A review on Utilizing Ontological Approaches in Integrating Assembly Design and Assembly Process Planning (APP)

Baha Hasan#1, Jan Wikander#2

KTH Royal Institute of Technology, Department of Machine Design
Stockholm, Sweden

Abstract *This review paper is focusing on utilizing ontological approaches in capturing and sharing assembly design knowledge towards better integration between the product design and manufacturing, more specifically between assembly design and assembly process planning (APP) domains. Most of the significant Approaches that have been proposed by researchers in the last decade are reported in this extended review paper. Comparison between various approaches and conclusion about the overall related literature are conducted as well.*

Keywords — *APP, Assembly design, Product design, Ontology, Manufacturing.*

I. Introduction

Ontologies and semantic web technologies have been widely applied to achieve semantic integration and to enhance semantic interoperability, which consequently can support knowledge sharing between product design and manufacturing. An ontology attempts to define concepts, mutual relations between concepts and constraints governing those relations.

In published literature many definitions have been addressed for an ontology [1, 2]. One of the most comprehensive definitions is the one reported by Uschold and Gruninger [3]: "*An ontology is a formal description of the entities within a given domain, the properties they possess, the relationships they participate in, the constraints they are subject to, and the patterns of behaviour they exhibit*". In this definition, formality is mentioned as one of the fundamental requirements for the ontology. Formality is highly related to interpretation by computers, the more formal an ontology is the more interpretable by computers it becomes [4]. Ontologies are also classified, according to the level of rigor and restrictions applied on the terminologies conceptualized in the ontology, into heavyweight and lightweight ontologies [4]. The heavyweight ontologies use axioms and constraints to restrict the meaning of the terms and facilitate deduction of new knowledge by using inference rules. Lightweight ontologies use textual definitions of concepts and terms, which may lead to ambiguities when defining the semantics of concepts [5].

Other classifications are based on the level of generality [6] and on the conceptualization structure [7]. Both classifications distinguish between upper ontologies (generic, top-level ontologies), domain ontologies and application ontologies. Upper ontologies consist of generic, abstract, and high level concepts which can be applied to a wide range of domains. Only a very general level of knowledge modelling can be done by using upper ontologies. An example of upper ontologies is the foundation ontologies. Foundation ontologies provide a knowledge base for more specialized ontologies [8] such as domain and application ontologies. Domain ontologies consist of domain-specific concepts whereas application ontologies capture semantics for a specific application belonging to a specific domain. The level of generality decreases from upper ontologies to domain and application ontologies.

Recently, ontologies and semantic web technologies have been widely applied in the manufacturing domain. The aims behind utilizing an ontological modelling approach in manufacturing systems could by summarized by three points:

- To support adaptability and re-configurability of the manufacturing systems. Several manufacturing paradigms based on ontologies have been proposed to support re-configurability of manufacturing systems [9, 10]. This is done by synchronizing the changes in the physical manufacturing system level with the corresponding modifications in the control software level. The ontological model approach provides a proper semantic model of the production system and makes it accessible to the control software through the use of semantically-enriched Web Services (i.e. Semantic Web Services). This approach can enhance re-configurability of production systems because the semantics makes the knowledge about the manufacturing system itself understandable to the control software. More details about the use of ontologies in supporting adaptability and re-configurability of manufacturing systems are available in [11, 12].

- To model manufacturing systems from different perspectives such as product, process and assembly process planning. Examples from literature are provided in [13, 14].
- To support semantic interoperability between manufacturing and other related domains in product-life cycle. Semantic interoperability can be achieved when the meaning associated to captured information and knowledge can be effectively shared across different domains and applications without any loss of the meaning and intent of the information and knowledge during the exchange process [15]. In order to achieve semantic integration, reference ontologies have been developed for information interoperability [16, 17]. Examples from literature are provided in [18, 19].
- To support knowledge sharing between the product design and different manufacturing domains [20, 16, 17, 21]. Attached to this point is the utilization of ontology capabilities in reusing knowledge and inferring new knowledge based on the existing knowledge, which will support decision making in manufacturing systems.

This research mainly concerns utilization of ontologies in integrating product design and manufacturing. Design - manufacturing knowledge integration includes two stages: product and production system knowledge capturing, which includes modelling and recognition of the knowledge, and product knowledge sharing over the manufacturing environment.

In published literature, there is a difference between the terms knowledge, information and data. According to [22] information is "*the relationships between recognized data (i.e., numbers and symbols) in some context*", while knowledge is *"information with added detail relating how it should be used or applied"* such as *"rules describing what actions to take when certain information exists"* [22]. Another important aspect regarding knowledge is addressed by [23] stating that *"knowledge is composed of concepts".* Based on the former two definitions, knowledge is here considered as a set of concepts with additional related rules that describe the actions to be taken based on the available information. An example that illustrates the difference between knowledge and information in the context of design-manufacturing integration is that tolerance information for cylindrical surfaces could be interpreted as manufacturing knowledge if additional details are added, based on the available tolerance information. Rules for deriving this knowledge would e.g. specify certain types of fitting processes and corresponding fitting resources.

Product design knowledge has a very important role in influencing all other stages of the product-life cycle, and especially so for manufacturing. And the other way around, manufacturing knowledge supports the product design related decisions [24]. According to [25] manufacturing knowledge is *"a sum of facts and data leading the manufacturing community's activities of implementing a production".* Manufacturing knowledge includes a wide variety of knowledge types that are required to facilitate production, such as process planning, assembly knowledge and many others. More about the types of manufacturing knowledge could be found in [24, 25].

II. DESIGN - MANUFACTURING INTEGRATION

In the published literature, several design-manufacturing integration frameworks have been proposed; Fig 1 gives an illustration of the main proposed integration approaches. From a technical point of view, the proposed design-manufacturing integration frameworks are categorized into three main approaches: ontology-based approach, Interface (internal) approach and file-based (external) approach. Several examples are listed under each approach. The main focus on this paper is on the ontology-based approach, the other two categories will be discussed only briefly in this section.

Internal and external product- manufacturing integration approaches are the most popular in the published literature [26]. In the internal approach, API (application programmable interface) functions are used to recognize and extract product design data from a CAD model. While in the external approach product data is transferred using a standard neutral data format, such as IGES, XML or STEP file. Both methods have some limitations in sharing product / assembly knowledge. In both methods, is transferred from user to user or from application to application but not from domain to domain. Another limitation is the data lost during conversion from one format to another format in the external approach, and the syntactic (i.e not semantic) transfer of data in the internal method. More details about the interfacing and the file-based approaches are available in author's previous publications [27, 28].

