CDM-Core: A Manufacturing Domain Ontology in OWL2 for Production and Maintenance

Luca Mazzola*, Patrick Kapahnke, Marko Vujic and Matthias Klusch
I2S - Intelligent Information Systems Research Team, ASR- Agents and Simulated Reality Department,
German Research Center for Artificial Intelligence (DFKI) Stuhlsatzenhausweg 3, D-66123 Saarbruecken, Germany

Keywords: CDM-Core, Applied Ontology, Semantic Annotation, Knowledge Engineering, CREMA H2020 RIA Project, Ontology Quality Measurement.

Abstract: Ontology engineering is known to be a complex, time-consuming, and costly process, in particular, if an ontology has to be developed from scratch, and respective domain knowledge has to be formally encoded. This paper presents the largest publicly available manufacturing ontology CDM-Core in the standard formal ontology language OWL2[1]. The CDM-Core ontology has been developed within the European research project CREMA in close collaboration with the user partners in order to sufficiently cover the CREMA use case domains of metal press maintenance and automative exhaust production. CDM-Core makes use of many relevant standard vocabularies and ontologies, with only about one fifth of its size being CREMA use case specific. The practical applicability of CDM-Core for semantic annotation of domain-related process models, sensor data and services has been approved by the user partners, and its quality according to selected common criteria of verification and validation was successfully evaluated. From the public release of the CDM-Core, we expect to cover the lack of a base common ontology for the manufacturing domain, thanks to feedbacks from industrial reuse and improvements from the community.

1 INTRODUCTION

Developing ontologies is considered nowadays a standard activity in research project dealing with semantics. Unfortunately, this is not a common result of applied projects, where the effort and knowledge required to develop an ontology from scratch is considered not sustainable, in respect of the expected benefits. For this reason, in the context of the EU-founded Horizon2020 CREMA[2] project, we decided to develop the ”CREMA Data Model, Core module” (CDM-Core), as a manufacturing ontology taking into account both the general manufacturing domain applicability and the specific project use cases coverage. As a result, this is the first publicly available applied manufacturing ontology (Mazzola et al., 2016), composed by three different parts: a general manufacturing-related flat layer, a set of domain-specific or standard-based vertical slices such as ”Conditional Monitoring” or ”Semantic Sensor Network”, and some use cases specialized segments (automotive exhaust production and metallic press maintenance), that can be a guidance for developing other specific applications.

The rest of the paper is organized as follows: In section 2 the CDM-Core requirements are presented, together with the obtained results from the distributed process for the ontology creation; some of the CDM-Core usages for semantic annotation of process models, services and data streams are showcased in Section 3; whether Section 4 closes the paper with an analysis of the quality measures for the developed ontology.

2 THE CDM-Core ONTOLOGY

From the use cases description and the user partner inputs, a set of high level requirements was designed, and subsequently validated.

The elicited requirements include multifaceted aspects, such as the CDM-Core capability to represent domain knowledge for both use cases, in order to allow annotation of process model, services and sensors

---

*Corresponding author, mazzola.luca@gmail.com

[1] Available at: http://sourceforge.net/projects/cdm-core/

[2] CREMA is ”Cloud-based Rapid Elastic MAnufacturing” and its website is http://www.crema-project.eu/

Mazzola, L., Kapahnke, P., Vujic, M. and Klusch, M.
CDM-Core: A Manufacturing Domain Ontology in OWL2 for Production and Maintenance.
DOI: 10.5220/0006056301360143
In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 2: KEOD*, pages 136-143
ISBN: 978-989-758-203-5

data; its expected (logical) consistency; the adoption of selected relevant freely available standards; and the usage of a standard W3C modelling language. Based on these specifications, the development started with the engineering phase, as briefly described in the next section.

2.1 Ontology Engineering

The CDM-Core is modeled in the standard W3C ontology language OWL2-DL (Consortium et al., 2012). The CDM-Core has been developed by the task partners according to the distributed ontology engineering (OE) methodology DILIGENT (Tempich et al., 2005). Coherently with the literature (Sure et al., 2009), (Simperl et al., 2010) there are some common steps, that were followed also in this case.

Initially, the ***Domain analysis*** concerns the elicitation of requirements from use cases, the identification of relevant information sources and the ranking and adoption of relevant semantic data models which are available for public reuse. Then there is the ***Conceptualisation of knowledge***, where these inputs are tranformed into of a semi-formal conceptual model of objects and concepts with taxonomic relationships. Following there is the ***Formalisation of the conceptual model*** where the conceptual model is translated into a knowledge representation language with formal semantics. Eventually, the ***Evaluation of the formal ontology*** phase analyzes the sufficient coverage and description of the domains by users and domain experts and the syntactic correctness, consistency and normalisation of the ontology by the ontology engineer. At the end of each phase, an iteration loop can happen, if some of the involved professionals ask for a refinement based on the reached results. In our particular case, we observed multiple iteration, in particular triggered by the user partners.

