-11-19	2019-11-20	1	100%
1.4	Establish the new services	R&D Manager	2019-11-20	2019-11-21	1	100%
1.5	Define the objectives and goals	R&D Manager	2019-11-20	2019-11-21	1	100%
1.6	Conduct a feasibility study	R&D Manager	2019-11-21	2019-11-23	2	100%
1.7	Designate the project team and the human and material resources involved	R&D Manager	2019-11-25	2019-11-26	1	100%
1.8	Create a schedule of activities	R&D Manager	2019-11-25	2019-11-26	1	100%
2	Build of semantic descriptions					
2.1	Define competency questions	Author 1	2019-11-26	2019-11-27	1	0%
2.2	Search Ontology	Author 1	2019-11-27	2019-12-11	14	0%
2.2.1	Design of ontology	Author 1	2019-11-27	2019-12-04	7	0%
2.2.2	Build of ontology	Author 1	2019-12-04	2019-12-11	7	0%
2.3	Validation of ontology against competency questions	Author 1	2019-12-11	2019-12-12	1	0%
2.4	Selection of knowledge platform	Author 2	2019-12-12	2019-12-17	5	0%
2.5	Load ontology into knowledge platform	Author 2	2019-12-17	2019-12-19	2	0%
2.6	Test knowledge platform performance	Author 2	2019-12-19	2019-12-24	5	0%

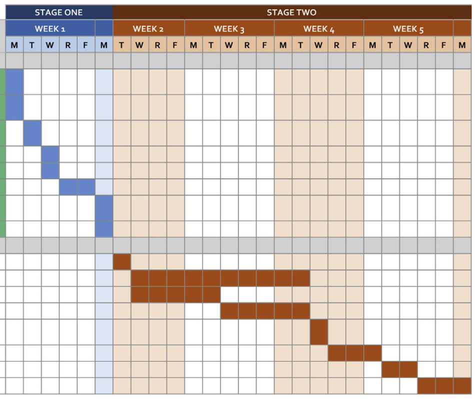


Fig. 2. Excerpt of the project roadmap.

```
PREFIX: <
  http://bdi.si.ehu.es/bdi/ontologies/ExtruOnt/Extruder01#>
PREFIX rdf: < http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: < http://www.w3.org/2002/07/owl#>
PREFIX s4inma: < https://w3id.org/def/saref4inma#>
PREFIX om: < http://www.ontology-of-units-of-measure.org/
  resource/om-2/>
PREFIX dcterms: < http://purl.org/dc/terms/>
SELECT ?batch ?size ?phenomenon ?description ?value
  WHERE { :E01 a ?restriction.
?restriction a owl:Restriction;
    owl:onProperty ?property;
     owl:allValuesFrom ?allValues.
?property owl:inverseOf s4inma:needsEquipment.
?allValues owl:intersectionOf ?intersection.
?intersection rdf:first ?batch;
    rdf:rest*/rdf:first ?node.
?batch rdfs:subClassOf s4inma:ItemBatch.
?node owl:hasValue ?size.
?size om:hasPhenomenon ?phenomenon.
?phenomenon dcterms:description ?description;
    om:hasNumericalValue ?value.
}
```

After selecting the ontology, it proceeded to choose the knowledge platform. Due to the fact that the project deadlines were quite tight, it was decided to support the selection activity in evaluations that have already been carried out. Therefore, three different RDF stores were selected taking into account the evaluation carried out in Addlesee (2019a) and Addlesee (2019b): Virtuoso, Stardog and RDFox. For the evaluation of these RDF stores, the SPARQL queries resulting from the competency questions in the previous stage were used. Although these three RDF stores provide similar response times for most of the queries, Stardog presents too high response times for specific types of queries and does not offer a free version, reasons why it was discarded. On the other hand, RDFox is an in-memory RDF store, which presents a great disadvantage in terms of data persistence, that is, if there is a crash or a system restart, the modifications made would be lost. The use of this RDF store involves creating and maintaining a backup system, which must respond to any eventuality. This, added to the fact that neither it offers a free version, has made this option be ruled out. Finally, Virtuoso is a hybrid database engine that combines the functionality of different types of databases in a single system. Virtuoso has fairly low response times (<500 ms). Moreover, it offers an open-source version and a SPARQL[42] endpoint for connection from external systems, therefore it was the RDF store selected.

Finally, an instance of Virtuoso Open-Source edition version 7.2.5 was installed in a Google Cloud virtual machine, in which the ExtruOnt ontology was loaded and the necessary namespaces and prefixes were defined. In addition, the operation of the endpoint was checked and the firewall rules for accepting requests were added. Furthermore, the SPARQL queries were executed again to check the correct functionality and performance of the knowledge platform.

5.3. Build of the 3D visualization

As a product manufacturer, Urola Solutions owns their CAD models. These models were created using a desktop CAD application, whose license is quite expensive and for which the period of free updates had already expired, and in addition it does not support an export method other than saving these models on the local disk. Within the objectives of this phase it was decided to look for an alternative CAD application that offered additional advantages with respect to the current CAD application as well as an export method according to the needs of the project.

Among the different CAD packages found for exporting CAD models, those based on cloud processing (Onshape, Autodesk Fusion 360, etc.) caught our attention for the reasons explained in Section 4.3. Among those options, it was decided to use Onshape since it has a free plan with which all the functionalities can be tested without incurring in any cost, it also allows collaboration between teams from mobile and desktop devices and has an API which can be used for exporting CAD models to the user application.

Using two CAD models of extruders provided by Urola solutions, it was verified that these models were fully customizable within the Onshape application and that the connection to the Onshape API for the export will work correctly. It should be noted that Onshape provides a client repository for its API[43] in different pro-

[42] https://www.w3.org/TR/sparql11-overview/.
[43] https://github.com/onshape-public.

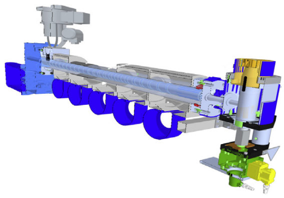


Fig. 3. 3D extruder visualization on the rendering interface.

gramming languages and runtime environments, one of them for NodeJS, which was the one used considering that the software engineers involved in the project had previous experience developing on it.

The fact that the new services would be available to the client through web browsers constituted a compelling reason for the selection of the 3D platform, for this reason it was necessary to look for frameworks or libraries that would support the rendering of CAD models through this medium. There are different open-source frameworks based on WebGL (an OpenGL based javascript library) for 3D rendering in web browsers. Among these, XML3D, X3DOM and Three.js stand out (Evans et al., 2014). The later one was selected because it has a simple learning curve, extensive documentation, lots of tutorials, and a powerful community. Using the selected 3D rendering framework, an interface was designed and built for the visualization of the extruder CAD models. The interface allows to visualize and interact with the CAD models of the extruders using actions controlled with the mouse and keyboard, this includes moving, rotating, zooming in, zooming out, selecting, making cuts to the model, etc. Fig. 3 shows an example of the visualization of an extruder with a vertical cut in the rendering interface.

The last activity was to perform communication tests between the Onshape API and a test application that incorporated the rendering interface in order to check the compatibility, synchronization and visualization of CAD models exported from Onshape. Those tests gave satisfactory results.

5.4. Architecture design

Following the methodology activities for this stage, the hardware resources available to configure the architecture were identified. Urola Solutions has outsourced its infrastructure, so any inclusion of resources to this infrastructure would generate an impact on the budgeted expense. For this reason, the company has a development environment on Google Cloud for the proof of concept of its new projects, which includes more than 20 free products (with a monthly usage limit) such as: Firebase, Firestore, Compute Engine, Cloud Storage, App Engine, among others. Therefore, it was decided to deploy the architecture on this development environment.

From the different types of architecture considered, the 3-Tier architecture type was the one that best suited the needs of the project, since in the future a native client for mobile devices could be developed, so it was necessary to separate the presentation tier from the application tier.

For the presentation tier of the selected architecture it was decided to use React,[44] a javascript library created by Facebook for the construction of user interfaces, with which the software engineers of the project had experience enough which would shorten programming times. In addition, FireBase[45] was used as hosting service since it has tools that facilitate administration, user management and online testing. FireBase also provides a free version (Spark plan) with more than enough resources for the development of the application, such as 1 GB of database storage in Cloud Firestore, 10 GB of hosting storage, custom domains and SSL security.

For the application tier, a REST API in charge of the functional business logic and the communication between the presentation tier and the data tier was developed using NodeJS,[46] an execution environment that allows to use javascript code in server; and ExpressJS,[47] a web application framework for NodeJS designed for the creation of APIs among others.

For the data tier, a virtual machine instance was created in the Google Cloud Compute Engine, specifically of type n1-standard-1 (1 virtual CPU, 3.75 GB of memory) with a hard disk of 30 GB, in which the knowledge platform was deployed. It was also decided to use the same virtual machine instance to host the API (application tier), in order to emulate the production deployment of the project since they would share the same server. Fig. 4 shows the main blocks of the designed architecture. The architecture was tested configuring one user in the Cloud Firestore database and making calls to the API, using a demo React app, to query the available extruders in the knowledge platform. Security configurations were applied to the virtual machine instance (application and data tiers) in order to only accept incoming requests from the React App deployed in Firebase (presentation tier).

5.5. Implementation of customer services

Three services that guarantee an improvement in the customer life cycle and that integrate the technologies selected in the previous stages have been implemented. These are a catalogue, a searching module and a virtual technician. The first two services are related to the Discover & Shop phase and the last one is related to the Use & Service phase of the customer life cycle.

Before proceeding with the design of the services, an analysis of similar services was made as stated in the methodology. The development of this activity gave us important guidelines on how the new services should be approached taking into account the characteristics, advantages and shortcomings of the service types analyzed, which are presented next.