Under the interfacing and the file-based approaches (Fig. 1) several literature examples are listed for integrating product design and some manufacturing planning applications, such as CAD-CAX integration [29], CAD and Computer Aided Process Planning (CAPP) integration [30, 31], CAD/CAM and CAPP integration [32], CAD/CAM/CAE integration [33], and CAD-manufacturing process planning integration [34, 35, 36]. Many other examples can also be found in the published literature.

Under ontology-based integration approach, several examples of integration frameworks are listed. Integration frameworks in this approach can be categorized into application integration approaches and domain integration approaches. The first listed literature example under ontology-based approach, [37], represents an ontology-based application integration framework. They developed a feature ontology to exchange product design knowledge between CAD and CAPP applications. The proposed feature ontology facilitates the transition of CAD and CAPP files into a neutral format that could be understood by both applications. Feature-based technologies have been used widely in published literature to enhance knowledge sharing between different manufacturing applications such as CAD, CAM and CAPP applications. Similar ontological feature-based integration applications exist, and in the one addressed by [38], a common design features ontology is proposed to facilitate CAD knowledge sharing between SolidWorks (SW) and CATIA.

There are many frameworks proposed for utilizing ontology in manufacturing related domains such as product design, manufacturing, maintenance, assembly design, process planning and APP. In this section we only review some very representative literature examples on applying ontologies in manufacturing, product design and APP sides.

On the manufacturing side, the Manufacturing's Semantic Ontology (MASON) is proposed by [13] to support capturing and sharing of manufacturing knowledge. Manufacturing knowledge is captured though three main concepts: entities, operations and resources. Entities represent geometrical and non-geometrical knowledge (e.g. material and cost), operations represent manufacturing processes (e.g. machining and logistic) and resources represent manufacturing and other resources (e.g. human resources). A machining ontology is proposed by [39] to represent the manufacturing domain knowledge. This machining ontology includes geometrical knowledge (form features), machining processes and machining resources.

On the product design side, a product design and manufacturing process ontology for manufacturing is proposed by [14]. This ontology is focused on reusing knowledge resulting from Design Failure Mode and Effects Analysis (DFMEA) and Process Failure Mode and Effects Analysis (PFMEA) processes in the manufacturing industry. Product ontology to integrate production planning systems with product design applications is proposed by [40]. In this ontology, concepts are adopted based on product data management [41] and enterprise integration [42] standards.

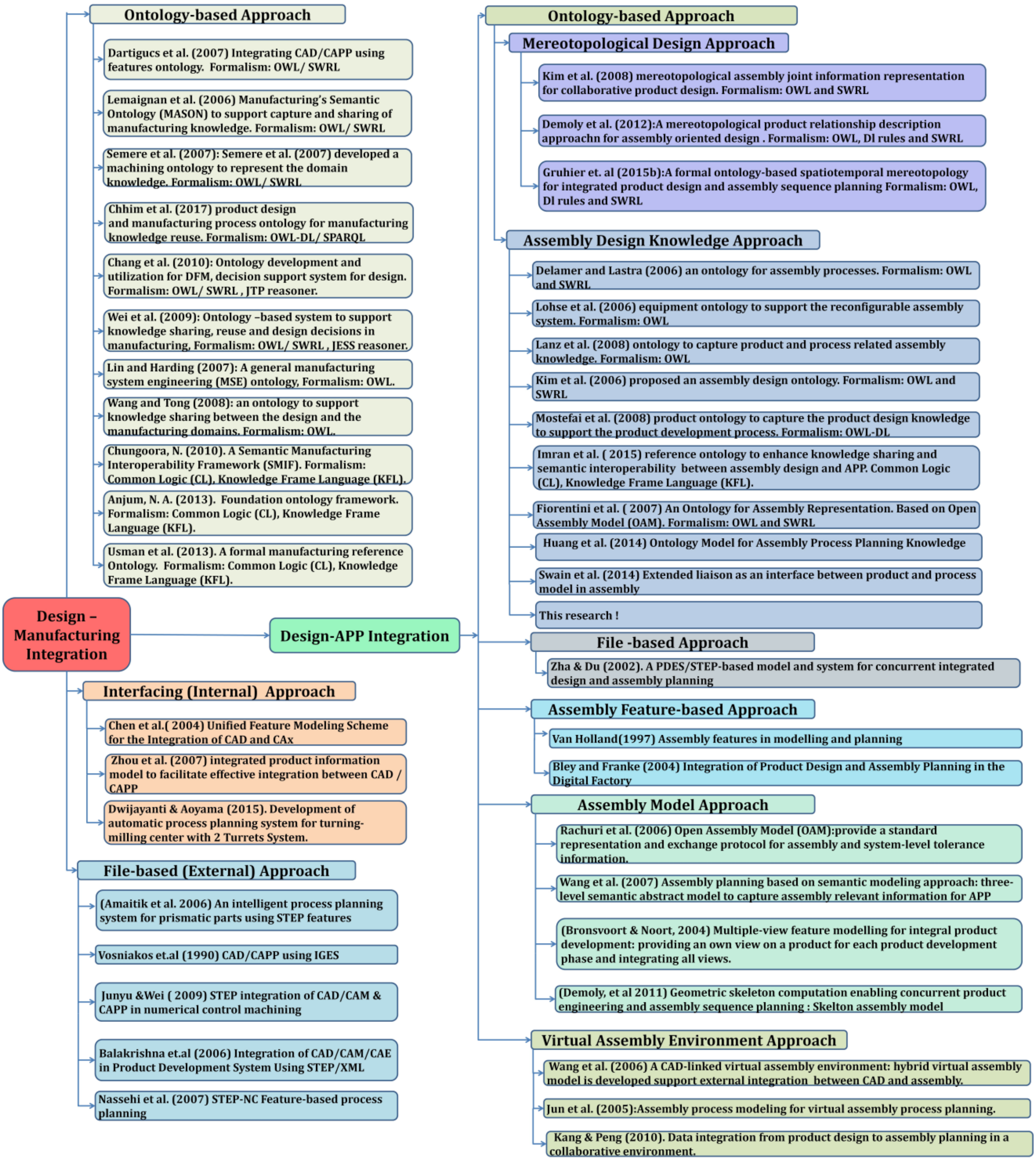

Fig 1: Design-manufacturing sharing tree based on background literature

The ontology includes in this case concepts such as Product, Part, Component and Assembly-component-relationship to represent assembly areas within production planning and product design domains. Another product design ontology is proposed by [43] to integrate product design and Design For Manufacture (DFM). Their work also includes ontology a development methodology for DFM. Product development ontology is proposed by [44] for reuse, integration and sharing of design knowledge to support decision-making during the product development process. The Manufacturing System Engineering (MSE) ontology is proposed by [19] for exchanging knowledge across multi-disciplinary design areas. An ontology to support knowledge sharing between design and manufacturing domains is also proposed by [45]. Their ontology supports capturing manufacturing knowledge to support decision making and knowledge sharing.

