International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:15, No:4, 2021

A Game-Based Product Modelling Environment for Non-Engineer

Guolong Zhong, Venkatesh Chennam Vijay, Ilias Oraifige

Abstract**—**In the last 20 years, Knowledge Based Engineering (KBE) has shown its advantages in product development in different engineering areas such as automation, mechanical, civil and aerospace engineering in terms of digital design automation and cost reduction by automating repetitive design tasks through capturing, integrating, utilising and reusing the existing knowledge required in various aspects of the product design. However, in primary design stages, the descriptive information of a product is discrete and unorganized while knowledge is in various forms instead of pure data. Thus, it is crucial to have an integrated product model which can represent the entire product information and its associated knowledge at the beginning of the product design. One of the shortcomings of the existing product models is a lack of required knowledge representation in various aspects of product design and its mapping to an interoperable schema. To overcome the limitation of the existing product model and methodologies, two key factors are considered. First, the product model must have well-defined classes that can represent the entire product information and its associated knowledge. Second, the product model needs to be represented in an interoperable schema to ensure a steady data exchange between different product modelling platforms and CAD software. This paper introduced a method to provide a general product model as a generative representation of a product, which consists of the geometry information and non-geometry information, through a product modelling framework. The proposed method for capturing the knowledge from the designers through a knowledge file provides a simple and efficient way of collecting and transferring knowledge. Further, the knowledge schema provides a clear view and format on the data that needed to be gathered in order to achieve a unified knowledge exchange between different platforms. This study used a game-based platform to make product modelling environment accessible for non-engineers. Further the paper goes on to test use case based on the proposed game-based product modelling environment to validate the effectiveness among non-engineers.

Keywords**—**Game-based learning, knowledge based engineering, product modelling, design automation.

I. Introduction

THE rapid development of technology has generated higher demands of industrial development capacity, productivity and agile response to market. They all drive the industry to design and produce more complex products at lower cost and with less time. Product design is widely regarded as one of the most important steps in the development of a product. Without a design, there would be no product. The United Kingdom Department of Trade and Industry (DTI) identified that investing money and resources at the design stage yields the biggest return on investment of a product [1]. Product design automation is an impactive differentiator among business competitors as a reduction in time and manpower cost of a product design will allow companies to make flexible business strategies and be price competitive in the market. Companies and industries are able to achieve more sales and high revenues by accelerating the formulation of the new product and offering multiple product variants and options to customers, in the way of reusing and modifying an existing product. However, due to incomplete representation of products and components at the detail design stage, most of the design tasks can only be performed by experienced designers [2]. Modification of a complex CAD model of an engineering product is usually difficult and time-consuming for a new designer. For a non-engineer without enough knowledge of this product, it is not easy to completely understand the propose of each design detail in the existing CAD model or to unravel the complex design references created by other designers before making changes or adding additional geometric features. Thus, at the beginning of the product design, an integrated product model which can represent the entire product information and its associated knowledge will enable designers to work on the design tasks without previous design experience. Time and cost on tutorial sessions and trainings for new designers could be saved. Also, a product model that captures all required design knowledge can perform as a knowledge base for different elements, varying from originality to final product. In this manner, even a complex product, which has features and structures that are special and cannot be modelled or physically produced in straightforward way, can be represented by a comprehensive model.

In primary design stages, the descriptive information of a product is discrete and unorganized while knowledge is in various forms instead of pure data. Capture, store and share the knowledge of experienced designers through a generic product model turns-out to be a challenge. This research focuses on developing a product modelling method that captures and maps the design knowledge into a framework for enhancing the automation of the engineering product design. The proposed framework presents a knowledge capture method in capturing design engineers' knowledge with respect to geometric and non-geometric information. Further explains how the captured knowledge are modelled and mapped into a product modelling framework for enhancing the product design in a knowledge-based environment.

Mr. Guolong Zhong is a PhD student with the Birmingham City University, Millennium Point, Birmingham, B4 7XG, United Kingdom (e-mail: guolong.zhong@bcu.ac.uk).

Dr. Venkatesh Vijay is the lead of Knowledge-based Engineering Lab, Birmingham City University, Millennium Point, Birmingham, B4 7XG, United Kingdom (e-mail: venkatesh.vijay@bcu.ac.uk).

