Semantic Web 0 (0) 1
IOS Press

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

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

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 the components of an extruder, the spatial connections, features, and 3D representations of those components, and finally about 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 experts from a manufacturing company.

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 [1], mechatronics for advanced manufacturing systems, manufacturing strategies, knowledge-workers and modelling, simulation and forecasting methods and tools. 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 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*

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

[1] In which some material is forced through a series of dies in order to create a desired shape.

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] development environment. Alignments with terms from other ontologies such as MSDL[2], SAREF4INMA[3], GEOSPARQL[4], OM [5], SOSA/SSN[6] and 3DMO[7] have been defined.

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 the other modules should be adapted to meet the requirements of the new type of machine; 2) Expressiveness of Spatial Connections. It incorporates a hierarchical description of possible relations in Region Connection Calculus (RCC) 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, an ontology-based visual query system developed for a Smart Manufacturing scenario, whose core element is the *ExtruOnt* ontology, will bring the following main benefits to the different types of workers of a manufacturing plant:

- *Novice workers.* The 3D rendering of an extruder machine will be obtained from descriptions in the ontology and it 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 the steps that we followed to develop the *ExtruOnt* ontology. Then, the modules that constitute the *ExtruOnt* ontology are described. Next, we show the results of an evaluation process considering three different approaches: pitfall evaluation, quality criteria and ontology metrics. Finally, some conclusions are presented.

2. Development of the *ExtruOnt* ontology

In Smart Manufacturing scenarios different ontologies have been defined with distinct purposes such as: PSL (Process Specification Language) ontology [8], for describing the fundamental concepts of manufacturing processes; MASON (Manufacturing's Semantics Ontology) ontology [9], which shared the same goals with PSL, but was modelled with OWL; MSDL (Manufacturing Service Description Language) ontology [2], defined to provide a common semantic model for describing manufacturing services; P-PSO (Politecnico di MilanoProduction Systems) ontology [10], which considers four aspects in the manufacturing domain (Product, physical aspect, technological aspect and control aspect) for information exchange, design, control, simulation and other applications; and OntoSTEP (ONTOlogy of Standard for the Exchange of Product model data), which contains product information mainly related to geometry [11]. However, we did not found any ontology that described industrial machines as a whole in detail, and more particularly, extruder machines. For that reason we built the *ExtruOnt* ontology, which is aligned with concepts included in an ontology-based context model for industry presented in [12].

Different methodologies can be found in the literature to adequately develop well-founded ontologies. Among those methodologies we can mention On-To-Knowledge [13], Diligent [14] and NeOn [15]. We selected the NeOn methodology because it suggests a variety of paths to develop an ontology.

In collaboration with experts in the domain of extrusion we created the Ontology Requirements Specification Document (ORSD) that contains among others, the purpose of the *ExtruOnt* ontology, its scope and

[2]https://www.w3.org/TR/owl2-overview/
[3]https://protege.stanford.edu/

the Competency Questions(CQs), see Table 1. Several competency questions were identified. 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 divided 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 of NeOn).

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.

- *Components of an extruder*: There exist a few ontologies that describe manufacturing resources. Among them, we can mention MCCO (Manufacturing Core Concepts Ontology) [16], and more recently SAREF4INMA [3], (a SAREF [17] extension for the industry and manufacturing domain, that still is in an initial state in its development). However, they present shortcomings to describe specific machine types with a fine-grained detail. Therefore, in order to describe the components we relied 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 [18]. 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 [19], 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 of NeOn). Also, the PartOf[4] ontology design pattern was used in order to specify parthood between the extruder and its components, as well as between different parts that constitute each component (scenario 7 of NeOn).
- *Spatial connections between components*: In the specialized literature we can find the Region Connection Calculus (RCC)[20, 21], 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 that consists of 8 basic relations that are possible between two regions. Different ontologies have tried to represent the RCC descriptions (GeoSPARQL[4], Spatial Relations Ontology[5], NeoGeo Spatial Ontology [6]) but none of them is adequate for answering competency question CQ2.2. Thus, we opted for a twofold approach: we selected the GeoSPARQL ontology, which models the RCC8 relations (scenario 4 of NeOn), because is the base for the other spatial ontologies, and we incorporated information about other RCC spatial relations obtained from the aforementioned non-ontological RCC resources (scenario 2 of NeOn).
- *Features of the components*: Different ontologies can be found for representing measurements. In [22], a comparison and evaluation of eight different ontologies is presented using a collection of scripts to get descriptive statistics. Based on the mentioned article, two ontologies were considered as a potential base for *OM4ExtruOnt*: QUDT[7] [23] and OM[8] [5]. 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 to model concepts and relations in the context of food research and it was the largest unit ontology compared. In the 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 4 of NeOn).
- *3D representation of components*: We selected the 3D Modeling Ontology (3DMO) [7] because this ontology maps the entire XSD-based vocabulary of the industry standard X3D[9] (ISO/IEC 19775-19777) to OWL 2. Therefore, it can be