A semantic manufacturing interoperability framework (SMIF) is proposed by [20] mainly to facilitate interoperability across product design and manufacturing domains. The SMIF consists of a multilayer ontological framework, in which a foundational ontology for modelling the domain is provided by the first layer. Domain ontologies to model product design and manufacturing domains are provided by the domain ontology layer. The next two layers, Semantic Reconciliation and Semantic Interoperability, are dedicated to support interoperability and knowledge sharing across domain ontologies.

Another multilayer framework, the Interoperable Manufacturing Knowledge System (IMKS) is proposed by [16], to provide seamless computer-based knowledge sharing between departments of a manufacturing enterprise, more specifically product design and manufacturing. The IMKS supports interoperability, ability to overcome semantic and syntactic differences during computer-based knowledge sharing, through the use of foundation and domain ontologies. The IMKS is composed of four layers, a foundation ontology is provided by the first layer, which was later developed by [17], to provide a common ground to the process specification language (PSL) ontology and a core ontology of product design-manufacturing concepts in the second layer. The third layer represents the domain ontologies (design and manufacture domains). The bottom layer in the proposed framework is knowledge base layer. The knowledge base is built by using concepts from the domain ontologies in the upper layer. In the proposed approach the knowledge base is populated by facts through a verification mediator. The verification mediator establishes integration between product design and manufacturing domains through the use of queries for the verified facts. The verified facts are sent to the manufacturing knowledge base to be checked such that no objections are raised by the manufacturing domain ontology; the verification mediator sends the verified facts back to the design knowledge base.

This research is specifically focused on product design-APP integration approaches, which as shown in Fig. 1, can be categorized into five approaches: Ontology-based approach, file-based approach, assembly feature-based approach, assembly-model approach and virtual assembly environment approach. Our main focus in this survey is on the ontology-based approach, the other approaches will be discussed briefly first. The file-based approach represents an extension of the file-based approach in product –manufacturing integration. The assembly knowledge is exported into an external file (e.g. STEP file in the literature example). [46] developed an integrated model and system for concurrent assembly design and planning using STEP file. By applying STEP as an information model. The final aim of their proposed agent-based integration strategy is to support CAD/CAM applications in assembly.

Regardless of the different approaches, product features have been exploited as a basis for most of the integration frameworks between product design and manufacturing. A useful understanding of a feature is given by the definition addressed by [47], *"a feature is a partial form or a product characteristic that is considered as a unit and that has a semantic meaning in design, process planning, manufacture, cost estimation or other engineering discipline"*. The importance of features as a bridge between product design and manufacturing has been addressed extensively in published literature [48], [49]. Assembly feature-based design [50] has been considered as an extension of feature-based design for linking assembly design and APP. Van Holland [51] utilized assembly features in process planning by defining assembly feature as *''features with significance for assembly processes''* . The same author introduced more specific assembly features, from a process perspective: connection features *"such as final position, insertion path/point, tolerances"* and handling features *"characteristics that give the locations on an assembly component that can be safely handled by a gripper during assembly!"*. Another example of using assembly features in assembly process planning is addressed by [52], where assembly features are used to integrate a product design CAD application (CATIA) to assembly planning in digital factory simulation.

The assembly feature integration approach has been extended further by deriving object oriented assembly feature representation [53] and assembly feature semantics [54]. A new integration approach based on assembly feature semantics is known as assembly semantic model approach. This approach is mainly used for modelling assembly design knowledge and to provide enough information about the relations between connected components/parts and precedence constraints between the connections. Since the efficiency of APP is highly dependent on the way in which assembly design is modeled, this approach is considered as a very important stage in integrating assembly design and APP. Open Assembly Model (OAM) [55] is one of the remarkable examples about deriving an assembly design model based on assembly semantics. OAM provides a standard representation and exchange protocol for assembly, which includes tolerance and kinematics representations. A three-layer assembly semantic model is proposed by [56] and is used mainly for APP. The using of semantics in assembly design representations provides better understanding of assembly designer's and planner's intents. These representations allow reasoning about assembly knowledge for APP in a more formal and efficient way.

The multiple –view feature modelling approach [57] added more flexibility to product design modelling by allowing a designer to focus on the information that is relevant for a particular product development phase such as conceptual design, assembly design, part detail design and part manufacturing planning. This is achieved by providing a specific view of a product for each phase and facilitating integration of all views. Each view contains a specific feature model for the corresponding phase. This modelling approach is characterized by establishing different but consistent views to represent the same product.

A new assembly modelling approach is invented by defining new geometric elements in an assembly design called ''skeleton'' [58] to provide support for the geometric product modelling phase. Skeleton models are specialized models of components of an assembly that define basic geometries (point, curve, surface, plane, axis and coordinate system), constraints (dimensions, mates, DOF) and other physical properties that may be used to define geometry of components. The Skelton concept has been utilized in converting a 3D solid model to a skeleton model that is used in assembly modelling [58]. According to the same author, using assembly interface geometries with their constraints in the assembly model could improve design efficiency and reduce the cost of product development. Assembly interface geometries facilitate the integration between process behaviour knowledge and product design form knowledge [59]. More details about utilizing the Skelton model in assembly design and assembly sequence planning could be found in [59].

The Virtual Assembly Design Environment (VADE) approach has been used to integrate product design and APP. According to [60], VADE is "a Virtual Reality (VR) based engineering application which allows engineers to "*evaluate, analyze, and plan the assembly of mechanical systems*". VADE has been used to integrate assembly design applications such as CAD software with other assembly planning applications such as CAPP, assembly simulation, robot path planning and assembly equipment design. Wang et al. (2006) [61] introduced a novel approach to integrate VR and CAD for virtual assembly. Virtual Assembly Process Planning (VAPP) has been used to provide a more efficient, intuitive and convenient method for assembly process planning [62].

The diversity and the complexity of assembly design have attracted many researchers to utilize ontology capabilities in integrating and migrating assembly knowledge and provide rich conceptualization of a complex domain such as the assembly domain. Ontology-based approaches for integrating product design and APP can be categorized further into two main approaches: Mereotopological design approach and assembly design knowledge approach. The assembly design semantics in the first approach is based on mereotopological description for assembly knowledge, while in the second approach the assembly design semantics is based on geometrical and topological description for the part knowledge and mates for the assembly knowledge.