As result, the requirement that CDM-Core represents knowledge of the CREMA use cases was eventually achieved to a satisfactory degree. In fact, the semantics of the given process models and sensor data can be basically described using the ontology.

The distributed engineering of the shared CDM-Core ontology has been performed by task partners according to the DILIGENT methodology (Pinto et al., 2004).

The role of ontology engineer is manifold and includes (a) the support of domain analysis and conceptualisation of knowledge, (b) the formalisation of the conceptual model in OWL2, and (c) the technological evaluation of the CDM-Core. A set of selected partners plays the role respectively of domain experts and users of the CDM-Core. All task partners are members of the control board for ontology analysis, revision and evaluation.

The CDM-Core ontology engineering process is cyclic. It is based on four main steps with controlled iterations: the first step is ***Build*** where the ontology engineering team builds a very small and basic consensual version of the CDM-Core ontology. These initial activities are carried out by the domain experts intensively supported by the ontology engineer. The second operation is a ***Local refinement*** where each domain expert performs an in-depth refinement of the shared CDM-Core version at the local site, towards a refinement of the conceptual model per use case. These activities are carried out concurrently and at geographically dispersed sites. Every local ontologies is evaluated by domain experts and ontology engineer, and then formalised. As third step, the ***Analysis and revision*** requires that the control board analyses the locally refined ontologies and revises the shared CDM-Core ontology accordingly, by means of identification of similarities and their respective alignment. Eventually, after a new release of the ontology, a ***Local update*** is performed by domain experts before initiating further local refinements (i.e. 2^{nd} operation).

The user partners informed that for the considered processes and sensor data in the use cases currently no standard data models were used at their sites.

The result of the initial search and assessment of relevant non-semantic standard data models carried out by the task partners is shown in Table 1. In particular, the main domains of listed data models according to their public description are shown in the column Domain Coverage; in column Public Reuse the availability of these data models for their translation (e.g. to OWL2) and inclusion into the publicly available semantic data model CDM-Core is shown. Instead, in Table 2 are presented the initial set of considered semantic data models and extensions in the OWL2 language.

2.2 Result

The distributed development based on the presented requirements guided the creation of the ontology. In Table 3 the actually reused public ontologies in CDM-Core are stated, together with their role and their characteristics (numbers of classes, properties and axioms). The main result, beside the ontology itself, is its usage for covering the identified requirements, examples of which are presented in the next section.

Table 1: Selected non-semantic standard data models for CREMA use cases.

Data Model	Modeling Language	Domain Coverage	Public Reuse
ISO 13372:2012	UML Txt/Tab	Condition monitoring and diagnostics of machines, predictive maintenance	(Y)[*]
ISO 10303 (STEP) APs	UML EXPRESS EXPRESS-G	AP214 in AP242:2104 - Core data for automotive mechanical design processes AP239 - Product Life Cycle Support AP224 - Mechanical product definition for process plans AP240 - Process plans for machined products	N
ISO STEP PDM Schema V1.2	Graphical notation, Txt/Tab	Product data management (common subset extracted from STEP APs 214, 203, 212, 232)	(Y)
UN/CEFACT CCL, UNTDED-ISO 7372	Graphical notation, Txt/Tab, XML(S)	Supply chain and cross-border trading transaction messages for buy, ship and pay business processes	Y
ASD S-2000M V6.0	UML Txt/Tab	Material management incl. spare parts, focus on aerospace industry	N
ISA-88	UML Txt/Tab B2MML	Batch control configuration and communication between components in batch manufacturing plants	N
ISA-95	UML Txt/Tab B2MML	Business logistics and manufacturing control incl. production scheduling, maintenance management - at the level of enterprise, site, area	N
ISO 3166 , ISO 4217	Txt/Tab, XML	Country and currency codes	Y

[*] : only the Informative sections

Table 2: Selected semantic data models and extensions in OWL2 language for CREMA use cases.

Ontology	Type	Relevant Domain Coverage	Standard	Public Reuse
SSN (extends DUL)	Domain	Sensory, Sensor Networks	W3C	Y
MASON	Upper	Manufacturing		Y
DUL (DOLCE+DnS Ultralite)	Upper	Concepts of physical and social context, temporal and spatial relations	W3C	Y
GeoNames	Domain	Geolocation	W3C	Y
ONTO-PDM [PDC12]	Upper	Manufacturing product data	IEC 62264, ISO 10303	N
SCORVoc	Domain	Supply chain operations reference (SCOR)	APICS	(Y)[**]
CM (extends SSN)	Domain	Condition Monitoring, Machinery Maintenance	ISO/IEC 13372	Y

[**] : the original N answer is overrided by the written permission received by APICS

Table 3: Publicly available ontologies (or non-semantic standards translated) used inside CDM-Core release.