[4]http://ontologydesignpatterns.org

[5]http://data.ordnancesurvey.co.uk/ontology/spatialrelations/

[6]http://geovocab.org/doc/neogeo/

[7]http://www.linkedmodel.org/catalog/qudt/1.1/index.html

[8]https://enterpriseintegrationlab.github.io/icity/OM/doc/index-en.html

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

used for the representation, annotation, and efficient indexing of 3D models (scenario 4 of NeOn).
- *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[6] ontology defines general concepts about sensors, which can be specialized with information obtained from non-ontological resources about extruders [18] to reflect the specificities of the extrusion domain (scenarios 4 and 2 of NeOn respectively).

One module was developed for each of the five dimensions, which altogether form the *ExtruOnt*[10] ontology. They were implemented in OWL 2 DL using Protégé.

3. Ontology modules

As said before, *ExtruOnt* is divided in five modules aiming to describe several characteristics of an extruder machine (see Fig. 1).

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

3.1. components4ExtruOnt

The *components4ExtruOnt*[11] module is the main module of the *ExtruOnt* ontology and is intended to describe the components of an extruder. According to [18] 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 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`[12] 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. 2. To facilitate integration with other domain ontologies, the terms `saref4inma:ProductEquipment`[13] and `MSDL:MSDL_0000033`[14] (labeled "Product equipment" and "production machine" respectively) were included as superclasses of `Extruder`.

Moreover, the specialization of each component was represented using `rdfs:subClassOf` relations. An example is illustrated in Fig. 3.

With respect to the extrusion head, the classification that can be found in [19] was used to provide a detailed representation of this component. Figs. 4 and 5 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?

A SPARQL query to answer the competency question CQ1.4 is as follows[15]:

```
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.
```

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

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

[12] http://www.ontologydesignpatterns.org/cp/owl/partof.owl

[13] https://w3id.org/def/saref4inma

[14] http://infoneer.wp.txstate.edu/ontology-download/msdl-ontology/

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

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 250 degrees Celsius in extruder E10?
  * ...

7. **Pre-glossary of terms**

Extruder, feed system, observation, sensor, tangential proper part, measure, 3D canvas ...