Mereotopology (MT) is a branch of logic that has the ability to provide qualitative formalization of two fundamental relationships between entities: parthood (i.e. one entity being part of another) and connection [63]. Design mereotopology (DMT) [63] is developed in order to identify and logically describe some regions within geometrical entities and the interrelations between regions within a product. Ontological axioms have been used to restrict DMT to describe only regions of interest in the product design model. In this sense, (DMT) has been utilized in ontology-based assembly design modelling. Kim et al (2008) [64] developed a methodology to represent and differentiate assembly joint information in a collaborative product design environment based on formal mereotopological ontology. In this ontology, assembly joining constraints are explicitly represented by using SWRL rules and OWL triples, which are derived from mereotopological definitions. The main contribution in [64] is the developed semantic definitions (e.g., for machine interpretation) for a theoretical mereotopological foundation of the assembly design. Demoly et al, (2012) [65] have proposed a novel approach called PRONOIA (PROduct relatioNships description based On mereotopologIcAl theory) for defining product relationships based on mereotopological theory. This approach is based on using product relationships described by mereotopological primitives, and Assembly Sequence Planning (ASP) in order to integrate assembly modelling and planning. The final integration stage is implemented by using an ontology with OWL-DL and SWRL languages.

Gruhier et al. [66] introduced a novel mereotopological approach called JANUS (Joined AwareNess and Understanding in assembly-oriented design (AOD) with mereotopology) for integrated assembly design and assembly sequence planning. The JANUS approach based on describing the assembly design and assembly process in the three dimensions (i.e. spatial, temporal and spatiotemporal) in order to describe the evolution of the product assembly. The spatial dimension describes the spatial mereotopological relationships between spatial objects. The temporal dimension describes the lifetime of a temporal object, which is broken down into several smaller temporal objects, which are created each time a part (object) is added into the assembly. The object change is generally associated to an assembly process. The spatiotemporal dimension is used to describe an objects spatial location over time during deformation, modification or transformation. A proposed ontological development framework called PRONOIA2 [67], an extension of PRONOIA, is fully based on the JANUS mereotopological approach. PRONIOIA2 is composed of three layers; meta ontology, domain ontology and application ontology. It is implemented using OWL-DL and SWRL languages. This ontology-based approach is very promising in integrating assembly design and APP, nevertheless it has been used mainly to integrate the product design process and Assembly Sequence Planning (ASP). More efforts are needed to use this approach in integrating product design from one side and manufacturing (processes and resources) from the other side.

For semantic integration, several researchers have used ontology-based approaches to integrate assembly design and APP. Delamer and Lastra [68] proposed an ontology for the modelling of assembly processes. This ontology enables reasoning and queries of assembly knowledge based on OWL and SWRL rules. Lohse et al. [69] proposed an equipment ontology to support reconfigurability in manufacturing /assembly systems by facilitating decisions related to the selection of assembly equipment. Lanz et al. [70] proposed an ontology to capture product and process related assembly knowledge and to integrate product design and assembly simulation. Kim et al. [71] proposed an assembly design ontology which provides formal description of assembly design related knowledge. The main focus of that ontology was on assembly joint formalism where a welding joint is provided as an example. Mostefai et al. [72] proposed a product design ontology to capture product design knowledge for supporting the product development process. This ontology supports inferences and queries but without reasoning capability. Fiorentini et al. [73] proposed an ontology for assembly design representation. Their ontology is based on the Open Assembly Model (OAM) [55] with a reasoning capability to support designer's decisions. To model APP knowledge, Huang et al. [74] proposed an ontological model that attempts to cover some important aspects in APP knowledge, such as assembly requirement, spatial information, assembly operation and assembly resource. Since no ontological formalism is reported for their work it represents an uncompleted attempt to model APP knowledge. Imran et al. [21] proposed a novel framework in the form of an assembly reference ontology, which can provide a common semantic base to support interoperability and knowledge sharing across the assembly design and assembly process planning domains. Their work represents a demonstration of the application of formal ontologies for assembly knowledge sharing and specifically focuses on concepts related to tolerance and fits, assembly feature, assembly method and assembly resource. The ontological formalism used by [21] is the knowledge frame language (KFL) [75], which is based on Common Logic (CL) [76]. According to the [77], CL is more expressive than (OWL) and more capable of representing the semantics of complex manufacturing concepts and relationships.

A majority of the researchers who carried out ontology based approaches in the manufacturing and/or assembly domains have used OWL as ontological formalism. The majority also has used SWRL to implement the reasoning capability in their ontologies. A few researchers have used other ontological formalism such as Common Logic (CL) and Knowledge Frame Language (KFL). Fig.1 also gives an indication about the ontological formalism that has been used for each ontology under the ontology-based approach. Next section will provide a review and comparison between different ontology formalism languages that have been used in published literature.

III. ONTOLOGY FORMALISM LANGUAGES

In published literature, many ontology development languages have been used to provide representation of the internal structure of an ontology [78], [79] and [80]. Ontology development languages can be classified into three main categories: (A) ontology mark-up languages (B) ontology schematic languages and (C) general ontology languages. In the following some of the most popular ontology languages that are mainly used in developing ontologies under these main categories:

A. *Ontology mark-up languages:*

Resource Description Framework (RDF) & RDF (Schema) (RDFS): is a language developed by W3C and used mainly to model information and web resources. RDF supports interoperability across a range of applications [81]. RDF is an object-attribute-value triple [82]. RDF (Schema) (RDFS) is considered as a semantic extension of RDF [83]. RDFS has the ability to provide as set of classes, subclasses, properties and sub properties to model the ontology structure. RDF / RDFS are considered as Lightweight formal ontology language. RDF / RDFS support Reasoning and considered as a web language with Protégé as ontology development tool. More information available at [84].

RDF Limitations: (i) The RDF data model does not provide mechanisms for defining the relationships between properties (attributes) and resources [85]. (ii)RDFS is mainly used to model simple ontologies [86]. (iii) RDF/RDFS cannot support the modelling of ternary or higher-arity relations.

Ontology Web Language (OWL): [87] is considered as a defacto standard highly expressive web semantic ontology language. OWL is based on a meta data model which is an abstract set of rules for representing, interpreting and processing content of information [87]. OWL is considered as a good tool for creating bridges between different domains. OWL provided more machine interpretability than the XML, RDF, and RDFS OWL also provides better interoperability to web content as compared to RDF and RDF Schema. OWL is considered as heavyweight expressive formal language but it is still limited in its expressive power, since it is based on RDF/RDFS. OWL supports automated reasoning [88] and it is a web language.

OWL has several limitations : (i) it's limited ability to represent relations and functions, For example, OWL cannot directly support relations having arity more than 2 and functions having arity more than 1 [89]. (ii) Another limitation is its complex and less efficient reasoning capabilities, this is due to the fact that OWL does not have conjunction, disjunction, and negation operators [88]. (iii) OWL semantics are based on the Open World Assumption (OWA), which means things are not known to be true may not be necessarily false [88]. Palmer et al. [89] argues that complex domains such as manufacturing and assembly are facts driven and need certainty that can be supported by the closed world assumption [89].