Ontology	Type	Characteristics		
		Classes	Properties	Axioms
MASON (Lemaignan et al., 2006)	Upper Ontology	224	40	370
SSN (Compton et al., 2012)	Domain	52	55	127
ConditionMonitoring (Günel et al., 2013)	Domain	182	41	363
vCard (Iannella and McKinney, 2013)	Domain	62	83	679
Org (Reynolds, 2014)	Domain	15	37	662
Time (Hobbs and Pan, 2006)	Upper Ontology	13	41	181
TimeLine (Raimond and Abdallah, 2006)	Upper Ontology	26	60	350
DUL (Gangemi, 2012)	Upper Ontology	76	109	517
SCORvoc (Petersen et al., a) (Petersen et al., b)	Domain	279	297	7657

3 USAGES

This section presents examples of semantic annotations of process models, services, and sensor data of the CREMA use cases based on the CDM-Core. This is complemented with examples of how these semantic annotations can be used by other CREMA components in the use cases, for example for the automatic implementation and optimization of process model with services (SOA paradigm).

3.1 Process Model Annotation

In CREMA, the process models are described in the standard BPMN (Business Process Model Notation). The formal semantics of a process model in BPMN can be described by means of its annotation with elements of formal ontologies. Such annotation allows applications to semantically reason on them in general, and assist process managers in their semantic service-based implementation in particular. Services can be more precisely selected and automatically composed for implementing a given process model based on its semantic annotation. This is in line with the semantic SOA paradigm[3] and reference model (MacKenzie et al., 2006).

In CREMA, a component will perform an optimal service-based implementation of given process models with advanced means of semantic service selection and planning. That requires the semantic annotation of both process models and available services. The semantic annotation of process models based on the CDM-Core is manually done by the process manager with the help of some specialized interfaces.

There is no standard format for the semantic annotation of process models in BPMN yet (Boissel-Dallier et al., 2015). On an abstract level, the semantics of processes and services can be basically described in terms of their input (I), output (O), precondition (P) and effect (E) of their execution. In particular, the task partners first informally described the basic IOPE semantics of all process tasks of given process models in the CREMA use cases. These informal semantic description consists of text, including relevant main concepts which were either identified in or newly added to the CDM-Core for this purpose. Next, these semi-formal descriptions were manually transformed into the formal semantic IOPE-based annotations with CDM-Core.

Figure 1 shows an example of a semantic IOPE-based annotation of one process task of one process model. In particular, this process task is concerned with the resource allocation of a suitable robot. Given a production schedule containing a list of orders to be fulfilled, this task identifies a robot cell and corresponding robot, and leases it for the schedule. A robot must be able to perform welding operations and has to be equipped with particular clamps to hold a certain type of exhaust as specified in the tasks of the schedule. The requirements described above are specified through different parameters, including the production schedule input, the robot cell and robot outputs, and a range of internal variables. These semantic annotations (such as the ones in Fig. 1) were made relying on domain-specific and upper ontologies together with the use case specific part of CDM-Core.

The following Listing provides an example of the semantic annotation of the process task in part A of the Figure 1 as extension in the BPMN XML source code. Additional attributes named *inputs*, *outputs*, *preconditions* and *effects* are used to attach the semantic annotations to the standard BPMN definitions.

```
<?xml version="1.0" encoding="UTF-8"?>
<bpmn:definitions ...
  xmlns:tco=http://<URI1>/wp8/tco.owl#
  xmlns:mas="http://<URI2>/mason.owl#">
  ...
  <bpmn:task id="Task_1w5d3zt"
    name="Select Robot Cell"
   inputs="tco:ProductionSchedule :PS"
   preconditions="tco:includes(:PS,:T)
       and tco:ProductionTask(:T)
       and tco:includes(:T,:OP)
       and mas:requiresTool(:OP,:TOOL)
       and tco:Welder(:TOOL)
       and tco:produces(:OP,:EX)
       and tco:Exhaust(:EX)"
   outputs="tco:RobotCell :CELL
       (tco:Robot
       and tco:supports some tco:ExhaustClamp
       and tco:supports some tco:Welder) :R"
   effects="tco:isAllocatedFor(:CELL,:PS)
       and tco:includes(:CELL,:R)
       and tco:supports(:R,:TOOL)">
    ...
  </bpmn:task>
    <bpmn:incoming>SeqFlow_0egvn5w</bpmn:incoming>
    <bpmn:outgoing>SeqFlow_03haiqx</bpmn:outgoing>
</bpmn:definitions>
```

3.2 Service Annotation

Semantic services are services whose functional and non-functional semantics are described with formal ontology-based annotations. On an abstract level, a semantic service description includes a semantic IOPE-based profile and a semantic process model that

---

[3] Semantic SOA is considered key to the development of semantics-empowered intelligent applications for the future Internet in various domains including manufacturing 4.0, and supported by an increasing number of industrial stakeholders such as Software AG, SAP, IBM, Siemens.

describe what this service does and how it actually works (Klusch, 2008a), (Klusch, 2008b).