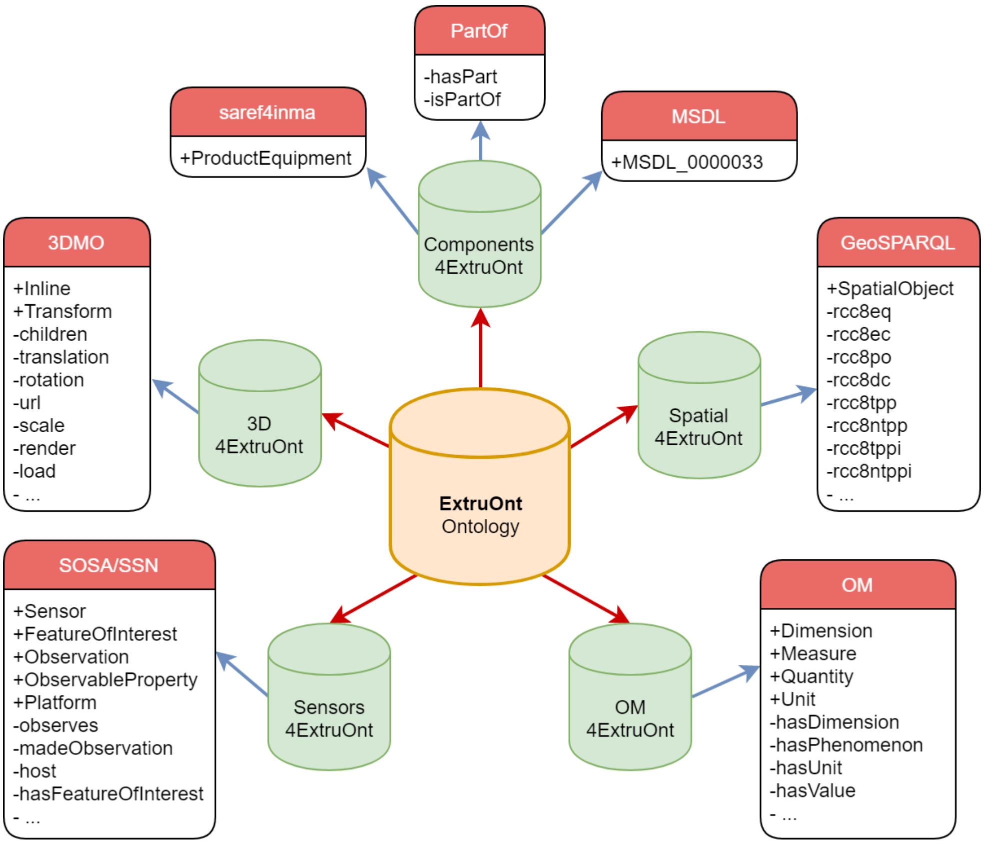

Fig. 1. *ExtruOnt* ontology diagram showing the reuse of terms from other domain ontologies.

```
    ?motor01 a c4e:AC_motor
        }
```

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.

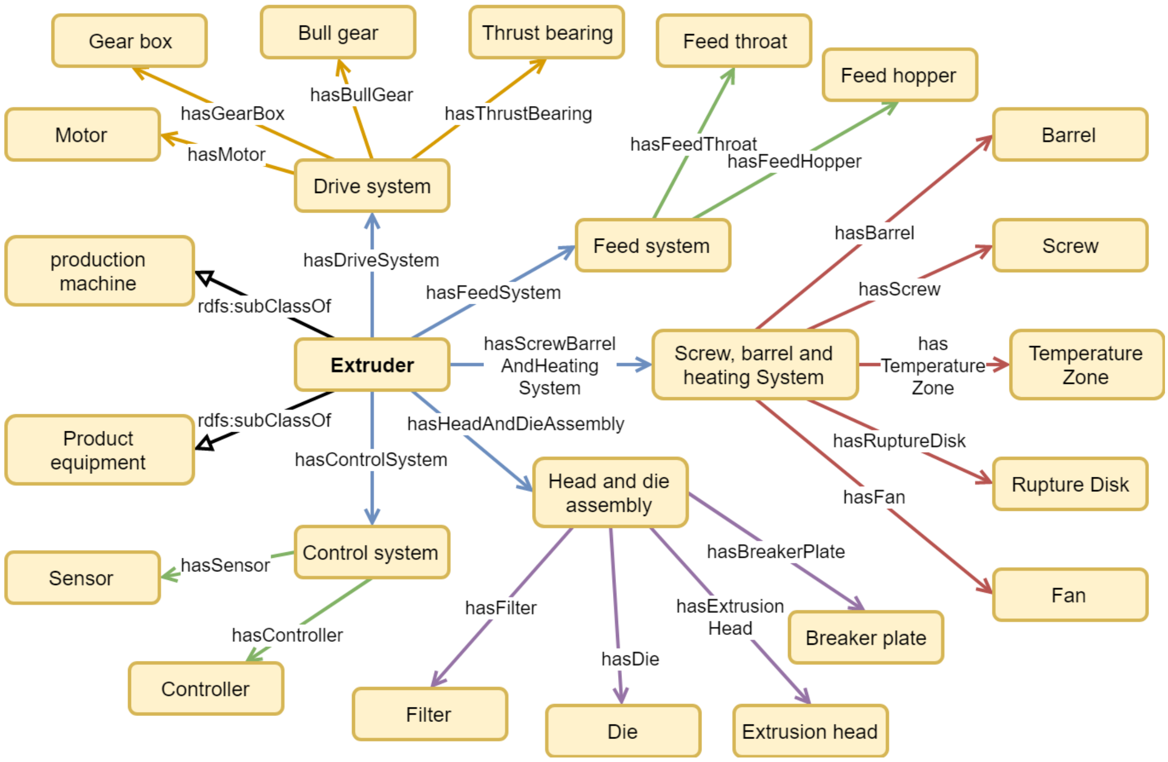

Fig. 2. Some components of an extruder.