Regarding the catalogues, on the one hand, different types of catalogues are used on e-commerce platforms (Business-to-Costumer (B2C) model). They are developed to improve the end customer shopping experience by offering intuitive visualizations and a recommendation system based on previous search and purchase preferences. On the other hand, in the Industry 4.0 environment, most of the companies provide their customers with a generic physical or digital brochure with a lack of depth description of the goods or services offered. Taking into account that many times their customers are other businesses, i.e. Business-to-Business (B2B) model, those brochures are insufficient because they do not incorporate many technical details and thus, the purchasing processes are carried out in extensive meetings between the purchasing and sales departments of each company, taking more time than actually necessary. A great advance in this aspect is introduced with the use of an online 3D rendering technology in the developed catalogue. Thus, a customer can see first-hand what the desired product looks like and moreover, the visualization is enriched with relevant information in such a way that the user experience is greatly improved.

44 https://reactjs.org/.

45 https://firebase.google.com/.
46 https://nodejs.org/.
47 https://expressjs.com/.

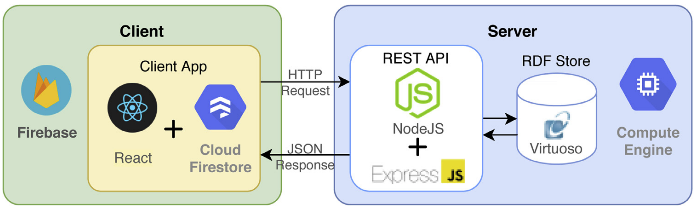


Fig. 4. Architecture.

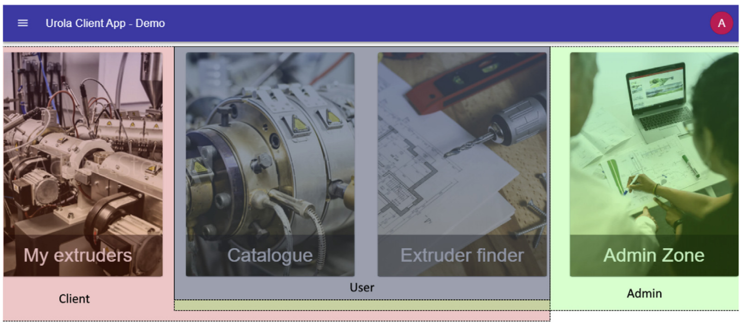


Fig. 5. Home screen options in the user application.

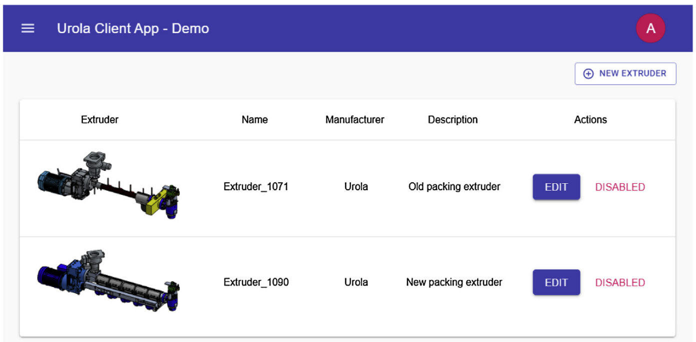
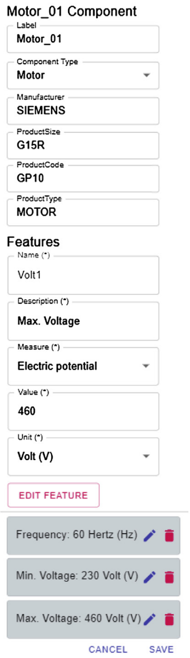


Fig. 6. Main screen of administration module.

Regarding the search modules, most of them are inserted into the catalogues and its operation is based on keywords, which can be a great challenge for novice users who do not know the correct terminology to perform a search (i.e. exact name, model, serial number, functionality, etc.). To avoid this limitation, the search module developed is initially based on a battery of simple questions that guide the customer in the selection process.

Regarding post-purchase services, the most common way of providing a post-purchase service that includes repair, maintenance and support is through the customer service lines. However, this solution presents different problems such as fixed service hours, congestion on the lines, relatively long waiting times and the high possibility of not finding an immediate solution. Those problems can generate high customer discomfort creating an effect contrary to the desired loyalty. The effect is much more damaging in the manufacturing industry, where enormous economic loss can be generated by not having the requested spare part or not having carried out preventive and corrective maintenance on time. In order to avoid those problems the developed service (virtual technician) provides a troubleshooting module available 24 h a day with the possibility of requesting spare parts directly from the main supplier or from other suppliers.

With the information resulting from the analysis of similar services, we proceeded to the design and code writing of the new services. However, before building those services, the REST API and the user application where the services will be hosted must




```
:_Motor01 rdf:type owl:NamedIndividual ,
                   components4ExtruOnt:Motor ,
                   eclass:C_AAB583003-gen ; #IRDI
          components4ExtruOnt:isMotorOf :_0DriveSystem01 ;
          om-2:hasPhenomenon :_MotorFrequency01 ,
                             :_MotorVoltageMaximum01 ,
                             :_MotorVoltageMinimum01 ;
          eont:ProductCode "GP10" ;
          eont:ProductSize "G15R" ;
          eont:ProductType "MOTOR" ;
          terms:creator "SIEMENS" ;
          rdfs:label "Motor01" .

:_MotorFrequency01 rdf:type owl:NamedIndividual ,
                            om-2:Frequency ;
                   om-2:hasValue :_MotorFrequencyMeasure01 ;
                   terms:description "Frequency" ;
                   rdfs:label "MotorFrequency01" .

:_MotorFrequencyMeasure01 rdf:type owl:NamedIndividual ,
                                   om-2:Measure ;
                          om-2:hasUnit om-2:hertz ;
                          om-2:hasNumericalValue 60 .

:_MotorVoltageMaximum01 rdf:type owl:NamedIndividual ,
                                 om-2:ElectricPotential ;
                        om-2:hasValue :_MotorVoltageMaximumMeasure01 ;
                        terms:description "Max. voltage" ;
                        rdfs:label "_MotorVoltageMaximum01" .

:_MotorVoltageMaximumMeasure01 rdf:type owl:NamedIndividual ,
                                        om-2:Measure ;
                               om-2:hasUnit om-2:volt ;
                               om-2:hasNumericalValue 460 .

:_MotorVoltageMinimum01 rdf:type owl:NamedIndividual ,
                                 om-2:ElectricPotential ;
                        om-2:hasValue :_MotorVoltageMinimumMeasure01 ;
                        terms:description "Min. voltage" ;
                        rdfs:label "_MotorVoltageMinimum01" .

:_MotorVoltageMinimumMeasure01 rdf:type owl:NamedIndividual ,
                                        om-2:Measure ;
                               om-2:hasUnit om-2:volt ;
                               om-2:hasNumericalValue 230 .
```

Fig. 7. Annotations generated when describing motor features.

be developed. Regarding the user application, it must contain an administration module (Admin Zone in Fig. 5) in which the information that will be available for the services will be managed. Fig. 5 shows the home screen of the user application with the administration module and the available services, as well as the roles that can access them.

In the administration module, the necessary annotations are generated, using the descriptions of the ExtruOnt ontology, to store the information of the extruder models in the knowledge platform. This module is accessible only to users with an administrator role. The first screen of this module (Fig. 6) presents a list of the extruder models already loaded, with the option to edit or disable their display, and a button to access a sub-module where a new extruder model can be created. This sub-module presents a dynamic form where the main characteristics of an extruder model (name, manufacturer, description, etc.) can be specified, in addition to the list of its components and their characteristics. It should be noted that the necessary information to create the form was completely extracted from the descriptions present in the ontology, with which the generation of a database schema from scratch was not necessary, being this one of the advantages of ontologies.

The left side of Fig. 7 presents an example of the form for describing a component of an extruder, more precisely its motor. The field *Component Type* links this instance of motor to the class `Motor` in ExtruOnt. Thanks to the information available in ExtruOnt about `Motor`s, the *Features* section of the form is dynamically personalized to allow only for measure types that can be applied to motors (e.g. Electric potential, Frequency). In the same way, once a measure type is selected (e.g. Electric potential) the field *Unit* is filled with appropriate units for that type of measure with regard to the information in ExtruOnt. The user can add as many as features as they want about the selected component (e.g. a frequency of 60 Hz, a minimum voltage of 230 V, a maximum voltage of 460 V), and then this information is internally transformed into a set of RDF triples annotated with the classes and properties of ExtruOnt, as can be seen in the right side of Fig. 7. Moreover, apart from the triples generated from the information available in the form, new triples that link the instance with additional relevant information inherited from the type of the component are generated. One example of additional information is the International Registration Data Identifier (IRDI) code of the type of component, whose semantic representation is available through the eClassOWL ontology.[48] This ontology models eCl@ss,[49] a classification standard for products, and is used to facilitate interoperability.

Additionally, this sub-module also includes a section where CAD models can be imported from the Onshape API and whose information is annotated in the knowledge platform using the descriptions of the 3D module from ExtruOnt. Figs. 8 and 9 show an example of the implemented import system and the generated annotations respectively. In the following the functionality of each of the services is described.

[48] http://www.heppnetz.de/projects/eclassowl/.
[49] https://www.eclass.eu.