According to Semantic SOA, process models can be implemented with executable services which semantics are described with a shared formal ontology. The implementation of each step or task of a process model with a relevant single or composite service by a process designer can be supported by means of automated high-precision semantic selection and planning of annotated services, either in fully automatically or in semi-automatically (with user interaction).

Since the CDM-Core is defined in OWL2, one natural choice of the semantic service description format would be OWL-S (Web Ontology Language for Web Services (Martin et al., 2004)) which allows a grounding of semantic services in WSDL or REST services (Lathem et al., 2007). As mentioned above, semantic service descriptions will be used to determine a functionally optimal assignment of services to given annotated process models.

3.3 Data Stream Annotation

In CREMA, CDM-Core can also be used to semantically annotate sensor data of the use cases. The user partners provided general information about the metal press system, the robots, their components and the attached sensors, as well as the relevant sensor data schema. The given data schema for the stream of time-stamped and sequentially ordered data buckets can be in different format, such as CSV, TSV (Tab Separated Values) or JSON (Gray et al., 2011).

In particular, each sensor observes or measures one property such as pressure or temperature. The semantic annotation of streamed sensor data will be automatically done usign the CDM-Core concepts. That requires the mapping of the sensor measurement label to the concept in the ontology, which defines the formal semantics of this label in XML-RDF encoded OWL2. As a result, the data item is described by a set of RDF triples as an instance of the corresponding concept in the ontology. This mapping table and respective naming of sensor classes and measurements is given by the partner generating the stream itself.

Figure 2 presents an example of semantic annotation of multi-variate sensor data from multiple sensors attached to the hydraulic drive system of a metal press. The semantic annotation of sensor data with the CDM-Core allows for a qualitative, that is domain knowledge-based, rather than quantitative statistics-based data interpretation.