In order to overcome these limitations, several flavors have been developed from OWL, the most important are OWL Lite, OWL DL, and OWL Full [87], [90].

OWL Lite is aimed for applications requiring classification hierarchies and simple constraints. It trades expressivity for efficiency of reasoning [80]. OWL DL it is the most important OWL sub-language, it's OWL with formal specifications highly affected by descriptive logic [91]. The fundamental concept of Description Logic is the use of axioms, which are logical statements relating roles (properties) and concepts (classes); this gives OWL-DL very powerful expressiveness so it supports applications requiring maximum expressiveness with Decidable reasoning that is considered less efficient than OWL Lite. OWL Full is used for application where maximum expressiveness is required as well but with Undecidable Reasoning. Protégé is used as OWL ontology development tool.

The eXtensible Markup Language (XML): XML is considered as the first web language to separate the markup of web content from web presentation, facilitating the representation of task-specific and domain- specific data on the web [92]. All the other mark-up ontology languages are built on XML. XML file document is considered as a powerful data format, which is proposed to transport and store data. XML solved many problems about representing, organizing, storing, sharing data in the face of enormous growth in size and complexity. However, an XML file does not provide any means regarding interpretation of the data. This is because it lacks semantics; designed to describe the structure of a document not the content. XML is considered as a lightweight formal ontology language. Another limitation for this language is that it does not provide any reasoning.

B. Ontology schematic languages

Unified Modelling Language (UML / UML 2): UML is the standard for Object- oriented graphical modelling language. It provides a way for the modelling of knowledge and information. UML provides various diagrams like Class Diagrams, Use-Case Diagrams, and Communication diagrams. The most widely used ones are the class diagrams. The main feature of UML 2 is its ability to represent ternary and higher order relations [89]. UML2 can be used for the lightweight representation of common logic (CL) based ontologies as it is presented in [21]. As object oriented language, Java, C and C++ etc codes can be generated from UML 2 diagrams. Enterprise architect is used as a UML ontology development tool.

C. General ontology languages

Common Logic (CL): is a formal ontological language based on the first order logic, and is used for sharing and transmission of information [76]. CL has higher expressive power and supports better inference and reasoning ability as compared to the languages like XML, RDF/RDFS and OWL-Lite, OWL-DL and OWL-Full [93]. The main features of CL over other ontological languages are: (i) CL is based on Closed World Assumption (CWA) [94]; according to date [95] CWA states that everything stated or implied is true and everything else is false. (ii) CL supports ternary relations (and relations having arity more than 3) [96] (iii) binary functions (and functions having arity more than 2), conjunction, disjunction, and the negation operators [89]. According to [20] and [96] these features make CL very suitable in modelling complex domains such as assembly and more competent than OWL in rigorously defining the semantics which is a key requirement for heavyweight modelling to support knowledge sharing as reported in [96].

IV. Conclusions

The paper showed a possible use for ontologies that is aimed at overcoming some of the problems related to integrating and knowledge sharing between product design and manufacturing with more focus on the integration of assembly design and APP domains in manufacturing environment.

Assembly is considered as one of the most complicated tasks in a manufacturing environment [97], and those tasks have not been investigated deeply compared to other manufacturing tasks [98]. The importance of assembly becomes evident from claims such as that 53% of manufacturing time is consumed in carrying out assembly tasks [99]. This paper showed how the complexity and diversity of the assembly design have attracted many researchers to utilize ontology capabilities in integrating and migrating assembly knowledge and provide rich conceptualizations within the complex assembly domain. More specifically, this paper focused in utilizing ontologies in integrating assembly design and APP. Assembly design and assembly process planning (APP) are very important engineering domains of successful manufacturing system design, which require an efficient collaborative environment for best utilization of the assembly resources. However, these domains represent different perspectives in understanding the same concepts, and both domains use different software applications, which might cause interoperability issues. The paper showed how ontology-based system could help to overcome problems such as interoperability, modelling and integration between these two domains.

The last part of the paper is dedicated for ontology development languages. Manufacturing and assembly needs very expressive and rich ontological languages with potential features to model this complex domain. According to Negri, [80] ontology development language have to meet four main requirements for semantic language representation of the manufacturing domain, namely support of conceptual modeling and data storage, ease of use and maintenance, interoperability, and automated reasoning. According to the same author [80] OWL and the OWL sublanguages: OWL Lite, OWL DL are recommended for modelling manufacturing domain One of the limitations of this recommendation is that it does not include any general ontology development languages such as CL.

From the study performed in this paper CL language is highly recommended to model manufacturing domain. CL has proofed to be very powerful semantic modelling tool regards modelling of manufacturing and assembly domains with effective knowledge representation and reasoning. This recommendation is based on the work performed by several researchers using this language such as [89], [20], [96] and [100]. Their works represent a valuable recommendation for the ability of this language to perform highly expressive semantic modelling in manufacturing and assembly domains.

Regardless of all its potential features CL has not got widespread acceptability in information systems

Community [101]. A study by carried out by Cardoso [102] showed that the OWL and

RDF(S) are the most used languages for data exchange and knowledge sharing in industry and academia with a percentage of 75.9 %, while CL only deployed with about 2.6 %. The widespread acceptability of OWL is due to free open source OWL ontology language tool protégé and its simplicity, also the availability of the OWL resources as a literature and examples over the internet is another factor. Another important factor is that protégé is a Java-based application, so it's easy to deploy Java functions and applications to increase the capabilities of the OWL language. On the other hand to use CL user has to use UML for the lightweight representation of the ontology, which requires software as enterprise architect. Enterprise architect is not free open source software and also it requires efforts to learn ontological modelling compared to protégé. Also CL as a rich ontological language needs more efforts to learn compared to OWL.

References

[1] R. Studer, V.R. Benjamins, and D. Fensel, Knowledge engineering: Principles and methods. Data & Knowledge Engineering, 1998, 25(1-2), pp. 161-197 .

[2] T.R. Gruber, A translation approach to portable ontology specifications. Technical Report KSL 92-71. Stanford, California 94305: Knowledge System Laboratory, Computer science depatrment, Stanford University, 1993.

[3] M. Uschold, and M. Gruninger, Ontologies: Principles, methods and applications. Knowledge Engineering Review, 11(2), 1996, pp. 93-155.

[4] A. Gomez-Perez, M. Fernandez-Lopez, & O. Corcho, Ontological Engineering with examples from the areas of Knowledge Management, e- Commerce and the Semantic Web. London: Springer, 2004.