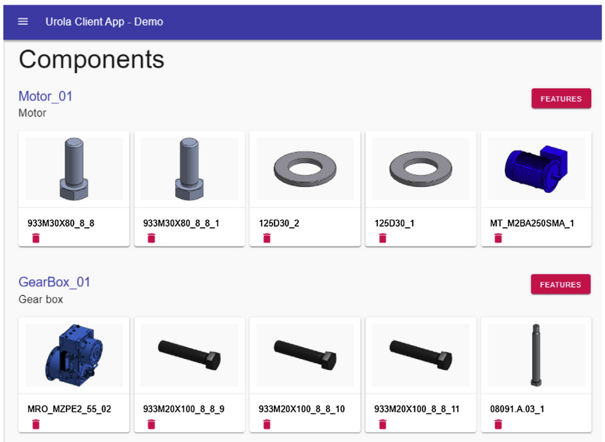


Fig. 8. Import system.

5.5.1. *Catalogue*

Algorithm 1. Get all extruders with their components

Input: ∅
Output: $\mathcal{R}$: Object list with extruders, components, properties and 3D models.
Variables: $\mathcal{E}$: Object list containing extruders with their main properties.
$\mathcal{P}$: Object list containing extruder components.
$\mathcal{C}$: Object list containing extruder components with their properties and 3D models.
function GetAllExtruders
  $\mathcal{R} \longleftarrow \{\}$
  $\mathcal{E} \longleftarrow$ *getDataS*(*"allExtrudersList"*) // Run SPARQL query named "allExtruderList"
  for each $e \in \mathcal{E}$ **do**
    if $e_{visible}$ = true **then** // Only visible extruders
      $e_{parts} \longleftarrow$ *partsById*(e_{id}) // Inject the extruder components
      append $\langle e \rangle$ to $\mathcal{R}$ // Append extruder to list
  return $\mathcal{R}$
function partsByIde$_i d$
  $\mathcal{C} \longleftarrow \{\}$
  $\mathcal{P} \longleftarrow$ *getDataS*(*"partsByExtruderId"*)
  for each $p \in \mathcal{P}$ **do**
    $p_{properties} \longleftarrow$ *propertiesById*(p_{id}) // Inject the component properties
    $p_{model} \longleftarrow$ *modelsById*(p_{id}) // Inject the component 3D models
    append $\langle p \rangle$ to $\mathcal{C}$ // Append component to list
  return $\mathcal{C}$

The developed catalogue (Fig. 10) is made up of a list of the different extruder models available together with the information corresponding to each model, manufacturer and production, as well as a navigation button which leads to 3D visualization. Algorithm 1 describes the process to obtain all extruders with their components, properties and 3D models inside the catalogue load workflow. This information is obtained from the annotated data in the knowledge platform managed by the administration module and rendered in a canvas using the Three.js framework (Fig. 11). The interaction with visualization and navigation is carried out using the mouse to control the events of the scene, in this way it is possible to zoom in, zoom out and rotate the 3D model. It is also possible to make cuts to the model in the three dimensional axes using the sliders in the upper left corner of the screen, allowing to view and select components that were hidden. The selection of the components of the extruder displays relevant information for the user related to that component such as type, model and brand. There is also a button to request more information, with which a contact form is displayed. Its information contains, in addition to the data filled in by the potential client, the information of the extruder model displayed, helping the sales representative to better guide the first contact with the user in order to make them a future customer.

5.5.2. *Searching module*

The searching module presents two types of interfaces: simple and advanced. For the simple type, five general questions are presented, focused on the production and dimension of the extruders, which help customers to find the ideal product according to their requirements. Based on the answers to the questions, a SPARQL query will be formulated, which will be executed against the annotated information about the different extruder models. These questions are:

- What is the volume of the bottle that you want to produce?
- What is the size of the bottle?
- How many bottles would you like to produce?
- How many hours does your company work per day?
- What is the available space (in meters) that you have for your new extruder? (see Fig. 12)

Algorithm 2 describes the procedure carried out when running a basic search.

Algorithm 2. Get all extruders with basic search parameters

Input: $\mathcal{P}_{vol}$: Volume of the bottle.
$\mathcal{P}_{wh}$: Object with width and height of the bottle.
$\mathcal{P}_{prod}$: Quantity of bottles to produce in a day.
$\mathcal{P}_{hpd}$: Working hours per day.
$\mathcal{P}_{esize}$: Object with width, height and length of the extruder.
Output: $\mathcal{R}$: Object list with extruders, components, properties and 3D models.
Variables: $\mathcal{E}$: Object list containing extruders with their main properties.
$\mathcal{F}$: String with filter options.
function GetAllExtrudersByParams$\mathcal{P}_{vol}, \mathcal{P}_{wh}, \mathcal{P}_{prod}, \mathcal{P}_{hpd}, \mathcal{P}_{esize}$
  $\mathcal{R} \longleftarrow \{\}$
  $\mathcal{Q}_{sparql} \longleftarrow$ *getQuery*(*"BasicSearchQuery"*) // Return query template for basic search

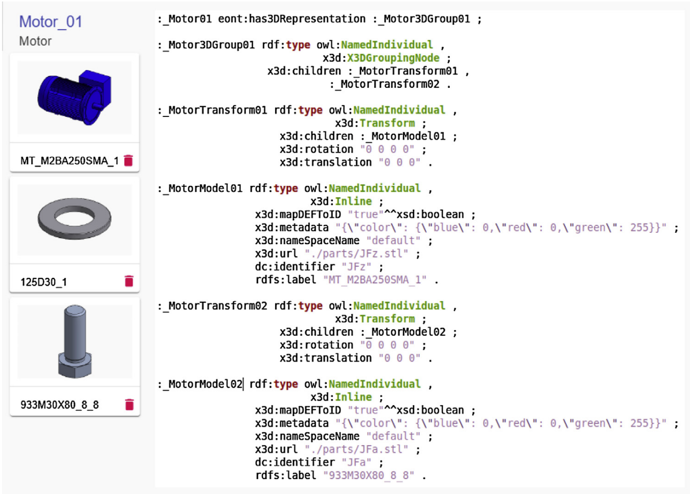


Fig. 9. Annotations generated when describing motor 3D models.

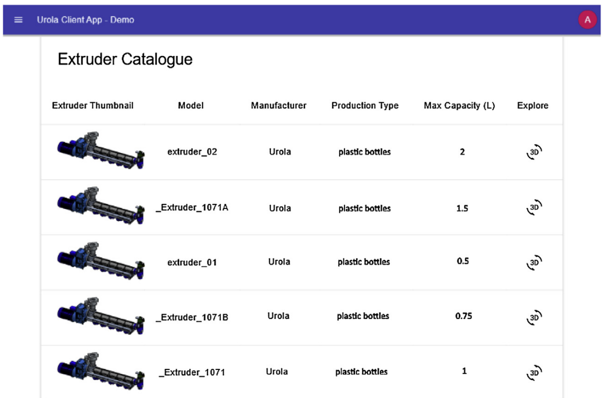


Fig. 10. Main screen of the catalogue.

```
4:  𝓕 ⟵ ValidateFilters(𝒫vol, 𝒫wh, 𝒫prod, 𝒫hpd, 𝒫esize) // Return filter options
5:  append (𝓕) to 𝒬sparql // Append filter options to query
6:  𝓔 ⟵ getDataQ(𝒬sparql) // Run SPARQL query with filter options
7:  for each e ∈ 𝓔 do
8:    if e_visible = true then // Only visible extruders
9:      e_parts ⟵ partsById(e_id) // Inject the extruder components
10:     append (e) to 𝓡 // Append extruder to list
11: return 𝓡
```

The advanced interface complements the previous one, allowing searches based on the components of the extruders, and taking advantage of the inheritance characteristics of the Web Ontology Language (OWL) with which the ExtruOnt ontology was built. More precisely, it allows to indicate the specific class to which a component must belong. This advanced search engine is activated in the option *Add advanced condition* that is displayed after answering the questions of the simple search engine and consists of two sections. First, it shows the component tree of an extruder (Fig. 13A), which is generated from the parthood relationships present in the ontology (Fig. 13D). Selecting a component will display the specializations of that component (subclasses), allowing further refinement of the search (Fig. 13B). Moreover, properties related to the selected specialization will be shown (Fig. 13C). Each of the properties will serve for one of the following two purposes: information or refinement. On the one hand, properties for information purposes are those that according to ExtruOnt are either associated to the selected specialization or inherited from its superclasses. These properties will have a fixed value for that component. In the example shown in Fig. 13, due to the description of the class `ExtrusionHeadForProfiles` in ExtruOnt (Fig. 13E), the property *has type of extrudate* has been assigned the fixed value *Profile*. On the other hand, as

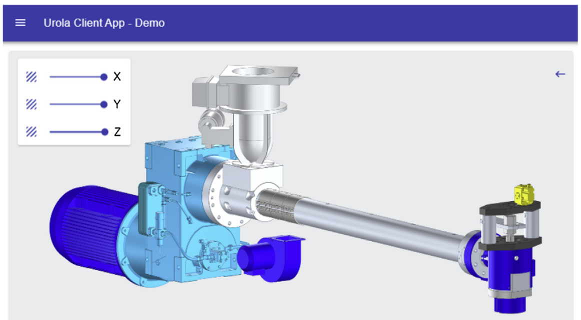


Fig. 11. 3D extruder rendered in a canvas.