Prof. Ilias Oraifige is the Head of Centre of Engineering, Birmingham City University, Millennium Point, Birmingham, B4 7XG, United Kingdom (e-mail: ilias.oraifige@bcu.ac.uk).

II. Background

A product model is an information representation that provides data contributing to build the form (geometry and topology), function (intent) and behaviour (load resistance, etc.) of a product in a modelling process [3]. It is employed throughout the entire lifecycle of a product to structure product data and design information. To date, engineering in product design uses four main approaches to generate a product model, which are: (i) solid product modelling, (ii) feature-based product modelling, (iii) knowledge-based product modelling and (iv) integrated product modelling [4]. In this paper, integrated product modelling is regarded as part of knowledge-based modelling methodology as it is a functional combination of different modelling methods.

A. Solid Product Modelling

Solid product modelling [5] is a technique that uses mathematical principles and computer modelling to achieve precise representation of three-dimensional objects . It is now a mature tool that is widely implemented in product modelling field [4]. There are two common methods of solid product modelling [9]: boundary representation (B-rep) [6] and constructive solid geometry (CSG) [5]. In B-rep method, the product is divided into a number of faces bounded by edges. In turn, the edges are bounded by two vertices at last. The B-rep method provides a fast display of a product geometry with basic information about the faces, edges and vertices [6]. The CSG method breaks the product into a binary tree of basic solids for example, cylinders, spheres, cones and cubes etc. The product itself in CSG is considered as a combination of those basic solids by utilizing union, difference and intersection operations [4]. Therefore, it can be seen that both B-rep and CSG present a clear and simple data structure of a product. However, [7] pointed out that the weakness of solid product modelling lies in providing all necessary information for an entire product development lifecycle. Solid product modelling works in a different way from a human product designer because it can only create models with basic geometric information such as dimension, tolerance etc. For a complete product lifecycle, more necessary information is still required, for example, "how the product will be manufactured", "what the function of the product is", etc.

B. Feature-Based Modelling

Feature-based product modelling [7] is seen as a well-developed extension of solid product modelling . The "Feature" here is defined by [8] as information sets that refer to aspects of form or other attributes of a part, such that these sets can be used in reasoning about design, performance or manufacture of the part or assemblies they constitute. Chen and Wei [7] mentioned that feature-based product modelling shows great advantages over conventional solid modelling methods, for example, capturing design intents, relating functionality with product geometry and working on high level shapes instead of geometric details, etc. In the past twenty years, much research has been conducted by using features to support product design [9], [10] and assembling process [11]. However, a product from feature-based modelling is not able to transfer knowledge such as expertise and experience to other designers. A product modelling methodology that could capture and reuse the involved knowledge is in demand.

C. Knowledge-Based Product Modelling

The methodology that allows to capture and structure knowledge about a product design and its design process is known as knowledge-based engineering or KBE. Knowledge-based product modelling is characterised by capturing and reusing engineering knowledge such as human expertise, product and process knowledge in modelling process.

In 1989, Lawrence [12] suggested a public recognition of the potential of knowledge-based engineering - "Although it's not yet widely known, knowledge-based engineering is having a profound effect on how a few companies are speeding their products to market". In 1990, Jurit et al. [13] integrated frame-based and rule-based knowledge representation in a feature-based modelling system. It is considered as an early attempt to combine knowledge and expertise into a product modelling system. In the last 20 years, KBE has shown its advantages in product development in different engineering areas such as automation, civil engineering and aerospace engineering in terms of modelling and cost saving [14], [15].

The use of KBE methods and techniques has played an important role in design automation for the development of a product in industry [16], [17]. However, the current design and implementation of the KBE systems are usually platform specific and domain dependent. These kinds of KBE systems are usually developed in a particular programming language (i.e. C, C++, Java) which means the principle design parameters, rules, constraints are all defined and written in code. Thus, a change of a platform will always require an entire re-write of the KBE system. Furthermore, transparency of KBE systems is also necessary in order to avoid black-box problems that happen in the communication between different KBE applications [18], [19]. This requires a framework that can provide adaptable, structured and reusable knowledge bases.