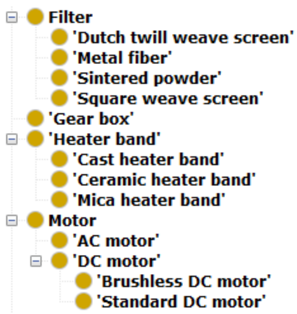

Fig. 3. Excerpt of the class hierarchy of the components.

3.2. *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), Partially 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. 6.

For the *spatial4ExtruOnt*[16] module, a submodule of the GeoSPARQL ontology was used, which contains the `SpatialObject` main class and the object properties referencing to the RCC8 relations. 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 classifica-

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

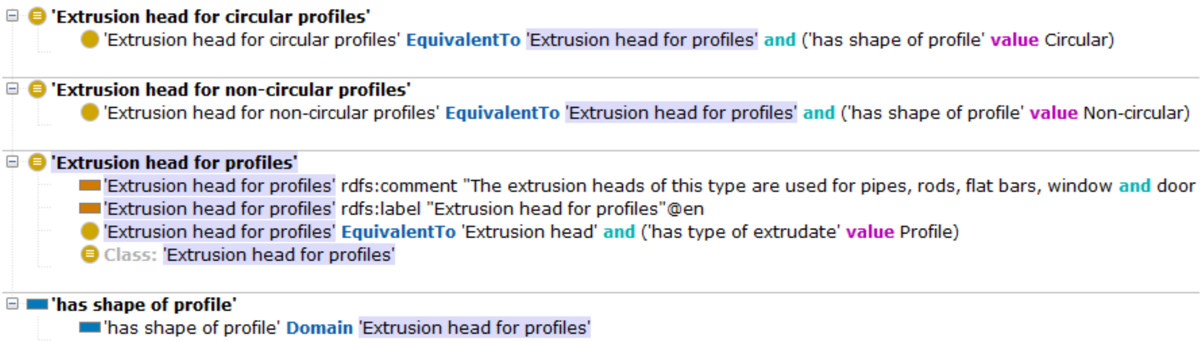

Fig. 4. Definition of the Extrusion head for profiles.

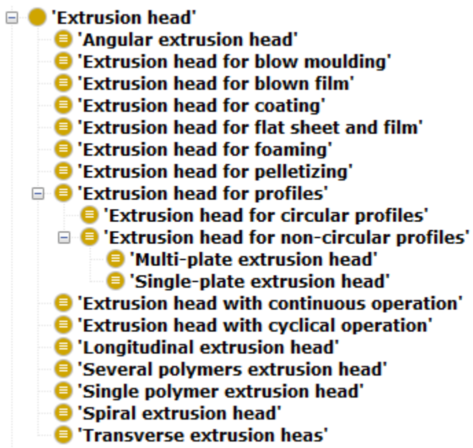

Fig. 5. Subclasses of Extrusion head.

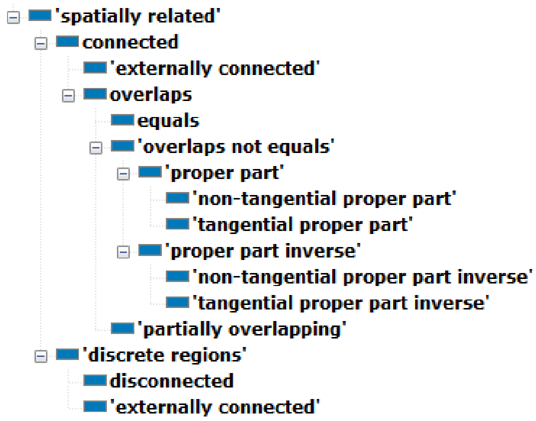

Fig. 7. Object property hierarchy in *spatial4ExtruOnt*.

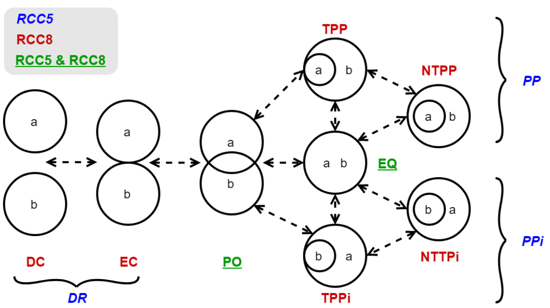

Fig. 6. RCC5 and RCC8 relations.

tion 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. 7.