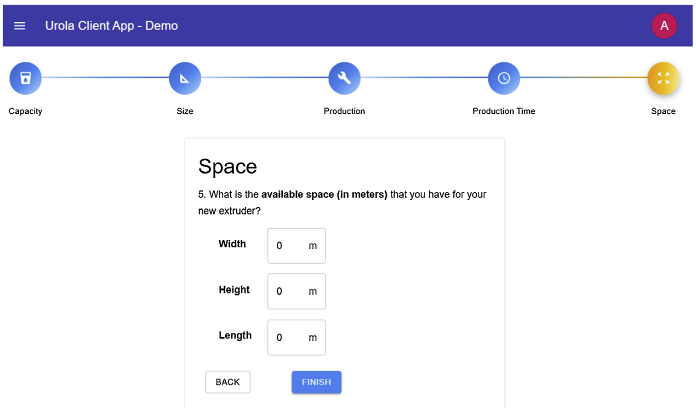


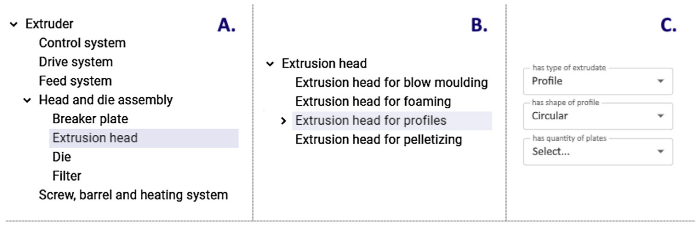
Fig. 12. Example of a question in the searching module.

their name indicate, properties for refinement will help refine the search by allowing the user to provide values for them. More precisely, they will be those properties in ExtruOnt appearing in the restrictions of the subclasses of the selected specialization. In the example, a restriction exists in classes `ExtrusionHeadForCircularProfiles` and `ExtrusionHeadForNonCircularProfiles` which states values of `Circular` and `Non-circular` respectively for property `hasShapeOfProfile`. Thus, those two values will appear for refinement in the property *has shape of profile* of the form. Although the related annotations are not shown for space matters, the same applies for the property *has quantity of plates*.

Once the search has been carried out, simple or advanced, a list of those extruders that meet the indicated conditions are displayed using the same format and functionality of the catalogue. It is worth mentioning that when the form to request more information is used on an extruder obtained through the search module instead of the catalogue, the search parameters are included in the information sent to the sales representative.

5.5.3. *Virtual technician*

Using this service, customers can see the list of their extruders bought or rented (B2C or B2B model, respectively) with the same format used for the catalogue and with the same interaction capacity in the 3D visualization (Fig. 14). Technically it is supported by two modules: (1) A library of solutions for the most common problems generated by the manipulation of extruders created by Urola Solutions from the experience gained over the years. This library is annotated in the knowledge platform and associated with the extruder components through the custom property "related to" whose domain and range are a problem and a component, respectively. The system loads the library by filtering through the extruder component that is selected in the 3D visualization, avoiding showing irrelevant information and guiding the user step by step in solving the problem. If there is no solution, a support ticket is opened with which the technician will receive the history of previous actions, so that the client does not have to explain the problem from scratch. (2) A module for requesting spare parts in which individual parts can be requested from the main supplier. In case of not having a stock of parts, other suppliers where stock is available are suggested. The query of the available stock and other suppliers is done through the already-existing internal company service using its own codes for the components. However, when needed, IRDI codes in the annotations of the components could be used to broaden the search to other components classified under the same IRDI code. Algorithm 3 describes the procedure carried out when a spare part is requested.



D.

```
:ExtrusionHead rdf:type owl:Class ;
               rdfs:subClassOf [ rdf:type owl:Restriction ;
                                 owl:onProperty :isHeadOf ;
                                 owl:someValuesFrom :HeadAndDieAssembly
                               ] ;
               rdfs:label "Extrusion head"@en .

:HeadAndDieAssembly rdf:type owl:Class ;
                    owl:equivalentClass [ owl:intersectionOf ( [ rdf:type owl:Restriction ;
                                                                 owl:onProperty :isHeadAndDieAssemblyOf ;
                                                                 owl:someValuesFrom :Extruder
                                                               ]
                                                               ...
                                                             ) ;
                                          rdf:type owl:Class
                                        ] ;
                    rdfs:label "Head and die assembly"@en .

:isHeadOf rdf:type owl:ObjectProperty ;
          rdfs:subPropertyOf <http://www.ontologydesignpatterns.org/cp/owl/partof.owl#isPartOf> ;
          rdfs:domain :ExtrusionHead ;
          rdfs:range :HeadAndDieAssembly ;
          rdfs:label "is head of"@en .

:isHeadAndDieAssemblyOf rdf:type owl:ObjectProperty ;
           rdfs:subPropertyOf <http://www.ontologydesignpatterns.org/cp/owl/partof.owl#isPartOf> ;
           rdfs:domain :HeadAndDieAssembly ;
           rdfs:range :Extruder ;
           rdfs:label "is head and die assembly of"@en .
```

E.

```
:ExtrusionHeadForProfiles rdf:type owl:Class ;
      owl:equivalentClass [
            owl:intersectionOf ( :ExtrusionHead
                                 [ rdf:type owl:Restriction ;
                                   owl:onProperty :hasTypeOfExtrudate ;
                                   owl:hasValue :Profile
                                 ]
                               ) ;
            rdf:type owl:Class
          ] ;
      rdfs:comment "The extrusion heads of this type are used for pipes ..."@en ;
      rdfs:label "Extrusion head for profiles"@en .

:ExtrusionHeadForCircularProfiles rdf:type owl:Class ;
      owl:equivalentClass [ owl:intersectionOf ( :ExtrusionHeadForProfiles
                                                 [ rdf:type owl:Restriction ;
                                                   owl:onProperty :hasShapeOfProfile ;
                                                   owl:hasValue :Circular
                                                 ]
                                               ) ;
                            rdf:type owl:Class
                          ] ;
      rdfs:comment "Extrusion head for a circular shape of profile."@en ;
      rdfs:label "Extrusion head for circular profiles"@en .

:ExtrusionHeadForNonCircularProfiles rdf:type owl:Class ;
     owl:equivalentClass [ owl:intersectionOf ( :ExtrusionHeadForProfiles
                                                [ rdf:type owl:Restriction ;
                                                  owl:onProperty :hasShapeOfProfile ;
                                                  owl:hasValue :Non-circular
                                                ]
                                              ) ;
                           rdf:type owl:Class
                         ] ;
     rdfs:comment "Extrusion head for a non-circular shape of profile."@en ;
     rdfs:label "Extrusion head for non-circular profiles"@en .
```

Fig. 13. Example of annotations used to construct the advanced interface of the searching module.

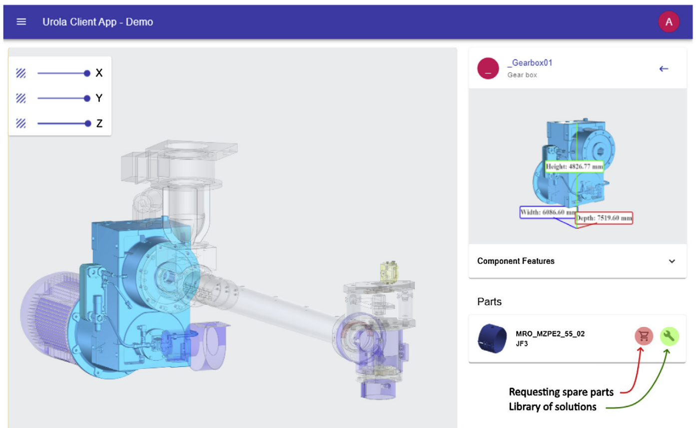


Fig. 14. Virtual technician: customer options after selecting an extruder component.

Table 3
Average response times for system evaluation (in seconds).

Action	Avg. response time (s)
Catalogue loading	0.938
Extruder insertion	2.457
Loading the 3D rendering of an extruder	9.631
3D models data import from OnShape	2.736
Loading the library of solutions	1.428

Algorithm 3. Requesting spare part to available provider

Input: $\mathcal{C}_{id}$: Component identifier.
$\mathcal{P}_{id}$: Provider identifier.
Output: $\mathcal{R}$: Response object with provider list or order status.
Variables: $\mathcal{A}$: Part availability.
$\mathcal{L}$: List of alternative providers.
$\mathcal{O}$: Order status.

function RequestSparePart$\mathcal{C}_{id}$, $\mathcal{P}_{id}$
  $\mathcal{R} \longleftarrow \{\}$
  $\mathcal{A} \longleftarrow$ *partAvailableInWarehouse*($\mathcal{C}_{id}$) // Always check availability in warehouse first
  if $\mathcal{A} = true$ **then**
    $\mathcal{O} \longleftarrow$ *requestPartToWarehouse*($\mathcal{C}_{id}$)
    $\mathcal{R} \longleftarrow \langle$"*Order*", $\mathcal{O}\rangle$ // Return order status
  els if $\mathcal{A} = false and \mathcal{P}_{id} = \emptyset$ **then**
    $\mathcal{L} \longleftarrow$ *GetProvidersByPartId*($\mathcal{C}_{id}$) // Get list of alternative providers or NULL
    $\mathcal{R} \longleftarrow \langle$"*Providers*", $\mathcal{L}\rangle$ // Return available providers
  else
    $\mathcal{L} \longleftarrow$ *GetProvidersByPartId*($\mathcal{C}_{id}$)
    if $\mathcal{P}_{id}$ **included in** $\mathcal{L}$ **then**
      $\mathcal{O} \longleftarrow$ *requestPartToProvider*($\mathcal{C}_{id}$, $\mathcal{P}_{id}$)
      $\mathcal{R} \longleftarrow \langle$"*Order*", $\mathcal{O}\rangle$
    else
      $\mathcal{R} \longleftarrow \langle$"*Providers*", $\mathcal{L}\rangle$
  return $\mathcal{R}$

At the time of writing this paper, the activities of usability analysis and testing of functionality, integration, interoperability and user acceptance are being carried out by Urola Solutions. However, the feedback of the customers that are trying the system is encouraging. They really value the possibilities it offers in relation to product exploration, training of employees and recommendations regarding spare parts. Moreover, the results regarding the evaluation of the system response times carried out in each of the developed services in those real manufacturing scenarios, varying the information load, can be seen in Table 3 (it shows the average times obtained for the most relevant tests). As can be observed, most actions are below the recommended average time of 4.7 s for loading web pages[50] even using the limited deployment environment that was available.