[5] R. Young, A. G. Gunendran, A. F. Cutting-Decelle, & M. Gruninger, Manufacturing knowledge sharing in PLM: a progression towards the use of heavy weight ontologies. International Journal of Production Research, 2007, 45(1), 1505-1519.

[6] N. Guarino, Some Ontological Principles for Designing Upper Level Lexical Resources, Proc. of the First International Conference on Lexical Resources and Evaluation , 1998, pp. 527-534.

[7] G. Van Heijst, A.T. Schreiber, and B.J. Wielinga, Using explicit ontologies in KBS development. International Journal of Human-Computer Studies, 1997, 46(2-3), pp. 183-292.

[8] S. Sanchez-Alonso, and E. Garcia-Barriocanal, Making use of upper ontologies to foster interoperability between SKOS concept schemes. Online Information Review, 2006, 30(3): pp. 253-277.

[9] H.A. ElMaraghy, Flexible and reconfigurable manufacturing systems paradigms, International Journal of Flexible Manufacturing Systems, 2005, 17: 261.

[10] D. Semere, M. Onori, A. Maffei, R. Adamietz, Evolvable assembly systems: coping with variations through evolution, Assembly Automation, 2008, Vol. 28 Issue: 2, pp.126-133.

[11] M. Onori, J. Barata, Evolvable production systems : mechatronic production equipment with process-based distributed control, IFAC Proceedings Volumes, Volume 42, Issue 16, 2009, Pages 80-85, ISSN 1474-6670.

[12] Y. Alsafi, V. Vyatkin, Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing, Robot. Comput. Integr. Manuf., 2010, 26 (4), 381–391.

[13] S. Lemaignan, A. Siadat, J. Dantan, A. Semenenko, "MASON: A Proposal For An Ontology Of Manufacturing Domain," Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications, 2006.

[14] P. Chhim, R.B. Chinnam, & N. Sadawi, Product design and manufacturing process based ontology for manufacturing knowledge reuse, J Intell Manuf, doi:10.1007/s10845-016-1290-2, 2017.

[15] N. Chungoora, & R. Young, A Framework to Support Semantic Interoperability in Product Design and Manufacture. In A. Bernard, Global Product Development, Proceedings of 20th CIRP Design Conference (pp. 435-443). Springer-Verlag, 2011.

[16] N. Anjum, J. Harding, R. Young, K. Case, Z. Zahid Usman, T. Changoora, Verification of knowledge shared across design and manufacture using a foundation ontology, International Journal of Production Research, 2013, 51:22, 6534-6552, DOI: 10.1080/00207543.2013.798051.

[17] Z. Usman, R. Young, N. Chungoora, C. Palmer, K. Case, & J. Harding, Towards a formal manufacturing reference ontology, International Journal of Production Research, 2013, 51:22.

[18] M. Cai, W.Y. Zhang, K. Zhang, ManuHub: a semantic web system for ontology based service management in distributed manufacturing environments, IEEE Trans. Syst. Man Cybern. A: Syst. Hum. 41 (May (3)) , 2011,574–582.

[19] H.K. Lin, J.a. Harding, A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration, Comput. Ind. 58 (5) (2007) 428–437.

[20] N. Chungoora, A Framework to Support Semantic Interoperability in Product Design and Manufacture. Loughborough University, (2010).

[21] M. Imran, & B.Young, The application of common logic based formal ontologies to assembly knowledge sharing, Journal of Intelligent Manufacturing, (2015), 26: 139. doi:10.1007/s10845-013-0768-4.

[22] C. Sean, R. Young, K. Case, J. Harding, J. Gao, S. Dani and D. Baxter, Knowledge reuse in manufacturability analysis, Robotics and Computer-Integrated Manufacturing, 2008. 24(4):508-513.

[23] D. Sánchez, M. José, and M. Esperanza, The road towards ontologies In Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems, ed. Rajiv Kishore and Ram Ramesh Raj Sharman: Springer, 2007.

[24] K. Wang, & S. Tong, An Ontology of Manufacturing Knowledge for Design Decision Support. Wireless Communications, Networking and Mobile Computing, (2008).

[25] Z. Jiehan and R. Dieng-Kuntz, Manufacturing Ontology Analysis and Design: Towards Excellent Manufacturing, Industrial Informatics, 2004. INDIN '04. 2004 2nd IEEE International Conference on, pp. 39-45, 2004.

[26] H. K. Miao, N. Sridharan, and J. J. Shah, CAD-CAM integration using machining features, International Journal of Computer Integrated Manufacturing, pp. 296–318, 2002.

[27] B. Hasan J. Wikander and M. Onori, Assembly Design Semantic Recognition Using SolidWorks-API. In: International Journal of Mechanical Engineering and Robotics Research, (2016). Vol. 5, No. 4, pp. 280-287: 10.18178/ijmerr.5.4.280-287.

[28] B. Hasan, & J. Wikander, Product Feature Modelling for Integrating Product Design and Assembly Process Planning. World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering, (2016), 10(10), 1693-1703.

[29] G. Chen, Y.-S. Ma, G. Thimm & S.-H. Tang, Unified Feature Modeling Scheme for the Integration of CAD and CAx, Computer-Aided Design and Applications, (2004), 1:1-4,595-601, DOI: 10.1080/16864360.2004.10738303.

[30] X. Zhou, Y. Qiu, G. Hua, H. Wang, X. Ruan, A feasible approach to the integration of CAD and CAPP, In Computer-Aided Design, Volume 39, Issue 4, 2007, Pages 324-338, ISSN 0010-4485, https://doi.org/10.1016/j.cad.2007.01.005.

[31] G. Vosniakos and B. Davies, An IGES post-processor for interfacing CAD and CAPP of 2 ½ D prismatic parts, International Journal of advanced Manufacturing Technology , (1990), 5:135-164.

[32] Y. Junyu, Z. wei, STEP_integration of CAD/CAM & CAPP in numerical control machining, IEEE Journal, (2009).

[33] A. Balakrishna, R. Babu, D. Rao, D. Raju, and S. Kolli, Integration of CAD/CAM/CAE in Product Development System Using STEP/XML. Concurrent Engineering, (2006), 14(2), pp.121-128.

[34] S. Amaitik, and S. Kiliç, An intelligent process planning system for prismatic parts using STEP features. The International Journal of Advanced Manufacturing Technology, (2006), 31(9-10), pp.978-993.

[35] K. Dwijayanti, and H. Aoyama, Development of Automatic Process Planning System for Turning-Milling Center with 2 Turrets System. Journal of the Japan Society for Precision Engineering, (2015), 81(5), pp.471-480.

[36] A. Nassehi, R. Liu, and S. Newman, A new software platform to support feature-based process planning for interoperable STEP-NC manufacture. International Journal of Computer Integrated Manufacturing, (2007), 20(7), pp.669-683.