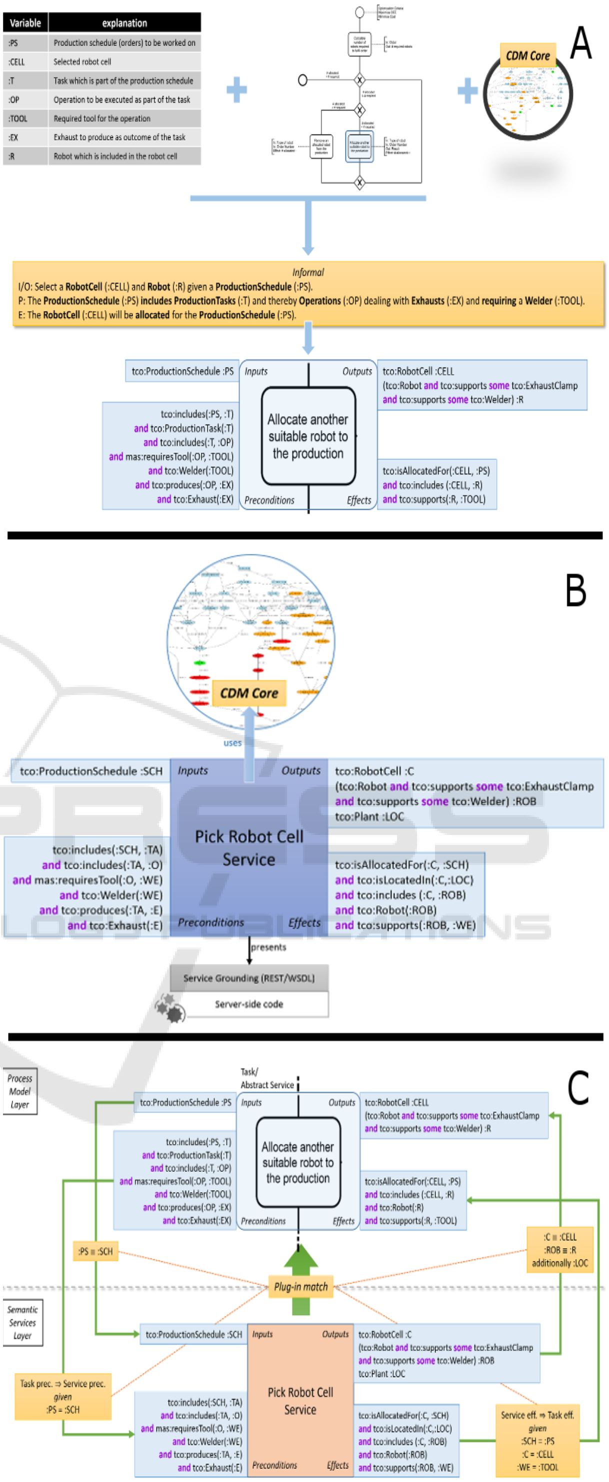
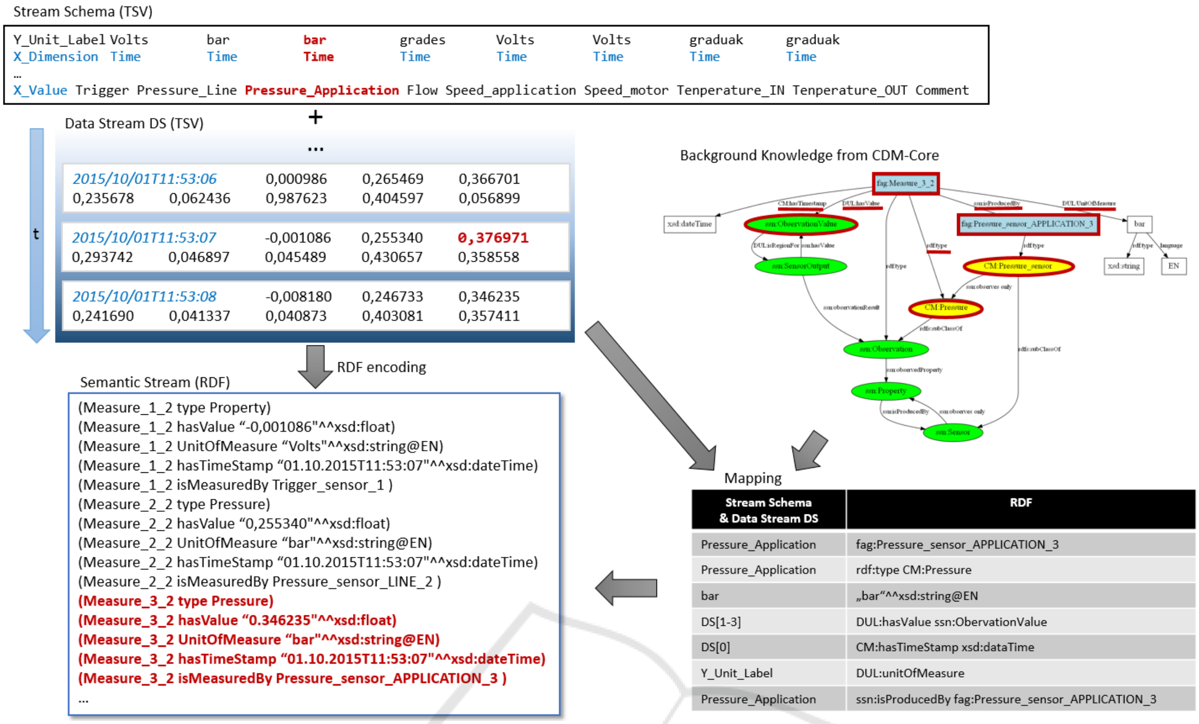


Figure 1: Usage of the CDM-Core ontology for (A) Process Task inside the BPMN semantic annotation, (B) Service semantic Annotation, and (C) a matching using the plug-in approach (Paolucci et al., 2002) between the semantic annotation on top and middle of this example.



Stream Schema (TSV)

```
Y_Unit_Label Volts      bar        bar        grades     Volts      Volts      graduak    graduak
X_Dimension  Time       Time       Time       Time       Time       Time       Time       Time
…
X_Value Trigger Pressure_Line Pressure_Application Flow Speed_application Speed_motor Tenperature_IN Tenperature_OUT Comment
```

Data Stream DS (TSV)

\+

…

```
2015/10/01T11:53:06     0,000986   0,265469   0,366701
0,235678   0,062436     0,987623   0,404597   0,056899

2015/10/01T11:53:07     -0,001086  0,255340   0,376971
0,293742   0,046897     0,045489   0,430657   0,358558

2015/10/01T11:53:08     -0,008180  0,246733   0,346235
0,241690   0,041337     0,040873   0,403081   0,357411
```