5.6. Deployment in production and maintenance

This stage and its activities will be addressed once the last activities of the previous stage have been completed.

6. Conclusions and future work

This paper presents a novel approach focused on the customer life cycle to facilitate the implementation of Industry 4.0 in those Small and Medium Enterprises belonging to traditional manufacturing sectors, whose limited resources and the high degree of complexity that this entails, make them desist from starting this process.

The main contribution of this paper is a methodology that describes a series of well defined stages and activities, easy to understand and execute, with which this transition can be carried out using minimal economic resources and taking advantage of new technologies that were not previously easily accessible. This methodology is mainly based on the use of semantic technologies and 3D visualization, which have been extensively explored individually, but to the best of our knowledge, they have not been used together. On the one hand, semantic technologies, in this case ontologies, provide a high degree of flexibility for the description of knowledge, in addition to allowing inference and reasoning capabilities that are difficult to achieve by traditional databases. On the other hand, 3D rendering technologies offer an enhanced visual representation that includes better graphics and navigation controls, allowing the user an interactive and improved experience. All these benefits focused on a particular type of user, the customer, throughout their life cycle will improve the relationship between the customer and the enterprise, achieving a high degree of loyalty.

[50] https://www.machmetrics.com/speed-blog/average-page-load-times-for-2020.

The second contribution is a system, developed as a proof of concept in a real manufacturing enterprise, in which the introduced methodology has been followed step by step and the previously described technologies have been used to generate a series of services that positively affect the relationship with the customer in two of the three phases of the customer life cycle. This system has an underlying ontology that allows to describe extrusion machines in a reliable and flexible way, making the proposal easily extensible to any company that works with this type of machine. Furthermore, the ontology can be modified to describe any other type of product (e.g. toys, furniture) and the system can be adapted to be used by other manufacturing enterprises. The development of this system, currently in the last stage of testing before being put into production, serves as an example of the effectiveness of the proposed methodology.

Future work contemplates the creation of a native application for mobile devices, improving the scope of the system. It is also important to explore the improvements that can be obtained in the second phase of the customer life cycle, i.e. Buy & Install, eventually expanding the number of services offered with services related to channel partners (e.g. dealers or distributors), such as visualizing and analyzing real-time data from these partners for optimizing delivery, predicting issues and making better operational decisions.

Author statement

Víctor Julio Ramírez-Durán: Conceptualization, Methodology, Software, Investigation, Formal analysis, Writing – Original Draft, Writing – Review & Editing, Visualization

Idoia Berges: Conceptualization, Methodology, Resources, Writing – Review & Editing, Supervision, Validation

Arantza Illarramendi: Conceptualization, Methodology, Resources, Writing – Review & Editing, Supervision, Validation, Funding acquisition, Project administration

Declaration of Competing Interest

The authors report no declarations of interest.

Acknowledgements

The authors wish to thank Urola Solutions for allowing us to carry out the proof of concept, for their help with information about the customer service process and for providing real data. This research was funded by the Spanish Ministry of Economy and Competitiveness, grant number FEDER/TIN2016-78011-C4-2R and the Basque Government under Grant No.: IT1330-19. The work of Víctor Julio Ramírez-Durán is funded by the contract with reference BES-2017-081193.

References

Ababsa, F., 2020]. Augmented reality application in manufacturing industry: maintenance and non-destructive testing (ndt). In: De Paolis, L.T., Bourdot, P. (Eds.), Augmented Reality, Virtual Reality, and Computer Graphics. Springer International Publishing, Cham, pp. 333–344.

Abburu, S., Golla, S.B., 2016]. Ontology storage models and tools: an authentic survey. J. Intell. Syst. 25 (4), 539–553, http://dx.doi.org/10.1515/jisys-2014-0167, 01 Oct. https://www.degruyter.com/view/journals/jisys/25/4/article-p539.xml.

Addlesee, A., 2019a]. Comparison of Linked Data Triplestores: Developing the Methodology. https://medium.com/wallscope/comparison-of-linked-data-triplestores-developing-the-methodology-e87771cb3011.

Addlesee, A., 2019b]. Comparison of Linked Data Triplestores: a New Contender. https://medium.com/wallscope/comparison-of-linked-data-triplestores-a-new-contender-c62ae04901d3.

Al-Ahmari, A.M., Abidi, M.H., Ahmad, A., Darmoul, S., 2016]. Development of a virtual manufacturing assembly simulation system. Adv. Mech. Eng. 8 (3), 1687814016639824.

Alicke, K., Rachor, J., Seyfert, A., 2016]. Supply Chain 4. 0 – The Next-Generation Digital Supply Chain (accessed 28.09.20) https://www.mckinsey.com/business-functions/operations/our-insights/supply-chain-40-the-next-generation-digital-supply-chain.

Ameri, F., Dutta, D., 2006]. An upper ontology for manufacturing service description. In: ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, pp. 651–661, http://dx.doi.org/10.1115/DETC2006-99600.

Barbau, R., Krima, S., Sudarsan, R., Narayanan, A., Fiorentini, X., Foufou, S., Sriram, R.D., 2012]. Ontostep: enriching product model data using ontologies. Comput.-Aided Des. 44 (6), 575–590, http://dx.doi.org/10.1016/j.cad.2012.01.008.

Bevan, N., Carter, J., Earthy, J., Geis, T., Harker, S., 2016]. New iso standards for usability, usability reports and usability measures. In: Kurosu, M. (Ed.), Human-Computer Interaction. Theory, Design, Development and Practice, Springer International Publishing, Cham, pp. 268–278.

Bottani, E., Vignali, G., 2019]. Augmented reality technology in the manufacturing industry: a review of the last decade. IIE Trans. 51, 284–310, http://dx.doi.org/10.1080/24725854.2018.1493244.

CASP, 2020]. Reflect (accessed 28.09.20) http://www.esmera-project.eu/reflect/.

CloudiFacturing, 2020]. Cloudification of Production Engineering for Predictive Digital Manufacturing (accessed 28.09.20) https://www.cloudifacturing.eu/.

Codescu, M., Kuksa, E., Kutz, O., Mossakowski, T., Neuhaus, F., 2017]. Ontohub: a semantic repository engine for heterogeneous ontologies. Appl. Ontol. 12 (3–4), 275–298.

Cotteleer, M., Sniderman, B., 2017]. Forces of Change: Industry 4.0. Deloitte, Tech. Rep. https://www2.deloitte.com/content/dam/insights/us/articles/4323_Forces-of-change/432_Forces-of-change_Ind4-0.pdf.

D2LAB, 2020]. Improving Quality Control and Maintenance at Manufacturing SMES Using Big Data Analytics (accessed 28.09.20) https://www.cloudifacturing.eu/experiment-3-improving-quality-control-and-maintenance-at-manufacturing-smes-using-big-data-analytics/.

de Roode, M., Daniele, L., Fernandez-Izquierdo, A., Poveda, M., Garcia-Castro, R., 2020]. Saref4inma: a saref extension for the industry and manufacturing domain. Semant. Web J. 11 (6).

Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R.S., Peng, Y., Reddivari, P., Doshi, V., Sachs, J., 2004]. Swoogle: a search and metadata engine for the semantic web. In: Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, CIKM'04, ACM, New York, NY, USA, pp. 652–659, http://dx.doi.org/10.1145/1031171.1031289 http://swoogle.umbc.edu.

Elmasri, R., Navathe, S., 2010]. Fundamentals of Database Systems, 6th Edition. Addison-Wesley Publishing Company, USA.

Erdmann, M., Waterfeld, W., 2012]. Overview of the NeOn Toolkit. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 281–301, http://dx.doi.org/10.1007/978-3-642-24794-1_13.

Evans, A., Romeo, M., Bahrehmand, A., Agenjo, J., Blat, J., 2014]. 3d graphics on the web: a survey. Comput. Graph. 41, 43–61, http://dx.doi.org/10.1016/j.cag.2014.02.002 http://www.sciencedirect.com/science/article/pii/S0097849314000260.

Fortineau, V., Paviot, T., Lamouri, S., 2013]. Improving the interoperability of industrial information systems with description logic-based models – the state of the art. Comput. Ind. 64 (4), 363–375, http://dx.doi.org/10.1016/j.compind.2013.01.001.

Gangemi, A., Presutti, V., 2009]. Ontology Design Patterns. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 221–243, http://dx.doi.org/10.1007/978-3-540-92673-3_10.

Garetti, M., Fumagalli, L., 2012]. P-pso ontology for manufacturing systems. IFAC Proc. Vol. 45 (6), 449–456, http://dx.doi.org/10.3182/20120523-3-RO-2023.00222, 14th IFA, Symposium on Information Control Problems in Manufacturing. http://www.sciencedirect.com/science/article/pii/S1474667016331901.

Gong, L., Berglund, J., Berglund, A.-F., Johansson, B., Wang, Z., Borjessong, T., 2019]. Development of virtual reality support to factory layout planning. Int. J. Interact. Des. Manuf., 10.1007/S12008-019-00538-X.

Gonzalez-Franco, M., Pizarro, R., Cermeron, J., Li, K., Thorn, J., Hutabarat, W., Tiwari, A., Bermell-Garcia, P., 2017]. Immersive mixed reality for manufacturing training. Front. Robot. AI 4 (02), http://dx.doi.org/10.3389/frobt.2017.00003.

