Ontological CAD Data Interoperability Framework

Luis Enrique Ramos García*
*University of Bremen
Cartesium, Enrique-Schmidt-Strasse. Bremen, Germany
s_7dns7r@uni-bremen.de
*Universidad Nacional Abierta
lramos@una.edu.ve

***Abstract*—Computer Aided Design (CAD) has become one of the fundamental activities in the modern industry. Nowadays several products are developed and modeled using this technology. Nevertheless, extracting product features from these kind of files to use them in production processes and parametric data exchange among heterogeneous CAD systems are still difficult to achieve. This work aims to propose OWL as CAD Data Exchange Format, giving the possibility for the addition of more descriptive information of products and processes in one self-content and self-descriptive file. With this CAD - OWL integration the feature extraction is facilitated, because this CAD - OWL model becomes a Knowledge Base and the reasoning tools of the Semantic Web become available. In this work a standard CAD file was mapped into the Web Ontology Language (OWL) and visualized using the Protégé API's architecture in order to deal with such problems.**

***Keywords*-Ontology; Web Ontology Language (OWL); Computer Aided Design(CAD); Protégé.**

I. Introduction

Computer Aided Design has been an important approach for designing of mechanical parts since the beginning of the 1970's. This technology has a fundamental role in the industrial processes of manufacturing. Software CAD tools such as AutoCAD®[1], Pro Enginer®[2], Free CAD [3] and others are used nowadays at the beginning of these processes, specifically in the products design phase (see Fig. 1). After this design is ended, the production process continues with the production planning phase, where among other things, tasks as determining manufacturing, getting valid raw material suppliers, calculating costs, time, quantity of production, selecting the kind of machines needed, sequence of operation, etc., take place [4]. In the manufacturing phase this design becomes a product. Cause these activities are highly time consuming, and repetitives, there have been efforts to make automatic extraction of information from CAD files using parser computer programs in order to generate Automatic Production Planing and Manufacturing in two research and application areas known as Computer Aided Process Planing (CAPP) and Computer Aided Manufacturing (CAM) [5]. This automated interaction and evolution has been limited cause by the semantic weakness of the CAD standards for these objectives, which in fact,

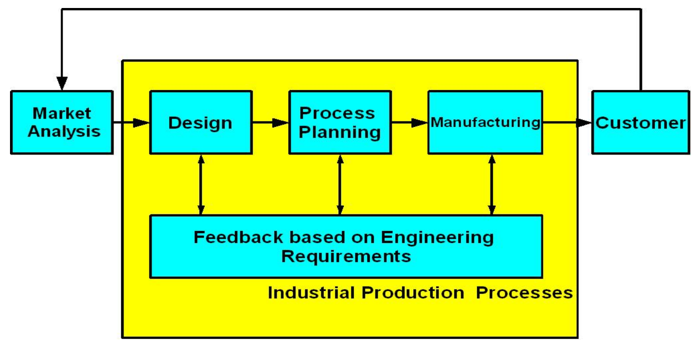

Figure 1. Industrial Production Process

are not designed for this kind of interaction, but for the representation of geometrical elements, such as lines, circles, surfaces, solids of revolution and so on [6]. Besides, in the process planing and production phases it is necessary interact with expert knowledge, which is difficult to represent and reuse [7]. To deal with this limitations and in order to facilitate the interaction with a CAD design we propose the ontological approach as a way to represent expert knowledge and the semantic web as a platform for CAD - CAPP - CAM Processes Interaction Automation. In this article a first experience is presented, where a CAD data exchange format called DXF was used to classify a two dimensional design in a set of classes and instance in order to populate a geometry ontology. By making queries to the ontology this design was rebuild for visualization using Protégé 3.4.4 and Java 2D as a prototyping platform.

II. Previous Works

Works related with the extraction of features from a CAD file have been reported since the 1980s. Henderson and Anderson [8] used PROLOG, a rules-based language to express a set of necessary and sufficient conditions to classify features. They found that PROLOG had limitations to handle trigonometric functions required to deal with general angular relationships. In their work, even, a not

 ISBN: 978-1-61208-104-5

optimized sequence of production was generated, mechanical parts were manufactured automatically. Sam Lazaro *et al* [9] developed an intelligent system to help designers in developing metal sheet parts, using an object based Knowledge Base System development package called NEXTPERT Object. Design rules were represented in this system as a set of IF <list of conditions>THEN <hypotesis><list of actions>. Although the work of [9] did not deal with the generation of production sequences, it is an example of the use of Knowledge Bases to store expert knowledge and its re-utilization to alert designer engineers when a design rule was violated. Soman *et al* [11], developed a system using C++, in which rules were used in the automatic production of a design for a sheet metal part. Here the application of a certain group of rules was controlled by a list of conditional loops, facilitating the automation of the design production process. In this work an estimation of manufacturing time and cost prediction was reported, although it was not described any optimization module or process.