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 result 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. 8).

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. 9).

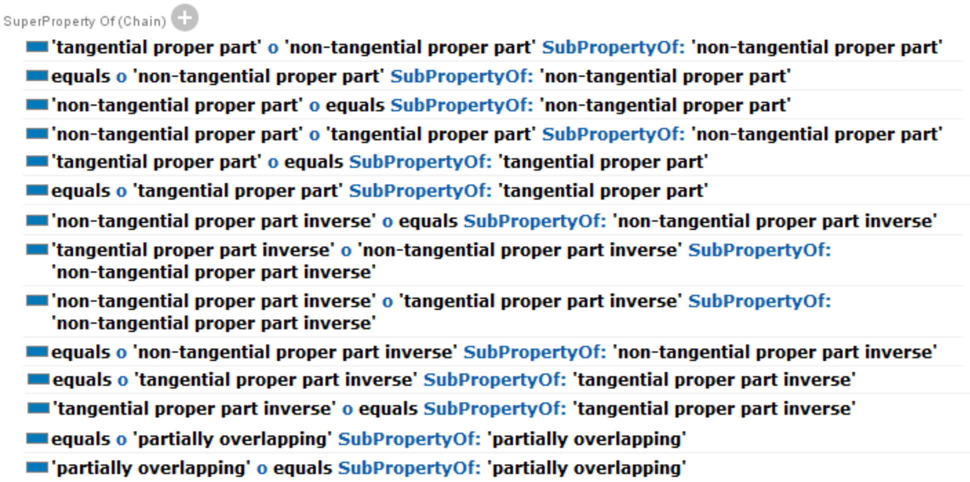

Fig. 8. Property chains defined in *spatial4ExtruOnt*

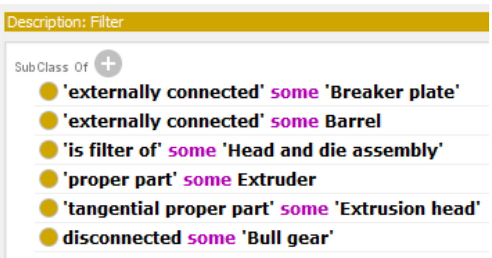

Fig. 9. Excerpt of the Filter class description.

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.

3.3. *OM4ExtruOnt*

The objective of the *OM4ExtruOnt*[17] 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*

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

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. 10 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.

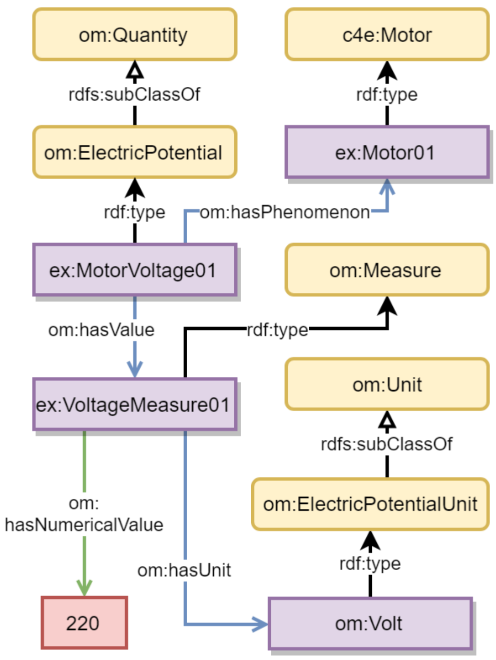

Fig. 10. Example of definition of a measure for the feature Motor voltage.

Once the features of the components are defined using the ontology, 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?
– 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 using 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.

3.4. 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 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. 11). On the contrary, a 3D representation of an extruder allows the viewer's , facilitating to move, rotate, zoom in and zoom out. This advantage provides each user with a personalized experience (Fig. 12).