Grüninger, M., 2009]. Using the PSL Ontology. Springer Berlin Heidelberg, Berlin Heidelberg, pp. 423–443, http://dx.doi.org/10.1007/978-3-540-92673-3_19.

Guo, Z., Zhou, D., Chen, J., Geng, J., Lv, C., Zeng, S., 2018]. Using virtual reality to support the product's maintainability design: immersive maintainability verification and evaluation system. Comput. Ind. 101, 41–50.

Hasselbring, W., 2018]. Software architecture: past, present, future. In: The Essence of Software Engineering. Springer, Cham, pp. 169–184.

Herr, D., Reinhardt, J., Reina, G., Krüger, R., Ferrari, R.V., Ertl, T., 2018]. Immersive modular factory layout planning using augmented reality. Proc. CIRP 72, 1112–1117.

Hood, J., Brady, A., Dhanasri, R., 2016]. Industry 4. 0 Engages Customers.

Icer, E., Weisenburger, S., Zechmair, M., Peters, A., Michalos, G., Knoll, A., Makris, S., 2018]. Esmera – European smes robotics applications. ISR 2018; 50th International Symposium on Robotics, 1–4.

Järvenpää, E., Siltala, N., Hylli, O., Lanz, M., 2019]. The development of an ontology for describing the capabilities of manufacturing resources. J. Intell. Manuf. 30 (2), 959–978, http://dx.doi.org/10.1007/s10845-018-1427-6.

Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C., Hendler, J., 2006]. Swoop: a web ontology editing browser. J. Web Semant. 4 (2), 144–153, http://dx.doi.org/10.1016/j.websem.2005.10.001, semantic Grid -The Convergence of Technologies. http://www.sciencedirect.com/science/article/pii/S1570826805000326.

Karnouskos, S., Bangemann, T., Diedrich, C., 2009]. Integration of legacy devices in the future soa-based factory. IFAC Proc. Vol. 42 (4), 2113–2118, http://dx.doi.org/10.3182/20090603-3-RU-2001.0487, 13th IFA, Symposium on Information Control Problems in Manufacturing., http://www.sciencedirect.com/science/article/pii/S1474667016341155.

Kharlamov, E., Mehdi, G., Savkovic, O., Xiao, G., Kalayci, E.G., Roshchin, M., 2019]. Semantically-enhanced rule-based diagnostics for industrial internet of things: the SDRL language and case study for siemens trains and turbines. J. Web Semant. 56, 11–29, http://dx.doi.org/10.1016/j.websem.2018.10.004.

Kusiak, A., 2018]. Smart manufacturing. Int. J. Prod. Res. 56 (1–2), 508–517, http://dx.doi.org/10.1080/00207543.2017.1351644.

Leitão, P., Barbosa, J., Vrba, P., Skobelev, P., Tsarev, A., Kazanskaia, D., 2013]. Multi-agent system approach for the strategic planning in ramp-up production of small lots. 2013 IEEE International Conference on Systems, Man, and Cybernetics, 4743–4748.

Lemaignan, S., Siadat, A., Dantan, J., Semenenko, A., 2006. Mason: a proposal for an ontology of manufacturing domain. IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06), 195–200, http://dx.doi.org/10.1109/DIS.2006.48.

Liagkou, V., Salmas, D., Stylios, C., 2019]. Realizing virtual reality learning environment for industry 4.0. Proc. CIRP 79, 712–717, http://dx.doi.org/10.1016/j.procir.2019.02.025, 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy. http://www.sciencedirect.com/science/article/pii/S2212827119301398.

Masood, T., Egger, J., 2019]. Augmented reality in support of industry 4.0-implementation challenges and success factors. Robot. Comput.-Integr. Manuf. 58, 181–195, http://dx.doi.org/10.1016/j.rcim.2019.02.003 http://www.sciencedirect.com/science/article/pii/S0736584518304101.

Masood, T., Sonntag, P., 2020]. Industry 4.0: adoption challenges and benefits for smes. Comput. Ind. 121, 103261, http://dx.doi.org/10.1016/j.compind.2020.103261.

Mourtzis, D., Zogopoulos, V., Vlachou, E., 2018]. Augmented reality supported product design towards industry 4.0: a teaching factory paradigm. Proc. Manuf. 23, 207–212, http://dx.doi.org/10.1016/j.promfg.2018.04.018, Advanced Engineering Education & Training for Manufacturing Innovation. 8th CIRP Sponsored Conference on Learning Factories (CLF 2018).http://www.sciencedirect.com/science/article/pii/S2351978918304906.

Musen, M.A., 2015]. The protégé project: a look back and a look forward. AI Matt. 1 (4), 4–12, http://dx.doi.org/10.1145/2757001.2757003.

Ordaz, N., Romero, D., Gorecky, D., Siller, H.R., 2015]. Serious games and virtual simulator for automotive manufacturing education & training. Proc. Comput. Sci. 75, 267–274.

Öztemel, E., Gursev, S., 2020]. Literature review of industry 4.0 and related technologies. J. Intell. Manuf. 31 (1), 127–182, http://dx.doi.org/10.1007/s10845-018-1433-8.

Pinto, H.S., Staab, S., Tempich, C., 2004]. Diligent: towards a fine-grained methodology for distributed, loosely-controlled and evolving engineering of ontologies. In: Proceedings of the 16th European Conference on Artificial Intelligence, ECAI'04, IOS Press, Amsterdam, The Netherlands, The Netherlands, pp. 393–397 http://dl.acm.org/citation.cfm?id=3000001.3000084.

Puder, A., Römer, K., Pilhofer, F., 2011]. Distributed Systems Architecture: a Middleware Approach. Elsevier.

Ramírez-Durán, V.J., Berges, I., Illarramendi, A., 2020]. Extruont: an ontology for describing a type of manufacturing machine for industry 4.0 systems. Semant. Web 11 (6), 887–909, http://dx.doi.org/10.3233/SW-200376.

Rao, N.R., Sekharaiah, K.C., 2016]. A methodological review based version control system with evolutionary research for software processes. In: Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, ICTCS'16, Association for Computing Machinery, New York, NY, USA, http://dx.doi.org/10.1145/2905055.2905072.

Riboldi, N., Scurati, G., Pedrini, S., Ferrise, F., Bordegoni, M., 2021]. Improving Maintenance Services through Virtual Reality. World Scientific.

Schweichhart, K., 2016]. Reference Architecturial Model Industrie 4. 0 (rami 4. 0), An Introduction, Available from: https://www.plattform-i40.deI40.

Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., Diakopoulos, N., 2016]. Designing the User Interface: Strategies for Effective Human-Computer Interaction, 6th Ed. Pearson.

Slater, M., 2009]. Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. Philos. Trans. R. Soc. Lond. Ser. B, Biol. Sci. 364, 3549–3557, http://dx.doi.org/10.1098/rstb.2009.0138.

Šormaz, D., Sarkar, A., 2019]. Simpm – upper-level ontology for manufacturing process plan network generation. Robot. Comput.-Integr. Manuf. 55, 183–198, http://dx.doi.org/10.1016/j.rcim.2018.04.002, extended Papers Selected from FAIM2016. http://www.sciencedirect.com/science/article/pii/S0736584517302119.

Suárez-Figueroa, M.C., Gómez-Pérez, A., Fernández-López, M., 2012]. The NeOn Methodology for Ontology Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 9–34, http://dx.doi.org/10.1007/978-3-642-24794-1_2.

Sure, Y., Staab, S., Studer, R., 2004]. On-To-Knowledge Methodology (OTKM). Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 117–132, http://dx.doi.org/10.1007/978-3-540-24750-0_6.

Talhi, A., Fortineau, V., Huet, J., Lamouri, S., 2019]. Ontology for cloud manufacturing based product lifecycle management. J. Intell. Manuf. 30 (5), 2171–2192.

Tao, W., Lai, Z.-H., Leu, M., Yin, Z., Qin, R., 2019]. A self-aware and active-guiding training & assistant system for worker-centered intelligent manufacturing. Manuf. Lett. 21, 45–49.

Tao, W., Lai, Z.-H., Leu, M., 2021]. Manufacturing Assembly Simulations in Virtual and Augmented Reality. World Scientific.

Tomic, D., Davidovic, D., Subic, T., Mesaric, J., Skala, K., 2020]. Improving 3d printing of garments by using hpc cloud. In: 43rd International Convention on Information Communication and Electronic Technology, Mipro 2020, pp. 249–252.

Trunzer, E., Calà, A., Leit ao, P., Gepp, M., Kinghorst, J., Lüder, A., Schauerte, H., Reifferscheid, M., Vogel-Heuser, B., 2019]. System architectures for industrie 4.0 applications. Prod. Eng. 13 (3–4), 247–257.

Uschold, M., 2015]. Ontology and database schema: what's the difference? Appl. Ontol. 10 (3–4), 243–258.

Usman, Z., Young, R.I.M., Chungoora, N., Palmer, C., Case, K., Harding, J., 2011]. A manufacturing core concepts ontology for product lifecycle interoperability. In: van Sinderen, M., Johnson, P. (Eds.), Enterprise Interoperability. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 5–18, http://dx.doi.org/10.1007/978-3-642-19680-5_3.

Vandenbussche, P.-Y., Atemezing, G.A., Poveda-Villalón, M., Vatant, B., 2016]. Linked open vocabularies (LOV): a gateway to reusable semantic vocabularies on the web. Semant. Web 8 (3), 437–452, http://dx.doi.org/10.3233/sw-160213.

van Lopik, K., Sinclair, M., Sharpe, R., Conway, P., West, A., 2020]. Developing augmented reality capabilities for industry 4.0 small enterprises: lessons learnt from a content authoring case study. Comput. Ind. 117, 103208, http://dx.doi.org/10.1016/j.compind.2020.103208 http://www.sciencedirect.com/science/article/pii/S0166361519306207.