RDF encoding

Semantic Stream (RDF)

```
(Measure_1_2 type Property)
(Measure_1_2 hasValue "-0,001086"^^xsd:float)
(Measure_1_2 UnitOfMeasure "Volts"^^xsd:string@EN)
(Measure_1_2 hasTimeStamp "01.10.2015T11:53:07"^^xsd:dateTime)
(Measure_1_2 isMeasuredBy Trigger_sensor_1 )
(Measure_2_2 type Pressure)
(Measure_2_2 hasValue "0,255340"^^xsd:float)
(Measure_2_2 UnitOfMeasure "bar"^^xsd:string@EN)
(Measure_2_2 hasTimeStamp "01.10.2015T11:53:07"^^xsd:dateTime)
(Measure_2_2 isMeasuredBy Pressure_sensor_LINE_2 )
(Measure_3_2 type Pressure)
(Measure_3_2 hasValue "0.346235"^^xsd:float)
(Measure_3_2 UnitOfMeasure "bar"^^xsd:string@EN)
(Measure_3_2 hasTimeStamp "01.10.2015T11:53:07"^^xsd:dateTime)
(Measure_3_2 isMeasuredBy Pressure_sensor_APPLICATION_3 )
…
```

Background Knowledge from CDM-Core

Mapping

Stream Schema & Data Stream DS	RDF
Pressure_Application	fag:Pressure_sensor_APPLICATION_3
Pressure_Application	rdf:type CM:Pressure
bar	„bar"^^xsd:string@EN
DS[1-3]	DUL:hasValue ssn:ObservationValue
DS[0]	CM:hasTimeStamp xsd:dataTime
Y_Unit_Label	DUL:unitOfMeasure
Pressure_Application	ssn:isProducedBy fag:Pressure_sensor_APPLICATION_3

Figure 2: Example of annotation for a data stream schema, in the context of a CREMA project use case.

4 QUALITY MEASURES

Ontology evaluation is the task of measuring the quality of an ontology according to given criteria. It is "*a technical judgment of the content of the ontology with respect to a frame of reference during every phase and between phases of their lifecycle*" and can be classified into ontology verification and validation (Gómez-Pérez et al., 2003). There are several approaches, methods and tools for ontology evaluation available, but no best practices and guideline for the selection of measures for ontology quality criteria.

The result has been evaluated according to a selected subset of ontology quality criteria and measures defined in (Vrandecic, 2010). In particular, the selected criteria subsume the given requirements for the CDM-Core. Since there is no publicly available ontology for the manufacturing domain, our CDM-Core could not be evaluated against some gold standard as a reference. For this reason it was performed by all task partners with user partners as stakeholders.

In summary, all requirements for the CDM-Core are satisfied. The individual results are described in the following in the context of the selected criteria.

Verification was concerned with evaluating if CDM-Core specification is formally correct and meaningful in terms of syntactic validity, and logical consistency. *Syntactic validity* refers to the syntactically correct encoding of the ontology specification, which can be tested with appropriate validation tools such as the OWL validator[4] , SWOOP[5] , CityPulse Ontology Validator[6] , Eyeball[7] , OBO-Edit[8] , and OOPS![9].

Logical Consistency requires that the ontology specification does not include or allow for any logical contradiction. In other words, an ontology is inconsistent [10] if it does not allow any formal model to satisfy its axioms. Checking of consistency can be done with a classical ontological reasoner such as Pellet (Sirin et al., 2007). The above mentioned validation tools support consistency checking, but they differ in the extent they check for common problems or pitfalls.

Results: The CDM-Core is syntactically valid and consistent. The XML-RDF encoded specification of the CDM-Core ontology in OWL2 was successfully tested with the tool OOPS! (also, advanced evaluation option): it does not contain critical problems such as circular class hierarchies, redundant axioms, incon-

---

[4] http://mowl-power.cs.man.ac.uk:8080/validator/

[5] http://semanticweb.org/wiki/Swoop

[6] http://iot.ee.surrey.ac.uk/SSNValidation/

[7] https://www.w3.org/2001/sw/wiki/Eyeball

[8] http://oboedit.org/docs/index.html

[9] http://oops.linkeddata.es/

[10] It is is a kind of inconsistency (Flouris et al., 2006).

sistent naming schemes, or other logically inconsistent definitions of concepts and relations.

Validation was concerned with evaluating if the ontology is practically useful for the targeted stakeholders in terms of its accuracy and completeness of both covering the use case domains and supporting the tasks for which the ontology has been designed, its computational efficiency, and its adaptability to manufacturing domains and tasks of other stakeholders (additional criterion that the task partners identified later).

Accuracy checks if the specification complies with the knowledge of the stakeholders. Correctness in this case means compliance to the gold standards" of use case descriptions and respective conceptual models.