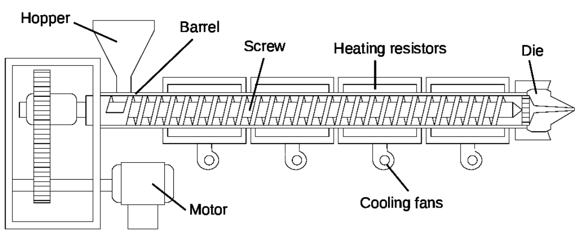

Fig. 11. 2D representation of the components of an extruder.

The purpose of the *3D4ExtruOnt*[18] 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. This is specially useful for novice workers who want to recognize the extruder and its components and for domain

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

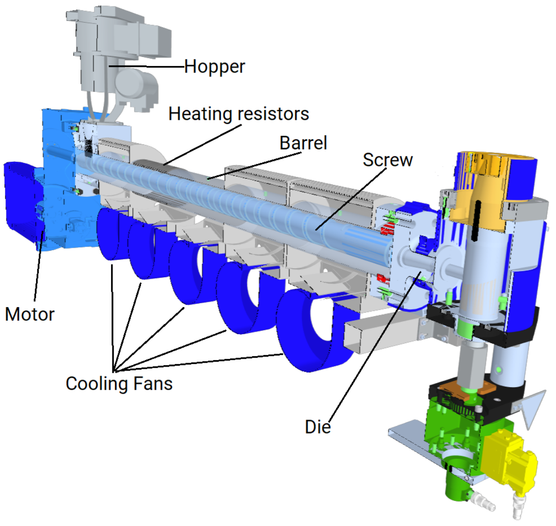

Fig. 12. 3D representation of the components of an extruder.

experts who want to detect possible failures in component design.

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:

- 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. An example of the 3D positioning of the motor is shown in Fig. 13. The 3D model is loaded using the `url` property of `Inline` class and the model is placed on a 3D canvas using the `translation` property of `Transform` class.

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

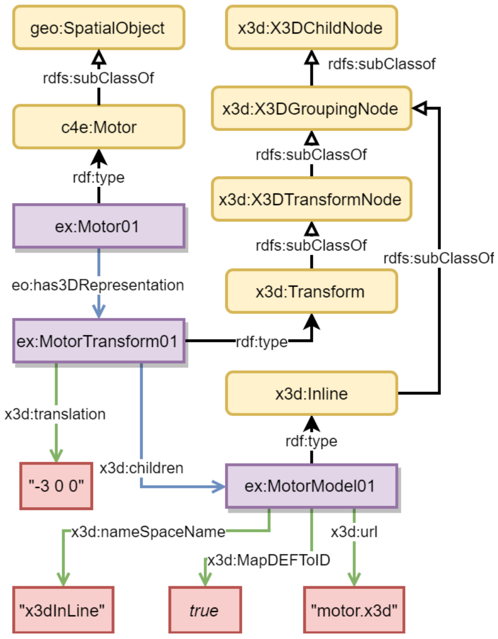

Fig. 13. Definition of motor model location in a 3D canvas.

- 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.

3.5. *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*[19] module imports the *SOSA/SSN* [6] 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 comsumption 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. An excerpt of the module can be found in Fig. 14.

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. 15).

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

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?
- 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 250 degrees Celsius 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 .
  ?obs sosa:hasSimpleResult ?resultValue ;
    sosa:resultTime ?resultTime .
  filter(?resultValue >= "250"^^xsd:double) .
  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