[37] C. Dartigues, P. Ghodous, M. Gruninger, D. Pallez, and R. Sriram, CAD/CAPP Integration using Feature Ontology. Concurrent Engineering, (2007), 15(2), pp.237-249.

[38] S. Abdul-Ghafour, P. Ghodous, B.Shariat, E. Perna, Ontology Development for the Integration of CAD Models in a Collaborative Environment. In: Frey D., Fukuda S., Rock G. (eds) Improving Complex Systems Today. Advanced Concurrent Engineering, (2011), Springer, London.

[39] D. T. Semere, S. Dilshad , & B. Lindberg, Machining Ontology and Knolwedge Modelling, Royal Institute of Technology, Institute of Industrial Production, (2007).

[40] A. Tursi, H. Panetto, G. Morel and M. Dassisti, Ontological approach for productscentric information system interoperability in networked manufacturing enterprises, Annual Reviews in Control, 2009, 33(2):238-245.

[41] ISO/TS-10303-1022. Industrial automation systems and integration --Product data representation and exchange --Part 1022: Application module: Part and Version Identification, (2004).

[42] ISO/ IEC 62264. Enterprise-control system integration, Part 1. Models and terminology, Part 2: Model object attributes. Geneva: ISO/IEC, (2002).

[43] X. Chang, R. Rai, and J. Terpenny, Development and Utilization of Ontologies in Design for Manufacturing. Journal of Mechanical Design, (2010), 132(2), p.021009.

[44] S., Wei, M., Qin-yi and G., Tian-yi, An Ontology-Based Manufacturing Design System. Information Technology Journal, 2009, 8: 643-656.

[45] K. Wang, & S. Tong, An Ontology of Manufacturing Knowledge for Design Decision Support. Wireless Communications, Networking and Mobile Computing, (2008).

[46] X.F. Zha, & Hongwu. Du, A PDES/STEP-based model and system for concurrent integrated design and assembly planning. Computer-Aided Design, (2002), 34. 1087-1110. 10.1016/S0010-4485(01)00186-5.

[47] LS.Wierda, Linking Design, Process Planning and Cost Information by Feature Based Modelling, Journal of Engineering Design, 1991, 2(1): pp. 3-19.

[48] OW. Salomons, FJ. Houten, A. Van, HJ. Kals, Review of Research in Feature-Based Design, 1993; 12(2): pp. 113–132.

[49] J. Shah, and M. Rogers, Functional requirements and conceptual design of the feature-based modelling system, Computer-Aided Engineering Journal, pp. 9-15, 1988.

[50] J. Shah, and M. Rogers, Assembly modeling as an extension of feature-based design, Research in Engineering Design; 5: pp. 218–37, 1993.

[51] W. Van Holland, Assembly features in modelling and planning, PhD thesis, Delft: Delft University of Technology; 1997.

[52] H. Bley, and C. Franke, Integration of Product Design and Assembly Planning in the Digital Factory. CIRP Annals, (2004), 53(1), pp.25-30.

[53] K. Case, W.A. Wan Harun, Feature-based representation for manufacturing planning, International Journal of Production Research, (2000) , 38:17, 4285-4300, DOI: 10.1080/00207540050205091.

[54] R. Bidarra, WF. Bronsvoort, Semantic feature modelling, Computer-Aided Design, (2000), 32(3): pp. 201-225.

[55] S. Rachuri, Y. Han, S. Foufou, S. Feng, U. Roy, F.Wang, R. Sriram, and K. Lyons, A Model for Capturing Product Assembly Information. Journal of Computing and Information Science in Engineering, (2006), 6(1), p.11.

[56] W. Hui, X. Dong, D. Guanghong and Z. Linxuan, Assembly planning based on semantic modeling approach. Computers in Industry, (2007), 58(3), pp.227-239.

[57] W.F. Bronsvoort, A. Noort, Multiple-view feature modelling for integral product development, Computer-Aided Design, (2004) , Volume 36, Issue 10, Pages 929-946, ISSN 0010-4485, https://doi.org/10.1016/j.cad.2003.09.008.

[58] C-S. Wang, Web-based modular interface geometries with constraints in assembly models, Computers & Industrial Engineering, (2009),Volume 56, Issue 4, Pages 1675-1686.

[59] F. Demoly, L.Toussaint, B. Eynard, D. Kiritsis, S. Gomes, Geometric skeleton computation enabling concurrent product engineering and assembly sequence planning, Computer-Aided Design, (2011), Volume 43, Issue 12.

[60] S.Jayaram, U. Jayaram, Y.Wang, H. Tirumali, K.Lyons, P. Hart, Vade: a virtual assembly design environment. IEEE Computer Graphics and Applications 1999,19(6):44–50.

[61] H. Wang, J. R. Li, and H. Q. Gong, A CAD-Linked Virtual Assembly Environment, 2006, International Journal of Production Research 44: 467–486.

[62] Y. Jun , J. Liu , R. Ning & Y. Zhang, Assembly process modeling for virtual assembly process planning, International Journal of Computer Integrated Manufacturing, (2005), 18:6, 442-451, DOI: 10.1080/09511920400030153.

[63] FA. Salustri, Mereotopology for product modeling. Anew frame work for product modeling based on logic.Journal of Design Research, (2002), 2(1):22.

[64] K.Y. Kim, H.J. Yang, and D.W. Kim, Mereotopological Assembly Joint Information Representation for Collaborative Product Design. Robotics and Computer-Integrated Manufacturing, (2008), Volume 24, Issue 6, Pages 744-754.

[65] F. Demoly, A. Matsokis, D. Kiritsis, A mereotopological product relationship description approach for assembly oriented design, Robot. Comput.-Integr. Manuf., (2012), 28 (6) 681–693.

[66] E. Gruhier, F. Demoly, S. Abboudi, S. Gomes, A Spatiotemporal Mereotopology-Based Theory for Qualitative Description in Assembly Design and Sequence Planning. In: Gero J., Hanna S. (eds) Design Computing and Cognition '14., (2015) Springer, Cham

[67] E. Gruhier, F. Demoly, S. Abboudi, S. Gomes, A formal ontology-based spatiotemporal mereotopology for integrated product design and assembly sequence planning, Advanced Engineering Informatics, (2015), Volume 29, Issue 3, August, Pages 495-512, ISSN 1474-0346, https://doi.org/10.1016/j.aei.2015.04.004.

[68] I. M., Delamer, & J. L. Lastra, Ontology Modeling of Assembly Processes and Systems using Semantic Web Services. International Conference on Industrial Informatics, (2006). (pp. 611-617). IEEE.

[69] N. Lohse, H. Hirani , & S. Ratchev, Equipment Ontology For Modular Reconfigurable Assembly Systems. International Journal of Flexible Manufacturing System, (2006), 17, 301-314.