Completeness verify that the use case domains are not only accurately but fully covered by the ontology, and that semantic annotations of given use case process models, services, and sensor data are sufficiently supported. It also covers the structural quality of the ontology for which measurement the following measures were selected:

(TD) maximal class Taxonomy Depth (Lozano-Tello and Gómez-Pérez, 2004), i.e. the maximum subsumption path length of the CDM-Core. It covers the intuition that a high TD reflects a more detailed concept knowledge represented by the ontology.

(RR) Relationship Richness (Vrandecic, 2010), i.e. the ratio between the number of property names and the number of class names and property names of the CDM-Core. It reflects the diversity of relations in the ontology and cover the intuition that detailing of existing classes would increase the relational richness of the ontology.

(AR) Attribute Richness (Tartir et al., 2005), i.e. the average number of properties (attributes) per class. It suggests the intuition that high attribute richness indicates more information about each class on average.

Computational efficiency is the reasoning complexity (RC) of CDM-Core, i.e. the complexity that applies to the common reasoning tasks for the OWL2 fragment that is actually being used in the CDM-Core.

Adaptability represents an indicator for the effort expected for effectively reusing the developed ontology in different cases inside the same domain.

Results: CDM-Core is *accurate*: as a consequence of the approval of its sufficient matching with the underlying conceptual model by the stakeholder, definitions and descriptions in CDM-Core are correct.

It is also *complete* according to the stakeholders. As a result of its joint engineering, the CDM-Core was eventually approved by the stakeholders to represent all relevant instances, concepts and relations of the conceptual model. Moreover, it allowed the annotation of each of the given process models with concepts and properties; all sensor measurements of the given data stream schema were semantically mapped to corresponding elements in it; and every used sensors, robot cells and metal press is represented by an appropriately designed individual in CDM-Core.

The structural quality factors of the CDM-Core are: *TD = 7*, *RR =* ***0.3993***, *AR =* ***0.8156***. Our interpretation of these values is that the developed CDM-Core features a very high number of domain-specific classes with very high attribute richness (AR), which indicates a high amount of detailed information about each class on average. Its class hierarchy is of the same moderate maximal depth (TD) as, for example, the generic manufacturing ontology MASON and the standard W3C SSN ontology, it partly builds upon.

The worst case reasoning complexity (RC) is computed as the intersection of the complexity of the different OWL2 fragments in the CDM-Core, and is equivalent to SROIQ(D). Anyway no definition of CDM-Core covers jointly all the operators of this complexity, which indicates that the reasoning complexity in practice might be of some magnitude lower.

Adaptability is limited due to its focus on covering the use case domains described (CREMA_consortium, 2016a), (CREMA_consortium, 2016b) and allowing the tasks of annotating the given process models, services and sensor data. However, the CDM-Core in particular builds on and includes generic and standard-based ontologies. These generic parts can serve other stakeholders to model knowledge of different manufacturing domains and tasks. The normalized proportion of the generic to the CREMA use case domain-specific parts is 21.09%.

The CDM-Core is specified in OWL2-DL and can be in principle extended and specialized monotonically, i.e. without the need to remove axioms.

5 CONCLUSIONS

In this paper we presented the first publicly available ontology for the manufacturing domain, together with the process for its development. It is one intermediate result of a shared effort of different organizations in the context of a collaborative project. The validation showed its capability for covering the requirements elicited as prerequisite for the ontology engineering phase. Additionally, some measures for the structural quality of the produced CDM-Core ontology are presented, together with its applications for process model, services, and data stream annotations. The resulting ontology is released for public reuse and we expect that industries can reuse it, provide feedbacks and ask for improvement to the community.

This work was partially supported by the Commission of the European Union within the CREMA H2020-RIA project (Grant agreement no. 637066).

REFERENCES

Boissel-Dallier, N., Benaben, F., Lorré, J.-P., and Pingaud, H. (2015). Mediation information system engineering based on hybrid service composition mechanism. *Journal of Systems and Software*, 108:39–59.

Compton, M., Barnaghi, P., Bermudez, L., GarcíA-Castro, R., Corcho, O., Cox, S., Graybeal, J., Hauswirth, M., Henson, C., Herzog, A., et al. (2012). The ssn ontology of the w3c semantic sensor network incubator group. *Web Semantics: Science, Services and Agents on the World Wide Web*, 17:25–32.

Consortium, W. W. W. et al. (2012). Owl 2 web ontology language document overview.

CREMA_consortium (2016a). D7.2 - use case 1 machinery maintenance definition. Public deliverable, The CREMA Project (H2020-637066).

CREMA_consortium (2016b). D8.2 - use case 2 (automotive) definition. Public deliverable, The CREMA Project (H2020-637066).

Flouris, G., Huang, Z., Pan, J. Z., Plexousakis, D., and Wache, H. (2006). Inconsistencies, negations and changes in ontologies. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, page 1295. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Gangemi, A. (2012). Dolce+ dns ultralite (dul) ontology, july 2012.