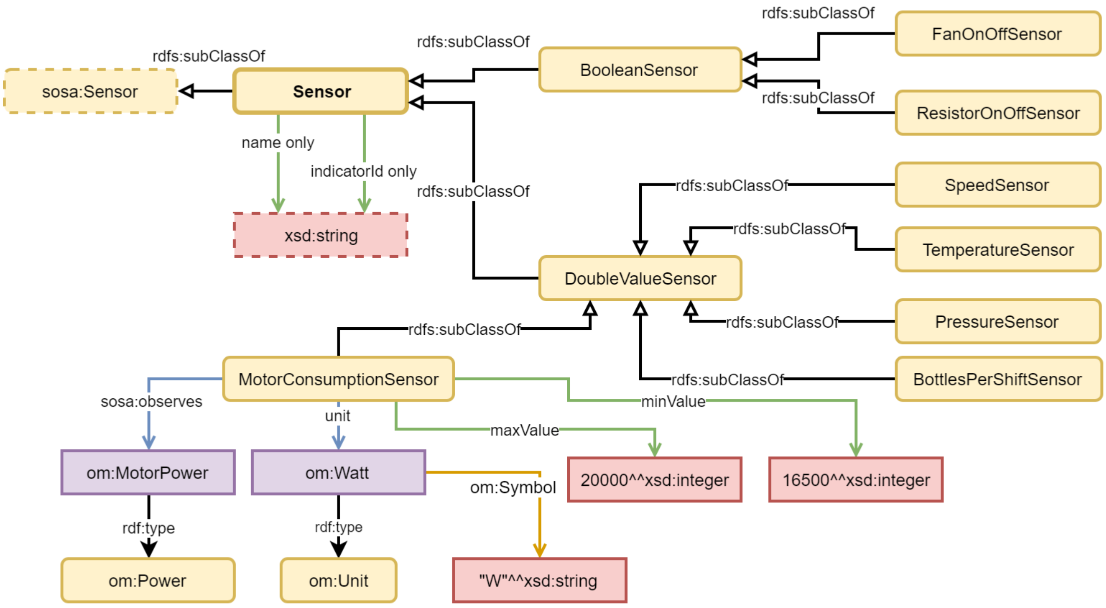

Fig. 14. Excerpt of the *sensors4ExtruOnt* module showing some classes and properties related to sensors.

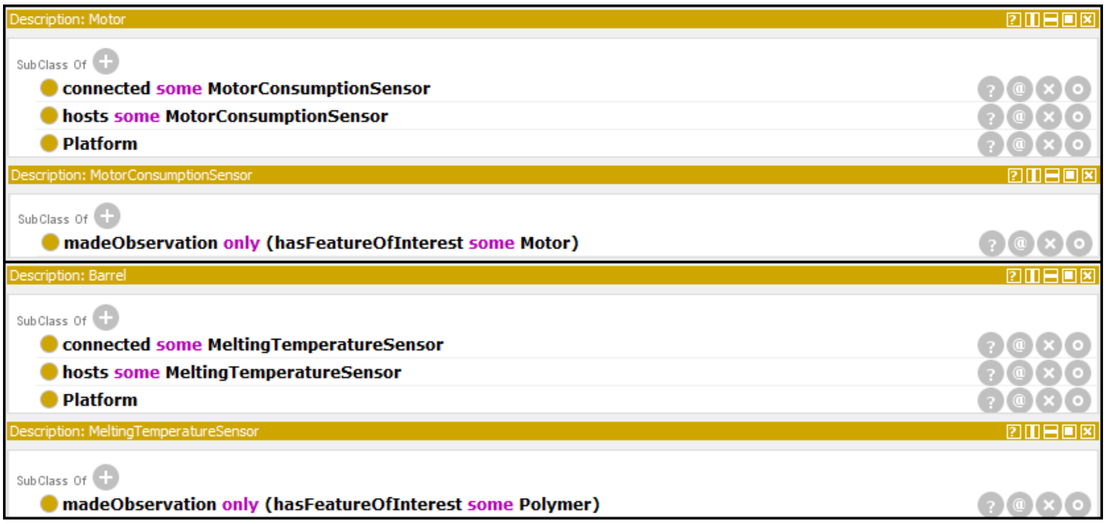

Fig. 15. Excerpt of the descriptions of classes Motor, MotorConsumptionSensor, Barrel and MeltingTemperatureSensor

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.

4. Evaluation

Three different approaches were taken into account for the evaluation of the *ExtruOnt* ontology. To begin with, a pitfall evaluation was carried out using the Ontology Pitfall Scanner (OOPS!) [24]. Next, using criteria related to ontology quality, a detailed evaluation was performed checking all the criteria that have been considered in the development process. Lastly, the ontology metrics are presented.

4.1. 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 ontol-

Table 2

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#ProductEquipment`
	`http://xmlns.com/foaf/0.1/Agent`
	`http://infoneer.txstate.edu/ontology/MSDL_0000033`
	`http://www.w3.org/2006/time#TemporalEntity`
	`http://purl.org/vocommons/voaf#Vocabulary`

ogy 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 2 presents the evaluation summary made by OOPS!.

4.2. Ontology quality

The criteria used for the evaluation of the ontology are described in [25]. They are presented as part of a common framework for aspects of ontology evaluation. These criteria are listed below with an explanation of their application in *ExtruOnt*.