[70] M. Lanz, F. Garcia , T., Kallela, & R. T, Product-Process Ontology For Managing Assembly Specific Knowledge Between Product Design and Assembly System Simulation. In S. K. Svetan Ratchev, Micro-Assembly Technologies and Applications, (2008), (Vol. 260, pp. 99-108). Boston: Springer.

[71] K. Y., Kim, D. G. Manley, and H. J. Yang, Ontology-based Assembly Design and Information Sharing for Collaborative Product Development. Computer-Aided Design, (2006), Vol. 38, No. 12, pp. 1233-1250.

[72] S. Mostefai, A. Bouras, & M. Batouche, Effective Collaboration in Product Development via a Common Sharable Ontology. International Journal of Computational Intelligence, (2008), 206-212.

[73] X. Fiorentini, I. Gambino, V.-C. Liang, S. Rachuri, M. Mani & C. Bock, An Ontology for Assembly Representation. National Institute of Standards and Technology, (2007).

[74] Z. Huang, L. Qiao, N. Anwer, Y. Mo, Ontology Model for Assembly Process Planning Knowledge. 21st International Conference on Industrial Engineering and Engineering Management, (2014), zhuhai, China.

[75] KFL Reference, Ontologies in KFL. HIGHFLEET®, (2012).

[76] ISO/IEC-24707. Information technology — Common Logic (CL): a framework for a family of logic based, (2007). Retrieved 11 22, 2017, from Common Logic Standard: https://www.iso.org/standard/39175.html.

[77] DL. McGuinness, F. van Harmelen, OWL web ontology language overview, (2009), http://www.w3.org/TR/owl-features/.

[78] J.R.G. Pulido, M.A.G. Ruiz, R. Herrera, E. Cabello, S. Legrand, D. Elliman, Ontology languages for the semantic web: A never completely updated review, In Knowledge-Based Systems, Volume 19, Issue 7, 2006, Pages 489-497, ISSN 0950-7051, https://doi.org/10.1016/j.knosys.2006.04.013.

[79] O. Corcho, M. Fernández-López, and A. Gómez-Pérez, Methodologies, tools and languages for building ontologies. Where is their meeting point?. Data & Knowledge Engineering, (2003), 46(1), pp.41-64.

[80] E. Negri, L. Fumagalli, M. Garetti, & L. Tanca, Requirements and languages for the semantic representation of manufacturing systems. Computers in Industry, (2016), 81, 55–66.

[81] O., Lassila, & R. R. Swick, Resource Description Framework (RDF) Model and Syntax Specification, (1999). Retrieved 11 21, 2017, from http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

[82] S. Decker, S. Melnik, F. V. Harmelen, D. Fensel, M. Klein, J. Broekstra, I. Horrocks, The Semantic Web: The Roles of XML and RDF. IEEE Internet Computing, (2000). 4(5), 63-74.

[83] D. Brickley, & R. Guha, RDF Vocabulary Description Language 1.0: RDF Schema, (2004), Retrieved 11 21, 2017, from http://www.w3.org/TR/rdf-schema/

[84] O. Corcho, A. Gomez-Perez, A Roadmap to Ontology Specijication Languages. 12th International Conference in Knowledge Engineering and Knowledge Management, Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin, Oct. 2000, pp. 80-96.

[85] A. Gomez-Perez, O. Corcho, Ontology Languages for the Semantic Web. IEEE Intelligent Systems & their applications. 17(1). JanuarylFebruary 2002. pp: 54-60.

[86] Z. J. Pan, Resource Description Framework. In S. Staab, & R. Studer (Eds.), Handbook on Ontologies, (2009). (pp. 71-90). Berlin Heidelberg: Springer- Verlag.

[87] W3C. OWL Web Ontology Language Overview. (D. L. McGuinness, & F. v. Harmelen, Editors), (2004). Retrieved 11 21, 2017, from http://www.w3.org/TR/owlfeatures/

[88] K. Sengupta, & P. Hitzler, Web Ontology Language (OWL). Wright State University, Joshi Research Center, Dayton, (2014). Retrieved 11 21, 2017, from http://corescholar.libraries.wright.edu/cgi/viewcontent.cgi?article=1183&context=cse

[89] C. Palmer, N. Chungoora, R.Young, A. G. Gunendran, Z. Usman, K. Case, & J. A. Harding, Exploiting unified modelling language (UML) as a preliminary design tool for Common Logic-based ontologies in manufacturing. International Journal of Computer Integrated Manufacturing, 1-17, (2012).

[90] G. Antoniou, & F. v. Harmelen, Web Ontology Language: OWL. In S. Staab, & R. Studer (Eds.), Handbook on Ontologies (pp. 91-110), (2009), Berlin: Springer-Verlag.

[91] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, The Description Logic Handbook. Cambridge University Press, 2002, Cambridge, London.

[92] J.R.G. Pulido, M.A.G. Ruiz, R. Herrera, E. Cabello, S. Legrand, D. Elliman, Ontology languages for the semantic web: A never completely updated review, In Knowledge-Based Systems, Volume 19, Issue 7, 2006, Pages 489-497.

[93] A. Sánchez-Ruíz, K. Umapathy, & P. Hayes, Toward Generic, Immersive, and Collaborative Solutions to the Data Interoperability Problem which Target End-Users. Journal of Computing Science and Engineering, (2009), 3(2), 127-141.

[94] N. Chungoora, A. Cutting-Decelle, R. Young, G. Gunendran, Z. Usman, J. A. Harding, & K. Case, Towards the ontology-based consolidation of production-centric standards. International Journal of Production Research, (2013), 51(2), 327-345.

[95] C. J. Date, Logic and Databases the Roots of Relational Theory. Trafford Publishing, (2007).

[96] M. Imran, Towards an Assembly Reference Ontology for Assembly Knowledge Sharing. Ph.D. Loughborough University, (2013).

[97] A. Delchambre, Computer-aided Assembly Planning. Chapman & Hall, (1992).

[98] D. E., Whitney, Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development. New York, Oxford: OXFORD UNIVERSITY PRESS, (2004).

[99] S. H. Cho, Mechanical Assembly. In F. Kreith, The Mechanical Engineering Handbook Series. CRC Press LLC, (2005).

[100] N-A., Anjum, Verification of Knowledge Shared Across Design and Manufacture Using Foundation Ontology. PhD Thesis: Loughborough University, 2011.

[101] H. S. Delugach, Representing metadata constraints in Common Logic. Int. J. Metadata, Semantics and Ontologies, (2009). 4(4), 277-286.

[102] J. Cardoso, The Semantic Web Vision: Where are We? IEEE Intelligent Systems, 22-26, (2007).