The critics made to these methodologies, based on rules, are the rules themselves, because the designer of such system needs to develop rules enough explicit for each case, some languages as C++ are not intended to make rules, some rule languages as PROLOG has limitation for certain mathematical operations, which limits these programs for this kind of application.

In recent years, there has been a movement toward the utilization of the ontological approach in engineering applications for the representation of CAD models to capture feature semantics and to use such model among different system maintaining the designer's intent. Ghafour *et al* [12], presented an architecture for a Data Exchange among different CAD software tools, in where ontologies are proposed to represent the terminologies of some commercial CAD software tools, and a main ontology would serve as a Common Design Feature Ontology. He proposed to write and store ontologies on each CAD system using the Web Ontology Language (OWL) a W3C standard, generating an ontology of such systems. These ontologies have to be mapped in a Common Design Ontology to make them interoperable among different software applications. Similarly, Odd and Vasilakis [13] proposed an ontology of CAD model information, this proposal is described as an introduction to ontologies and shapes representation. It deals with the Standard for the Exchange of Product data (STEP) [14] similar to [12], presenting a taxonomy of terminologies included in the STEP standard. Grüninger and Delaval [15], proposed a set of ontologies related with shapes, shape cutting and cutting process. Although this work is not related with direct feature extraction from a standard CAD file, his proposal aimed to deal with the lack of shareability and reusability related with the ruled based feature extraction approachs. He proposed his ontologies using First Order Logic to make a mathematical generalization and verification.

III. The Drawing Exchange Format (DXF)

The Drawing Exchange Format is a *de facto* standard for the interchange of CAD data, which facilitates reading a CAD design previously deployed using software tools as [1]. A detailed explanation of this standard can be found in the DXF Reference Manual [16]. This standard defines geometric primitives as entities such as LINE, CIRCLE, ARC and ELLIPSE. For them, a group of codes is specified indicating what type of data value or feature follows. Besides, from a DXF file is possible to extract descriptions of text, surfaces, color, texture, but the information about solids is not accessible [17], that limits the exchange of data and information with other CAD applications. Nevertheless this work is intended to identify primitive as LINE, CIRCLE, ARC and ELLIPSE stored in a DXF formatted file to populate an ontology and store it as an OWL file. In Table I a description of codes for the primitive LINE is shown, there can be seen that this "entity" is identified with AcDbLine, after that, features are presented, start point and end point values in X, Y, and Z axis are given for this LINE. This definition is similarly described by the DXF specification for CIRCLE, ARC and ELLIPSE, but considering their geometric features.

Table I
DESCRIPTION OF AN ENTITY AS IT APPEARS IN A DXF FILE

DXF file code	Meaning
100	Sub Class
AcDbLine	Name of entity
10	Start point in X
20.83	Value of start point in X
20	Start point in Y
49.27	Value of start point in Y
30	Start point in Z
0.0	Value of start point in Z
11	End point in X
115.44	Value of end point in X
21	End point in Y
91.06	Value of end point in Y
31	End point in Z
0.0	Value of end point in Z
0	End of entity
ENDSEC	End of sequence

IV. Ontologies as Representation of Standards

Ontologies are defined as a specification of a share conceptualization [18]. An ontology includes concepts and relations, it has to be general enough to represent the sharable knowledge in an specific domain. For CAD systems, a domain ontology should represent the common elements of the most accepted and used CAD standard formats. Based on the quantity of scientific and technical references and CAD software tools that we have reviewed until now, we consider that those formats are DXF, IGES and STEP. So, our proposed ontology consist of a group of geometry

 ISBN: 978-1-61208-104-5

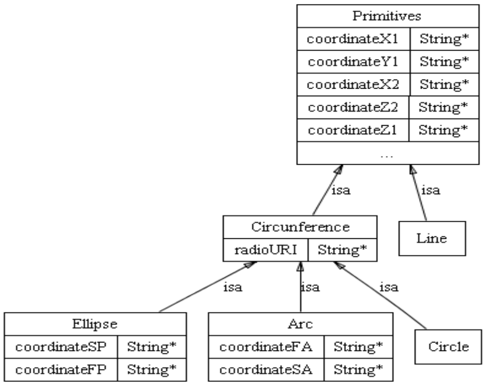

Figure 2. Ontology representation for CAD exchange


```
<Arc rdf:ID="Arc_7">
  <coordinateY1
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >364.7587689599982</coordinateY1>
  <radioURI rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >100.0</radioURI>
  <coordinateZ1
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >0.0</coordinateZ1>
  <coordinateSA
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >90.0</coordinateSA>
  <coordinateX1
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >156.199984283925</coordinateX1>
  <coordinateFA
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >180.0</coordinateFA>
</Arc>
```


Figure 3. Part of a DXF file represented in OWL

primitives defined as classes and its features (Data type properties), which are shown in Figure 2. These classes were defined in a Java Program using the API of Protégé 3.4.4.