D. STEP Standard

To improve interoperability of the product model among CAD software and to avoid the data conversion faults, such as data missing and mismatching, an international standard needs to be followed. STEP is the international standard for exchange of product data. It addresses product data from mechanical and electrical design, geometric dimensions and tolerances, analysis and manufacturing, as well as additional information specific to various industries such as automotive, aerospace, building construction, ship, oil and gas, process plants and others. Because of the complexity, STEP is divided into smaller component parts, including a series of 'Application Protocols' or APs, each covering a particular industrial domain. And each APs is titled by the domain that it applies to [20]. The research of Yang et al. [4] shows the potential of using STEP as an interoperable data standard to exchange and share product data in different engineering environment. Although the readability of the STEP data model is low due to the tedious STEP definition, modelling product with STEP can provide a standardized, neutral data exchange between CAD systems.

III. Challenges

In a traditional product modelling environment, the designer is able to create and provide the visible geometry information in the product model. However, only geometry information is not enough to describe a product model completely. The utilization of the existing associated knowledge of a product can greatly reduce the unnecessary re-analysis, re-design, and re-planning, simplify the modelling tasks, and ensure the modelling quality. Having said that, capturing and transferring the knowledge of experienced designers are difficult. According to [21], the main challenge that exists in small and medium companies and industries for implementing knowledge capture initiatives is lack of awareness of knowledge capture benefits. Because of this, individuals, small and medium enterprises are lack of vision and strategy as well as structure for knowledge capture. They have strong reliance on informal networks and collaboration to locate the repository of knowledge. Therefore, one important issue of knowledge-based product modelling is to classify, structure and manage the captured knowledge. Since there is no clear formalised link between a generic product model and an interoperable format in KBE environment, it is essential to provide well-defined knowledge classes and a formalized knowledge capture method for individuals, enterprises and industries to capture and share knowledge instead of using informal oral communication or notes and spreadsheets in different formats. In this paper, geometric data contained in a CAD file are regarded as "geometry information" and non-geometry information, such as experience, expertise and design rules, is called "knowledge".

This research aims to provide a knowledge-based product modelling framework and method that could capture and integrate the geometry information and the associated knowledge applied by designers, into a generative product model. To achieve this target, two key factors are considered. Firstly, the generated product model must have well-defined classes that can represent the entire product information and its associated knowledge. Then, the captured knowledge in the product model needs to be represented in an interoperable schema to ensure a steady data exchange between different product modelling platforms and CAD software. The rest of the paper will discuss in detail on how the knowledge from the design engineer is captured and mapped into the framework for automation.

IV. A Knowledge-Based Product Modelling Framework

A. Conceptual Framework

A game-based product modelling environment does not mean turning the product modelling process into games. It means applying KBE methodology in a product modelling environment with the help of a game-based platform. To develop such a product modelling environment, a knowledge-based product modelling framework (Fig. 1) has been proposed in this research.