Gómez-Pérez, A., Fernández-López, M., and Corcho, O. (2003). Ontological engineering. advanced information and knowlege processing.

Gray, A. J., García-Castro, R., Kyzirakos, K., Karpathiotakis, M., Calbimonte, J.-P., Page, K., Sadler, J., Frazer, A., Galpin, I., Fernandes, A. A., et al. (2011). A semantically enabled service architecture for mashups over streaming and stored data. In *Extended Semantic Web Conference*, pages 300–314. Springer.

Günel, A., Meshram, A., Bley, T., Schuetze, A., and Klusch, M. (2013). Statistical and semantic multisensor data evaluation for fluid condition monitoring in wind turbines. In *Proc. 16th Intl. Conf. on Sensors and Measurement Technology, Germany*.

Hobbs, J. R. and Pan, F. (2006). Time ontology in owl. *W3C working draft*, 27:133.

Iannella, R. and McKinney, J. (2013). vcard ontology-for describing people and organizations. *Web Page. URL: http://www. w3. org/TR/vcard-rdf/accessed*, pages 19–10.

Klusch, M. (2008a). Semantic web service coordination. In *CASCOM: Intelligent Service Coordination in the Semantic Web*, pages 59–104. Springer.

Klusch, M. (2008b). Semantic web service description. In *CASCOM: intelligent service coordination in the semantic web*, pages 31–57. Springer.

Lathem, J., Gomadam, K., and Sheth, A. P. (2007). Sa-rest and (s) mashups: Adding semantics to restful services.

Lemaignan, S., Siadat, A., Dantan, J.-Y., and Semenenko, A. (2006). Mason: A proposal for an ontology of manufacturing domain. In *Distributed Intelligent Systems: Collective Intelligence and Its Applications, 2006. DIS 2006. IEEE Workshop on*, pages 195–200. IEEE.

Lozano-Tello, A. and Gómez-Pérez, A. (2004). Ontometric: A method to choose the appropriate ontology. *Journal of database management*, 2(15):1–18.

MacKenzie, C. M., Laskey, K., McCabe, F., Brown, P. F., Metz, R., and Hamilton, B. A. (2006). Reference model for service oriented architecture 1.0. *OASIS standard*, 12.

Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIlraith, S., Narayanan, S., Paolucci, M., Parsia, B., Payne, T., et al. (2004). Owl-s: Semantic markup for web services. *W3C member submission*, 22:2007–04.

Mazzola, L., Kapahnke, P., Vujic, M., and Klusch, M. (2016). Cdm-core. ontology public source code. http://sourceforge.net/projects/cdm-core/.

Paolucci, M., Kawamura, T., Payne, T. R., and Sycara, K. (2002). Semantic matching of web services capabilities. In *International Semantic Web Conference*, pages 333–347. Springer.

Petersen, N., Grangel-González, I., Auer, S., Coskun, G., Frommhold, M., and Tramp, S. Scorvoc: a vocabulary based on the supply chain operation reference model.

Petersen, N., Grangel-González, I., Coskun, G., Auer, S., Frommhold, M., Tramp, S., Lefrançois, M., and Zimmermann, A. Scorvoc: Vocabulary-based information integration and exchange in supply networks.

Pinto, H. S., Staab, S., and Tempich, C. (2004). Diligent: Towards a fine-grained methodology for distributed, loosely-controlled and evolving. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 2004)*, volume 110, page 393.

Raimond, Y. and Abdallah, S. (2006). The timeline ontology. *OWL-DL ontology*.

Reynolds, D. (2014). An organization ontology. *URL http://www. w3. org/TR/vocab-org*.

Simperl, E. P. B., Mochol, M., and Bürger, T. (2010). Achieving maturity: the state of practice in ontology engineering in 2009. *IJCSA*, 7(1):45–65.

Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., and Katz, Y. (2007). Pellet: A practical owl-dl reasoner. *Web Semantics: science, services and agents on the World Wide Web*, 5(2):51–53.

Sure, Y., Staab, S., and Studer, R. (2009). Ontology engineering methodology. In *Handbook on ontologies*, pages 135–152. Springer.

Tartir, S., Arpinar, I. B., Moore, M., Sheth, A. P., and Aleman-Meza, B. (2005). Ontoqa: Metric-based ontology quality analysis.

Tempich, C., Pinto, H. S., Sure, Y., and Staab, S. (2005). An argumentation ontology for distributed, loosely-controlled and evolving engineering processes of ontologies (diligent). In *European Semantic Web Conference*, pages 241–256. Springer.

Vrandecic, D. (2010). Ontology evaluation.