Winkelhaus, S., Grosse, E., 2020]. Logistics 4.0: a systematic review towards a new logistics system. Int. J. Prod. Res. 58, 18–43, http://dx.doi.org/10.1080/00207543.2019.1612964.

Wisniewski, D., Potoniec, J., Lawrynowicz, A., Keet, C.M., 2019]. Analysis of ontology competency questions and their formalizations in sparql-owl. J. Web Semant. 59, 100534, http://dx.doi.org/10.1016/j.websem.2019.100534 http://www.sciencedirect.com/science/article/pii/S1570826819300617.

Zhang, L., Longfei, Z., Lei, R., Yuanjun, L., 2019]. Modeling and simulation in intelligent manufacturing. Comput. Ind. 112, 103–123.

Bibliography

[AAAAD16] Abdulrahman M Al-Ahmari, Mustufa H Abidi, Ali Ahmad, and Saber Darmoul. Development of a virtual manufacturing assembly simulation system. *Advances in Mechanical Engineering*, 8(3):1687814016639824, 2016.

[Aba20] Fakhreddine Ababsa. Augmented reality application in manufacturing industry: Maintenance and non-destructive testing (ndt) use cases. In Lucio Tommaso De Paolis and Patrick Bourdot, editors, *Augmented Reality, Virtual Reality, and Computer Graphics*, pages 333–344, Cham, 2020. Springer International Publishing.

[AD06] Farhad Ameri and Debasish Dutta. An upper ontology for manufacturing service description. In *ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 651–661. American Society of Mechanical Engineers, 2006.

[Add19a] Angus Addlesee. Comparison of Linked Data Triplestores: A New Contender, 2019. https://medium.com/wallscope/comparison-of-linked-data-triplestores-a-new-contender-c62ae04901d3.

[Add19b] Angus Addlesee. Comparison of Linked Data Triplestores: Developing the Methodology, 2019. https://medium.com/wallscope/comparison-of-linked-data-triplestores-developing-the-methodology-e87771cb3011.

[ARS16] Knut Alicke, Jürgen Rachor, and Andreas Seyfert. Supply chain 4.0 – the next-generation digital supply chain, 2016. Accessed 28th September 2020.

[AV18] Francesco Antoniazzi and Fabio Viola. RDF Graph Visualization Tools: a Survey. In *23rd Conference of Open Innovations Association, FRUCT 2018, Bologna, Italy, November 13-16, 2018*, pages 25–36. IEEE, 2018.

[BB97] Willem Nico Borst and W.N. Borst. *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*. PhD thesis, UT, Netherlands, September 1997.

[BGA13] Josep Maria Brunetti, Roberto García, and Sören Auer. From Overview to Facets and Pivoting for Interactive Exploration of Semantic Web Data. *Int. J. Semant. Web Inf. Syst.*, 9(1):1–20, January 2013.

[BKS+12] Raphael Barbau, Sylvère Krima, Rachuri Sudarsan, Anantha Narayanan, Xenia Fiorentini, Sebti Foufou, and Ram D. Sriram. Ontostep: Enriching product model data using ontologies. *Computer-Aided Design*, 44(6):575–590, 2012.

[BLHL01] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 284(5):34–43, 2001.

[BLK18] Christian Block, Dominik Lins, and Bernd Kuhlenkötter. Approach for a simulation-based and event-driven production planning and control in decentralized manufacturing execution systems. *Procedia CIRP*, 72:1351–1356, 2018. 51st CIRP Conference on Manufacturing Systems.

[BPP19] Nikos Bikakis, George Papastefanatos, and Olga Papaemmanouil. Big Data Exploration, Visualization and Analytics. *Big Data Research*, 18:100123, 12 2019.

[BV19] Eleonora Bottani and Giuseppe Vignali. Augmented reality technology in the manufacturing industry: A review of the last decade. *IIE Transactions*, 51:284–310, 03 2019.

[C+13] World Wide Web Consortium et al. Sparql 1.1 overview. 2013.

[CAS20] CASP. Reflect, 2020. Accessed 28th September 2020.

[CBGG97] Anthony G. Cohn, Brandon Bennett, John Gooday, and Nicholas Mark Gotts. Qualitative spatial representation and reasoning with the region connection calculus. *GeoInformatica*, 1(3):275–316, Oct 1997.

[CCL+16] Doung Chankhihort, Sung Soo Choi, Gyu Jung Lee, Byung Mook Im, Da-Mi Ahn, Eun-Suk Choi, Aziz Nasridinov, Sun-Ok Kwon, Sang-Hyun Lee, Jeong-Tae Kang, Kyu-Tae Park, and Kwan-Hee Yoo. Integrative manufacturing data visualization using calendar view map. In *Eighth International Conference on Ubiquitous and Future Networks, ICUFN 2016, Vienna, Austria, July 5-8, 2016*, pages 114–116. IEEE, 2016.

[CCLB97] Tiziana Catarci, Maria F Costabile., Stefano Levialdi, and Carlo Batini. Visual Query Systems for Databases: A Survey. *Journal of Visual Languages & Computing*, 8(2):215–260, 1997.

[Clo20] CloudiFacturing. Cloudification of production engineering for predictive digital manufacturing, 2020. Accessed 28th September 2020.

[CMA12] Diego Valerio Camarda, Silvia Mazzini, and Alessandro Antonuccio. LodLive, exploring the web of data. In Valentina Presutti and Helena Sofia Pinto, editors, *I-SEMANTICS 2012 - 8th International Conference on Semantic Systems, I-SEMANTICS '12, Graz, Austria, September 5-7, 2012*, pages 197–200. ACM, 2012.

[CPS+19] Georgios Chatzigeorgakidis, Kostas Patroumpas, Dimitrios Skoutas, Spiros Athanasiou, and Spiros Skiadopoulos. Visual Exploration of Geolocated Time Series with Hybrid Indexing. *Big Data Research*, 15, 02 2019.

[D2L20] D2LAB. Improving quality control and maintenance at manufacturing smes using big data analytics, 2020. Accessed 28th September 2020.

[EF13] Bahaeddin Eravci and Hakan Ferhatosmanoglu. Diversity based relevance feedback for time series search. *Proceedings of the VLDB Endowment*, 7:109–120, 10 2013.

[Eis89] Kathleen M. Eisenhardt. Building theories from case study research. *The Academy of Management Review*, 14(4):532–550, 1989.

[ETS17] ETSI. SmartM2M; smart applicances; reference ontology and oneM2M mapping. Technical specification TS 103 264 V2.1.1, ETSI, 2017.

[Flo67] Robert W. Floyd. Nondeterministic algorithms. *J. ACM*, 14(4):636–644, October 1967.

[Für16] Christian Fürber. *Semantic Technologies*, pages 56–68. Springer Fachmedien Wiesbaden, Wiesbaden, 2016.

[GBB+19] Liang Gong, Jonatan Berglund, Asa-Fast Berglund, Bjorn Johansson, Zhiping Wang, and Tobias Borjessong. Development of virtual reality support to factory layout planning. *International Journal on Interactive Design and Manufacturing*, 2019.

[GF12] M. Garetti and L. Fumagalli. P-pso ontology for manufacturing systems. *IFAC Proceedings Volumes*, 45(6):449 – 456, 2012. 14th IFAC Symposium on Information Control Problems in Manufacturing.

[GFPC+17] Mar Gonzalez-Franco, Rodrigo Pizarro, Julio Cermeron, Katie Li, Jacob Thorn, Windo Hutabarat, Ashutosh Tiwari, and Pablo Bermell-Garcia. Immersive mixed reality for manufacturing training. *Frontiers in Robotics and AI*, 4, 02 2017.

[GLDK12] André Gellrich, Daniel Lunkwitz, Alexander Dennert, and Klaus Kabitzsch. Rule-driven manufacturing control based on ontologies. In *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies Factory Automation (ETFA 2012)*, pages 1–8, 2012.

[GMW04] H.F. Giles, E.M. Mount, and J.R. Wagner. *Extrusion: The Definitive Processing Guide and Handbook*. Plastics Design Library. Elsevier Science, 2004.

[Gru93] Thomas R Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.

[Grü09] Michael Grüninger. *Using the PSL Ontology*, pages 423–443. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[Gua97] Nicola Guarino. Semantic matching: Formal ontological distinctions for information organization, extraction, and integration. In Maria Teresa Pazienza, editor, *Information Extraction A Multidisciplinary Approach to an Emerging Information Technology*, pages 139–170, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.

[GZC+18] Ziyue Guo, Dong Zhou, Jiayu Chen, Jie Geng, Chuan Lv, and Shengkui Zeng. Using virtual reality to support the product's maintainability design: Immersive maintainability verification and evaluation system. *Computers in Industry*, 101:41 – 50, 2018.

[Hev07] Alan Hevner. A three cycle view of design science research. *Scandinavian Journal of Information Systems*, 19, 01 2007.

[HJC+18] Armin Haller, Krzysztof Janowicz, Simon J.D. Cox, Maxime Lefrançois, Kerry Taylor, Danh Le Phuoc, Joshua Lieberman, Raúl García-Castro, Rob Atkinson, and Claus Stadler. The modular SSN ontology: A joint w3c and OGC standard specifying the semantics of sensors, observations, sampling, and actuation. *Semantic Web*, 10(1):9–32, dec 2018.

[HK11] Ralph Hodgson and Paul J Keller. Qudt-quantities, units, dimensions and data types in owl and xml. *Online (September 2011) http://www.qudt.org*, 2011.