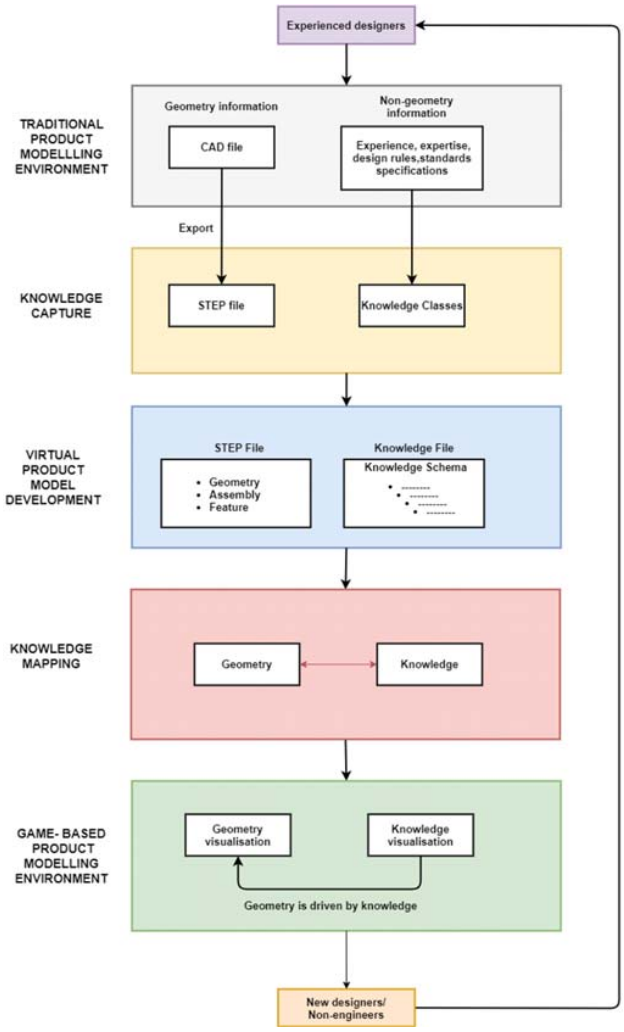


Fig. 1 Knowledge-based product modelling framework

This framework consists of four major blocks: knowledge capture, virtual product model development, knowledge mapping and product visualization and validation. The knowledge capture block provides an environment where the non-geometry knowledge that applied by experienced designers in the product modelling process is captured. The geometry information of the product can be captured by exporting the CAD model into an interoperable format - STEP.

After performing knowledge capture, a knowledge file that contains well decomposed non-geometric classes and properties is generated, and a geometry file with the geometric data that describes the product's geometry is also acquired. A virtual product model will be formed by utilising the knowledge that captured in the previous stage. This virtual product model is a comprehensive representation of the product with all its essential non-geometric and geometric information. Knowledge Mapping is the most significant stage of the framework, which builds the connection and interaction between the non-geometric information and geometry. In this stage, non-geometric information is transferred from the knowledge file to object-oriented programming environment, where key parameters and design rules are linked to the product geometry. The data from the knowledge capture tool are exported into a knowledge file in an XML format and parsed using the object-oriented programming. The last stage is the visualization and validation stage, where the developed virtual product model is visualised and checked for the correct representation of the initial model. Testing use cases is performed for identifying and evaluating the effectiveness of the proposed framework. If design engineer changes one dimension of the geometry of the initial product model, the virtual product model checks the rules that determine this geometry and propagates the rules and changes to the design engineer. In this case, design engineer can check what will be affected if geometry is to be changed in this model and the constraints of these changes.

B. Virtual Product Model

Variance of product models always results in extra work in transferring data from different models. Therefore, in order to make the representation as robust as possible without having to predefine attributes that might be relevant only in a given domain, it is necessary to create a generic and platform independent product model that can be used in all product design stages and different platforms.

In this research, a Virtual Product Model (VPM) is developed as a generative representation of a product to enhance the knowledge representation of the product model but maintain generic. The meta class sets (Fig. 1) defined for the VPM are derived from literature review analysis, previous related research in product modelling [22]-[29], own experience of product modelling in industry and expert's knowledge. Fig. 2 shows the how a product is represented with knowledge classes in VPM.

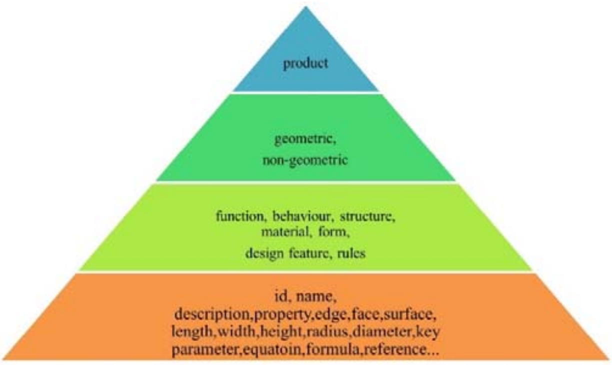


Fig. 1 VPM meta knowledge classes

When the VPM is transferred between different platforms, the geometry information will be stored and transmitted through the STEP file and the knowledge will be stored and transmitted through an XML document. To ensure the data structure of the XML document, an XML schema has been developed based on the framework of VPM. This XML document is named as Knowledge File (KF) in this research and will be used as an independent format for data exchange.