- **Accuracy:** The ontology creation was assisted by domain experts from a company that manufactures extruders. 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 [18, 19], *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.
- **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.
- **Consistency:** No inconsistencies were found in *ExtruOnt* when reasoning was performed. The reasoner used was Fact++[20].

4.3. Ontology metrics

The ontology metrics were extracted from Protégé and they are related to the amount of axioms, classes, properties and individuals in the ontology. They are listed in table 3.

5. Conclusions

The purpose of this paper is to present the *ExtruOnt* ontology, which contains terms to describe a

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

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 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 from three different points of view and the results show that the modules are correct from a design viewpoint, and they present a high degree of adaptability. Furthermore, it is aligned with related ontologies.

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] EFFRA, Factories of the Future Roadmap, Technical Report, EFFRA, 2019. https://www.effra.eu/factories-future-roadmap.

[2] 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.

[3] ETSI, SmartM2M; SAREF extension investigation; Requirements for industry and manufacturing domains, Technical specification, TS 103 507 V1.1.1, ETSI, 2018.

[4] R. Battle and D. Kolas, Geosparql: enabling a geospatial semantic web, *Semantic Web Journal* **3**(4) (2011), 355–370.

[5] H. Rijgersberg, M. Van Assem and J. Top, Ontology of units of measure and related concepts, *Semantic Web Journal* **4**(1) (2013), 3–13.

[6] A. Haller, K. Janowicz, S.J. Cox, M. Lefrançois, K. Taylor, D. Le Phuoc, J. Lieberman, R. García-Castro, R. Atkinson and C. Stadler, The SOSA/SSN ontology: a joint WeC and OGC standard specifying the semantics of sensors observations actuation and sampling, in: *Semantic Web*, Vol. 1, IOS Press, 2018, pp. 1–19.

[7] 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.

[8] M. Grüninger, Using the PSL Ontology, in: *Handbook on Ontologies*, Springer, 2009, pp. 423–443.

[9] S. Lemaignan, A. Siadat, J.-Y. Dantan and A. Semenenko, MASON: A proposal for an ontology of manufacturing domain, 2006, pp. 195–200. ISBN 0-7695-2589-X. doi:10.1109/DIS.2006.48.

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

[11] 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.

[12] 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, Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 22nd International Conference, KES-2018, Belgrade, Serbia. doi:https://doi.org/10.1016/j.procs.2018.08.001. http://www.sciencedirect.com/science/article/pii/S1877050918312791.

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

[14] 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, The Netherlands, The Netherlands, 2004, pp. 393–397. ISBN 978-1-58603-452-8. http://dl.acm.org/citation.cfm?id=3000001.3000084.

[15] M.C. Suárez-Figueroa, A. Gómez-PÃľrez 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, 2012, pp. 9–34. ISBN 978-3-642-24793-4. http://dblp.uni-trier.de/db/books/daglib/0028799.html#Suarez-FigueroaGF12.

[16] 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 Heidelberg, Berlin, Heidelberg, 2011, pp. 5–18.

[17] ETSI, SmartM2M; Smart Applicances; Reference Ontology and oneM2M Mapping, Technical specification, TS 103 264 V2.1.1, ETSI, 2017.

[18] H.F. Giles Jr, E.M. Mount III and J.R. Wagner Jr, *Extrusion: the definitive processing guide and handbook*, William Andrew, 2004.

[19] J. Sikora, Design of Extrusion Heads, *Department of Polymer Processing, Lublin University of Technology* (2008).

[20] D.A. Randell, Z. Cui and A.G. Cohn, A spatial logic based on regions and connection., *KR* **92** (1992), 165–176.

[21] 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.

[22] J.M. Keil and S. Schindler, Comparison and evaluation of ontologies for units of measurement, *Semantic Web Journal* (2018), 1–19.

[23] R. Hodgson and P.J. Keller, QUDT-quantities, units, dimensions and data types in OWL and XML, *Online (September 2011) http://www. qudt. org* (2011), 34.

[24] M. Poveda-Villalón, A. Gómez-Pérez and M.C. Suárez-Figueroa, Oops!(ontology pitfall scanner!): An on-line tool for ontology evaluation, *International Journal on Semantic Web and Information Systems (IJSWIS)* **10**(2) (2014), 7–34.

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

Table 3
Ontology metrics

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