After declaring that ontology, the class scanner of Java was used to read each line of a DXF file getting each one of the specific primitives and store them as instance of the ontology in their specific class, the features of the primitives are identified using the code defined for the instance on the DXF specification and stored in the ontology (model). After reading the whole DXF file, an OWL file is generated. On Figure 3, a resulting OWL file of a CAD model can be seen, this partial view shows an instance Arc_7 with its features (Datatype properties)

V. Implementation and Prototyping

Our first implementation, a DXF OWL API importer, was tested with two and three dimensional CAD models, and the extraction of primitives was successful, including all features of each instance in the files. But, as the DXF file belongs to a graphical representation, after we had the OWL Model it was decided to develop a second API in order to make possible the visualization of CAD - OWL models in Protégé as it can be seen in a CAD software tool. In Figure 4 a CAD model is presented, which was exchanged to OWL and can be seen in Fig. 5. This CAD viewer is limited to two dimensions, because our objective was not to make another CAD tool, but this viewer facilitates the human work for the verification of the correct exchange of the model.

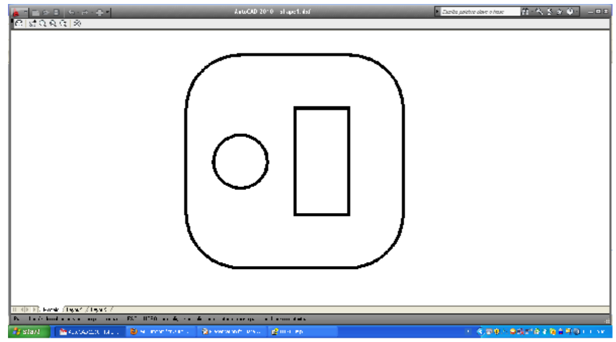

Figure 4. Shape modeled in AutoCAD 2010

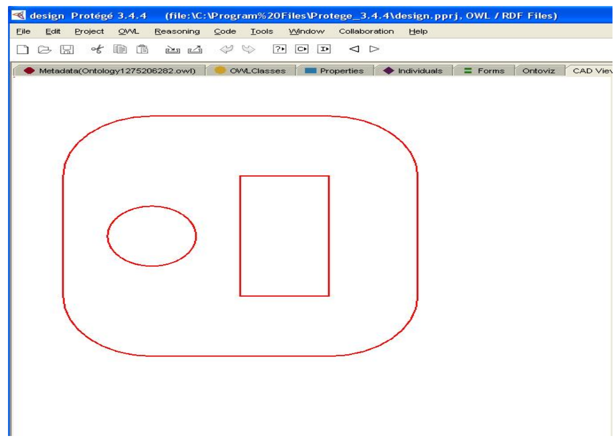

Figure 5. Shape modeled in Protégé

VI. Conclusions and Future work

We have proposed a method to exchange a standard CAD format as DXF into OWL and implement it using the API of Protégé 3.4.4. Following the process describe here, it is possible to get an OWL file from another CAD standards as IGES or STEP, and as a future work we will develop these API's and implement them in a software tool to propose OWL as a CAD data exchanger. These API's will be integrated in an architecture as indicated in Fig 6. Each CAD standard will need two API's, one to exchange from the respective standard to OWL (preprocessor) and another to exchange from OWL to the respective standard

 ISBN: 978-1-61208-104-5

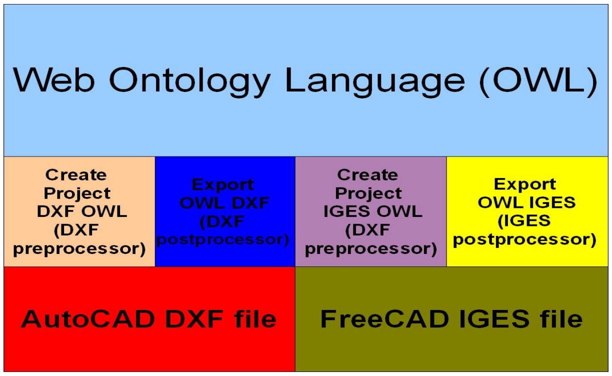

Figure 6. Data Exchanger Architecture

(postprocessor). Other elements of the semantic web as the Semantic Web Rule Language will be included in an improved architecture in order to make complete features extraction and generate self-content designs, which could have intelligent interaction in high automated production processes.