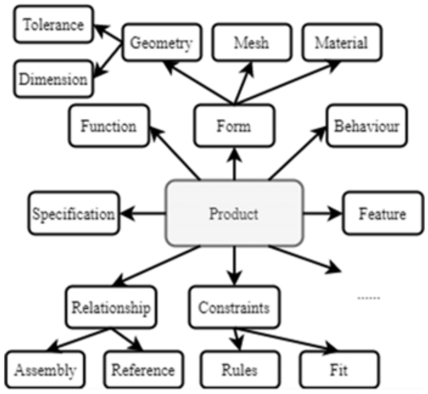
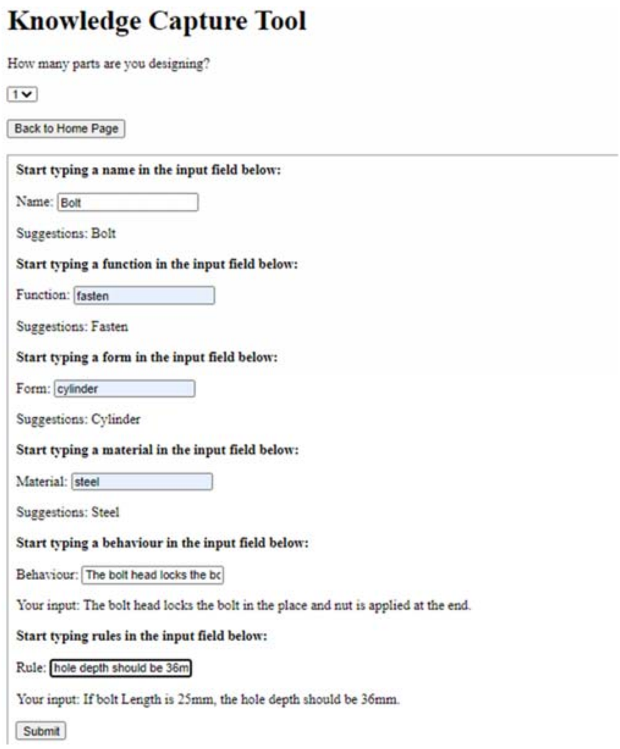
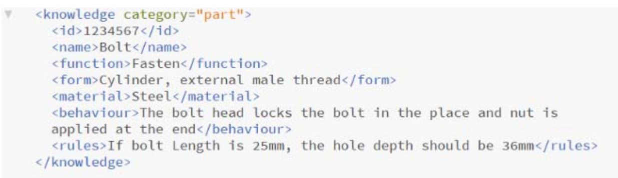


Fig. 2 A structure view of the product represented by the VPM atomic blocks



(a)



(b)

Fig. 3 (a) Knowledge capture tool interface (b) knowledge file generated from the knowledge capture tool

International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:15, No:4, 2021

The developed framework was applied into a game-based product modelling prototype system which was developed in Unity 3D software. Unity 3D has a license-free version for personal development which includes an environment for the development of interactive 2D and 3D content including a rendering and physics engine and a scripting interface to program interactive content [30]. It allows the developers to export their applications into all the mainstream operation systems (Windows, Mac, Linux, IOS, Android) through multiple platforms (desktop, web, mobile). Compared with traditional CAD software which always requires training and practice to get started, this game-based environment is more accessible for non-engineers to use.

V. Use Case

Bolts and nuts are the most basic components in mechanical design. This study uses the modelling of a bolt to validate the proposed method and framework. This use case shows how the non-geometry knowledge can be captured in the knowledge capture environment and then visualised in the game-based product modelling environment.

In this research, a knowledge capture tool (Fig. 3) has been developed that allow designers to input the non-geometric information that are utilized in the product modelling process. Those non-geometric information are then decomposed into the VPM classes. When a bolt is being modelled in the traditional CAD software, the knowledge capture tool asks the designers to provide the associated knowledge at the same time. The captured knowledge is structured by the classified knowledge classes (example shown in Table I) and stored into a Knowledge File which is generated at the end of the knowledge capturing process (Fig. 3 (b)).

TABLE I
Knowledge Capture Template with Captured Knowledge of a Product (Modelling a Bolt)

Knowledge class	Captured knowledge
Name	Bolt
Category	Part
Function	Fasten
Form	Cylinder, external male thread
Behaviour	The bolt head locks the bolt in the place and nut is applied at the end.
Material	Steel
Rules	If bolt Length is 25mm, the hole depth should be 36mm.