[HKWT18] Andreas Holzinger, Peter Kieseberg, Edgar Weippl, and A. Min Tjoa. Current advances, trends and challenges of machine learning and knowledge extraction: From machine learning to explainable ai. In Andreas Holzinger, Peter Kieseberg, A Min Tjoa, and Edgar Weippl, editors, *Machine Learning and Knowledge Extraction*, pages 1–8, Cham, 2018. Springer International Publishing.

[HLBE14] Florian Haag, Steffen Lohmann, Steffen Bold, and Thomas Ertl. Visual SPARQL querying based on extended filter/flow graphs. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, pages 305–312. ACM, 2014.

[HMPR04] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS Quarterly*, 28(1):75–105, 2021/05/03/ 2004.

[HRR+18] Dominik Herr, Jan Reinhardt, Guido Reina, Robert Krüger, Rafael V. Ferrari, and Thomas Ertl. Immersive modular factory layout planning using augmented reality. *Procedia CIRP*, 72:1112 – 1117, 2018.

[ILM+20] Nadeem Iftikhar, Bartosz Piotr Lachowicz, Akos Madarasz, Finn Ebertsen Nordbjerg, Thorkil Baattrup-Andersen, and Karsten Jeppesen. Real-time Visualization of Sensor Data in Smart Manufacturing using Lambda Architecture. In Slimane Hammoudi, Christoph Quix, and Jorge

Bernardino, editors, *Proceedings of the 9th International Conference on Data Science, Technology and Applications, DATA 2020, Lieusaint, Paris, France, July 7-9, 2020*, pages 215–222. SciTePress, 2020.

[IWZ+18] E. Icer, S. Weisenburger, M. Zechmair, A. Peters, G. Michalos, A. Knoll, and S. Makris. Esmera - european smes robotics applications. In *ISR 2018; 50th International Symposium on Robotics*, pages 1–4, 2018.

[JSH+20] Szilárd Jaskó, Adrienn Skrop, Tibor Holczinger, Tibor Chován, and János Abonyi. Development of manufacturing execution systems in accordance with industry 4.0 requirements: A review of standard- and ontology-based methodologies and tools. *Computers in Industry*, 123:103300, 2020.

[JSHL19] Eeva Järvenpää, Niko Siltala, Otto Hylli, and Minna Lanz. The development of an ontology for describing the capabilities of manufacturing resources. *Journal of Intelligent Manufacturing*, 30(2):959–978, Feb 2019.

[KS18] Jan Martin Keil and Sirko Schindler. Comparison and evaluation of ontologies for units of measurement. *Semantic Web*, 10(1):33–51, dec 2018.

[Kus18] Andrew Kusiak. Smart manufacturing. *International Journal of Production Research*, 56(1-2):508–517, 2018.

[KWH13] Henning Kagermann, Wolfgang Wahlster, and Johannes Helbig. Recommendations for implementing the strategic initiative industrie 4.0 – securing the future of german manufacturing industry. Final report of the industrie 4.0 working group, acatech – National Academy of Science and Engineering, München, apr 2013.

[LFK+14] Heiner Lasi, Peter Fettke, Hans-Georg Kemper, Thomas Feld, and Michael Hoffmann. Industry 4.0. *Business & Information Systems Engineering*, 6(4):239–242, Aug 2014.

[Llo17] Jorge Lloret-Gazo. A Survey on Visual Query Systems in the Web Era (extended version). *CoRR*, abs/1708.00192, 2017. http://arxiv.org/abs/1708.00192.

[LNHE16] Steffen Lohmann, Stefan Negru, Florian Haag, and Thomas Ertl. Visualizing ontologies with VOWL. *Semantic Web*, 7:399–419, 05 2016.

[LSDS06] S. Lemaignan, A. Siadat, J. . Dantan, and A. Semenenko. Mason: A proposal for an ontology of manufacturing domain. In *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, pages 195–200, June 2006.

[LSS19] Vasiliki Liagkou, Dimitrios Salmas, and Chrysostomos Stylios. Realizing virtual reality learning environment for industry 4.0. *Procedia CIRP*, 79:712 – 717, 2019. 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy.

[ME19] Tariq Masood and Johannes Egger. Augmented reality in support of industry 4.0—implementation challenges and success factors. *Robotics and Computer-Integrated Manufacturing*, 58:181 – 195, 2019.

[MGH+09] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, Carsten Lutz, et al. Owl 2 web ontology language profiles. *W3C recommendation*, 27:61, 2009.

[MM04] F. Manola and Eric Miller. Rdf primer: W3c recommendation. *Decision Support Systems - DSS*, 01 2004.

[Mow09] David C. Mowery. Plus ca change: Industrial R&;D in the "third industrial revolution". *Industrial and Corporate Change*, 18(1):1–50, 01 2009.

[MVH+04] Deborah L McGuinness, Frank Van Harmelen, et al. Owl web ontology language overview. *W3C recommendation*, 10(10):2004, 2004.

[MZV18] Dimitris Mourtzis, Vasilios Zogopoulos, and Ekaterini Vlachou. Augmented reality supported product design towards industry 4.0: a teaching factory paradigm. *Procedia Manufacturing*, 23:207 – 212, 2018. "Advanced Engineering Education & Training for Manufacturing Innovation"8th CIRP Sponsored Conference on Learning Factories (CLF 2018).

[ORGS15] Nestor Ordaz, David Romero, Dominic Gorecky, and Hector R. Siller. Serious games and virtual simulator for automotive manufacturing education & training. *Procedia Computer Science*, 75:267–274, 2015.

[PH12] Matthew Perry and John Herring. Ogc geosparql-a geographic query language for rdf data. *OGC implementation standard*, 40, 2012.

[PVGPSF14] María Poveda-Villalón, Asuncion Gomez-Perez, and Mari Carmen Suárez-Figueroa. Oops! (ontology pitfall scanner!): An on-line tool for ontology evaluation. *International Journal on Semantic Web and Information Systems*, 10:7–34, 04 2014.

[RCC92] David A. Randell, Zhan Cui, and Anthony G. Cohn. A spatial logic based on regions and connection. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, KR'92, page 165–176, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[RLS20] Jessica Rubart, Benjamin Lietzau, and Patrick Söhlke. Analyzing Manufacturing Data in a Digital Control Room Making Use of Semantic Annotations. In *IEEE 14th International Conference on Semantic Computing, ICSC 2020, San Diego, CA, USA, February 3-5, 2020*, pages 434–438. IEEE, 2020.

[RSP+21] Nicola Riboldi, Giulia Scurati, Simone Pedrini, Francesco Ferrise, and Monica Bordegoni. *Improving Maintenance Services through Virtual Reality*. World Scientific, 2021.

[RvAT13] Hajo Rijgersberg, Mark van Assem, and Jan Top. Ontology of units of measure and related concepts. *Semantic Web*, 4(1):3–13, 2013.

[SBF98] Rudi Studer, V.Richard Benjamins, and Dieter Fensel. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1):161–197, 1998.

[Sik17] Leslie F. Sikos. A novel ontology for 3d semantics: ontology-based 3d model indexing and content-based video retrieval applied to the medical domain. *International Journal of Metadata, Semantics and Ontologies*, 12(1):59–70, 2017.

[SKZ+18] Ahmet Soylu, Evgeny Kharlamov, Dmitriy Zheleznyakov, Ernesto Jiménez-Ruiz, Martin Giese, Martin G. Skjæveland, Dag Hovland, Rudolf Schlatte, Sebastian Brandt, Hallstein Lie, and Ian Horrocks. OptiqueVQS: A visual query system over ontologies for industry. *Semantic Web*, 9(5):627–660, 2018.

[Sla09] Mel Slater. Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 364:3549–57, 12 2009.

[SS19] Dušan Sormaz and Arkopaul Sarkar. SIMPM – Upper-level ontology for manufacturing process plan network generation. *Robotics and Computer-Integrated Manufacturing*, 55:183 – 198, 2019. Extended Papers Selected from FAIM2016.

[TDS+20] D. Tomic, D. Davidović, T. Subic, J. Mesaric, and K. Skala. Improving 3d printing of garments by using hpc cloud. In *43rd International Convention on Information Communication and Electronic Technology, Mipro 2020*, pages 249–252, 2020.

[TLL+19] Wenjin Tao, Ze-Hao Lai, M. Leu, Z. Yin, and R. Qin. A self-aware and active-guiding training & assistant system for worker-centered intelligent manufacturing. *Manufacturing letters*, 21:45–49, 2019.

[TLL21] Wenjin Tao, Ze-Hao Lai, and Ming Leu. *Manufacturing Assembly Simulations in Virtual and Augmented Reality*. World Scientific, 2021.

[TS15] Krishnaprasad Thirunarayan and Amit Sheth. Semantics-empowered big data processing with applications. *AI Magazine*, 36(1):39–54, Mar. 2015.

[UYC+11] Zahid Usman, Robert Ian Marr Young, Nitishal Chungoora, Claire Palmer, Keith Case, and Jenny Harding. A manufacturing core concepts ontology for product lifecycle interoperability. In Marten van Sinderen and Pontus Johnson, editors, *Enterprise Interoperability*, pages 5–18, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[VC11] Emanuele Della Valle and Stefano Ceri. *Querying the Semantic Web: SPARQL*, pages 299–363. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[Yin94] Robert K. Yin. Discovering the future of the case study. method in evaluation research. *Evaluation Practice*, 15(3):283–290, 1994.

[ZLLY19] Lin Zhang, Zhou Longfei, Ren Lei, and Laili Yuanjun. Modeling and simulation in intelligent manufacturing. *Computers in Industry*, 112:103–123, 2019.