ACKNOWLEDGMENT

*This author wants to thanks to the German Foundation for Academical Interchange (DAAD) and to the Fundación Gran Mariscal de Ayacucho (Fundayacucho) who have granted him his PhD scholarship.

REFERENCES

[1] AutoDESK, *AutoCAD 2010*, On line: http://usa.autodesk.com/adsk/servlet/pc/index?id=13779270&siteID=123112, Last access: July, 2010.

[2] TriStar, *ProEngineer*, On line: http://www.tristar.com/software/proengineer.asp, Last acess: July, 2010.

[3] R. Juergen, *Free CAD*, On line: http://sourceforge.net/projects/free-cad/, Last access: July, 2010.

[4] E, Nasr and A, Kamrani. *Computer Based Design and Manufacturing*. Springer Verlag. New York. 2007.

[5] B. Babic, N. Nesic, and Z. Miljkovic, *A review of automated feature recognition with rule-based pattern recognition*, Computers in Industry, 59(4):321-337,2008.

[6] A. Rappoport, *An architecture for universal CAD data exchange*. In Proceedings of the Eighth ACM Symposium on Solid Modeling and Applications (Seattle, Washington, USA, June 16 - 20, 2003). SM '03. ACM, New York, NY, Pages 266-269, 2003.

[7] I. Cayiroglu, *A new method for machining feature extracting of objects using 2D technical drawings*. Computer Aided Design, 41(12):1008 - 1019, 2009.

[8] M. Henderson and D. Anderson, *Computer recognition and extraction of form features: A CAD/CAM link*, Computers in Industry, 5(4):329-339,1984.

[9] A. de Sam Lazaro, D. Engquist, and D. Edwards, *An Intelligent Design for Manufacturability System for Sheet-metal Parts*, Concurrent Engineering, 1(2):117-123,1993.

[10] National Institute of Standard and Technology. *Initial Graphics Exchange Specification IGES 5.3*, On line: http://www.uspro.org/documents/IGES5-3_forDownload.pdf. Last access: July, 2010.

[11] A. Soman, S. Padhye, and M. I. Campbell, *Toward an automated approach to the design of sheet metal components*. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 17(3):187-204, 2003.

[12] A. Ghafour, P. Ghodous, B. Shariat, and E. Perna, *An Ontology-based Approach for Procedural CAD Models Data Exchange*. In Proceeding of the 2006 Conference on Leading the Web in Concurrent Engineering: Next Generation Concurrent Engineering P. Ghodous, R. Dieng-Kuntz, and G. Loureiro, Eds. Frontiers in Artificial Intelligence and Applications, vol. 143. IOS Press, Amsterdam, The Netherlands, 251-259, 2006.

[13] A. Odd and G. Vasilakis, *Building an Ontology of CAD Model Information*. Geometric Modeling, Numerical Simulation, and Optimization Norway: SINTEF, Pages 11-41, 2007.

[14] International Organization for Standarization. *Industrial automation systems and integration – Product data representation and exchange – Part 1: Overview and fundamental principles*, On line: http://www.iso.org/iso/catalogue_detail?csnumber=18348. Last access: July, 2010.

[15] M. Grüninger, and A. Delaval, *A First-Order Cutting Process Ontology for Sheet Metal Parts*. In Proceeding of the 2009 Conference on Formal ontologies Meet industry R. Ferrario and A. Oltramari, Eds. Frontiers in Artificial Intelligence and Applications, vol. 198. IOS Press, Amsterdam, The Netherlands, 22-33, 2009.

[16] AutoDesk. *DXF Reference*. On line : http://images.autodesk.com/adsk/files/acad_dxf.pdf. Last access: July, 2010.

[17] C. Guk-Heon, M. Duhwan, and H. Soonhung, *Exchange of CAD part models based on the macro-parametric approach*, in INTERNATIONAL JOURNAL OF CAD - CAM. 2(1): 13-21, 2002.

[18] T. Gruber, *A Translation Approach to Portable Ontology Specifications*. Knowledge Acquisition, 5(2):199-220, 1993.

[19] J. Cheng and K. Law, *Using process specification language for project information exchange.* On line: http://eil.stanford.edu/publications/jim_cheng/psl berkeley.pdf. Last access: July, 2010.

[20] R. Schevers and H. Drogemuller, *Converting the industry foundation classes to the web ontology language*. In CSIRO Manuf. and VIC; Inf. Technol., Highett, editors, Semantics, Knowledge and Grid, 2005. SKG '05. First International Conference on, Beijing, Pages 73-83, 2005.

 ISBN: 978-1-61208-104-5