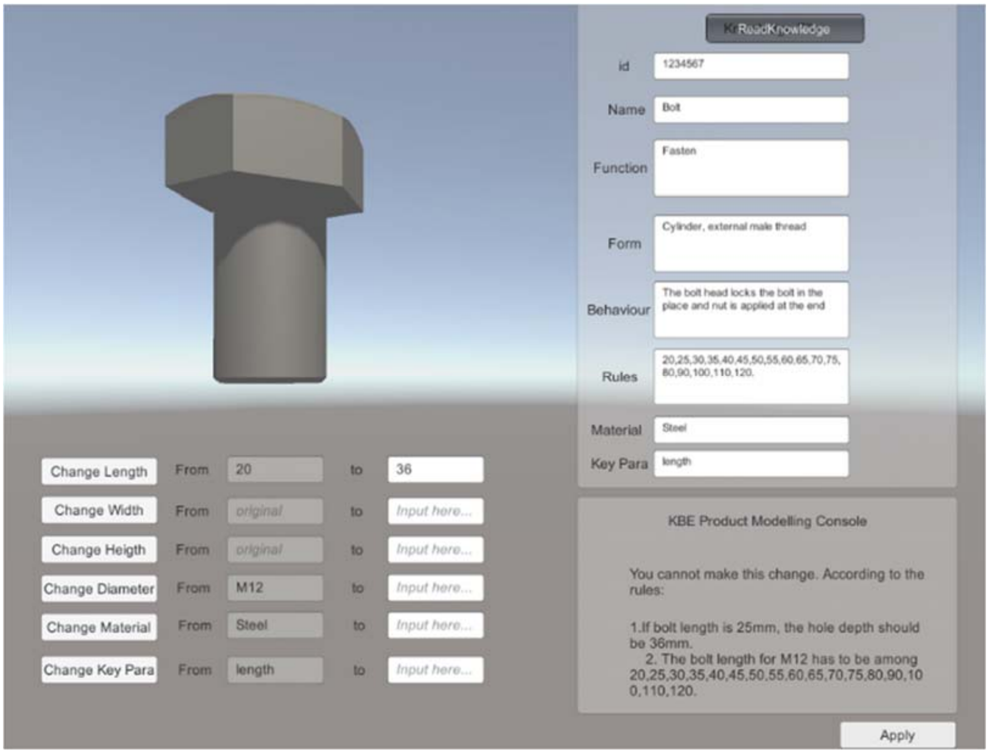
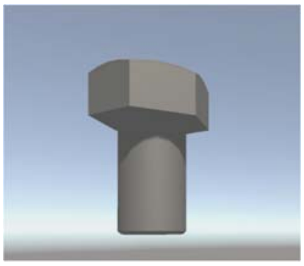


Fig. 4 The game-based product modelling environment developed in Unity3D

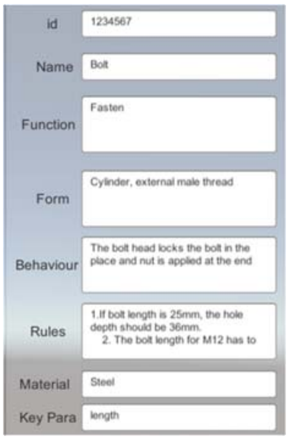
The developed game-based product modelling environment (Fig. 4) provides the ability to visualise the bolt's original geometry, as well as the possible changes that will happen and the associate knowledge the constrains the changes. The first type of visualisation is shown in the tool by importing the geometry file of the product into the interface through Unity3D plugin (Fig. 5 (a)). It provides the 3D view of the geometry of the product model. The second type of visualisation is the text representation of the all the non-geometric information that stored in the knowledge file (Fig. 5 (b)). This process is performed with the self-written code that automatically analyses and displays all the acquired non-geometric information that is stored in the knowledge file. The third type of visualisation is the possible change of geometry (Fig. 6 (a)). However, due to the natural of STEP, there are limited tools that support editing the geometry data in the STEP file and displaying the graphical changes directly in the modelling environment (tested implementation tools shown in Table II). To develop such a method that can manipulate and visualise STEP files is beyond the scope of this research. Instead of visualising the graphical changes in the geometry, this research uses text representation to indicate the changes that are made to the geometry.

TABLE II
TESTED IMPLEMENTATION TOOLS (TOOL VERSION UP TO 2020)

Tools	STEP Edit	Geometry View	User Interface Development
SolidWorks	x	✓	x
CATIA	x	✓	x
Siemens NX	x	✓	x
Autodesk Inventor	x	✓	x
Unity3D	x	✓(Plugin)	✓
Creo	x	✓	x
CAD Exchanger	x	✓	x
STP viewer	x	✓	x
IDA-STEP	x	✓	x
CAD Assistant	x	✓	x
Free CAD	x	✓	x



(a)



(b)

Fig. 5 (a) 3D view of the bolt geometry (b) visualisation of all the non-geometric information stored in the bolt knowledge file

(a)

(b)

Fig. 6 (a) Functions of making possible changes of bolt geometry (b) visualisation of the associated knowledge that are applied to constrain the change of geometry

```
#132,#133,#134,#135,#136,#137,#138,#139));
#95=TOROIDAL_SURFACE('',#372,2.613,0.2);
#96=CONICAL_SURFACE('',#360,4.,60.0000000000001);
#97=CONICAL_SURFACE('',#362,4.,60.0000000000001);
#98=CONICAL_SURFACE('',#364,4.,60.0000000000001);
#99=CONICAL_SURFACE('',#366,4.,60.0000000000001);
#100=CONICAL_SURFACE('',#368,4.,60.0000000000001);
#101=CONICAL_SURFACE('',#370,4.,60.0000000000001);
#102=CONICAL_SURFACE('',#373,2.1717,45.);
#103=CYLINDRICAL_SURFACE('',#356,2.413);
#104=FACE_BOUND('',#169,.T.);
#105=FACE_BOUND('',#170,.T.);
#106=FACE_BOUND('',#171,.T.);
#107=FACE_BOUND('',#172,.T.);
#108=FACE_BOUND('',#181,.T.);
#109=FACE_BOUND('',#182,.T.);
#110=FACE_BOUND('',#183,.T.);
#111=FACE_BOUND('',#184,.T.);
#112=CIRCLE('',#352,2.613);
#113=CIRCLE('',#354,2.413);
#114=CIRCLE('',#355,2.413);
#115=CIRCLE('',#357,2.1717);
#116=CIRCLE('',#359,4.);
#117=CIRCLE('',#361,4.);
#118=CIRCLE('',#363,4.);
#119=CIRCLE('',#365,4.);
#120=CIRCLE('',#367,4.);
#121=CIRCLE('',#369,4.);
#122=ADVANCED_FACE('',(#149),#140,.T.);
#123=ADVANCED_FACE('',(#150),#141,.T.);
#124=ADVANCED_FACE('',(#151),#142,.T.);
```

Fig. 7 Part of the STEP file of a bolt model that exported from NX10.0

The final stage of visualisation is to show the associated knowledge that are applied to make or constrain the change of geometry (Fig. 6 (b)). In this use case, the STEP file (example shown in Fig. 7) of the bolt model that exported from the CAD software and the Knowledge File (Fig. 3 (b)) from the knowledge capturing environment are loaded into game-based product modelling environment. When the designer is applying the changes of the length of the bolt, the environment will indicate the acceptable the length values according to the captured rules from the experienced designers (Fig. 6). In this way, new designers are guided by the knowledge to perform modelling activities.

VI. Discussion

The method proposed in this paper is aimed at providing a generative product model as a generative representation of a product, which consists of the geometry information and non-geometry information, through a product modelling framework. The presented method for capturing the knowledge from the designers through a knowledge file provides a simple and efficient way in collecting and transferring knowledge. Further, the knowledge schema provides a clear view and format on the data that needed to be gathered in order to achieve a unified knowledge exchange between different platforms.

One of the advantages of the VPM is the explicit classification of knowledge that required to in product design. The geometry information of a product is stored in VPM through a STEP file which has already been an interoperable format among CAD software. Hence, the proposed VPM can provide a generic product representation which enables seamless and cross-platform model communication.

A few limitations of the proposed method are discussed next. First, the designers need to spend extra time to use Knowledge Capture Tool, compared with modelling in traditional environment. This mainly happens at the first time when designers start to model without having a knowledge file. However, it is necessary to spend such time in order to capture the knowledge of designers and to build the knowledge repository. Secondly, functionality of modelling geometry in the developed game-based product modelling environment is limited. Due to the natural of STEP, there are no existing technologies that support editing the geometry data in the STEP file and displaying the graphical changes directly in the modelling environment. In this game-based product modelling environment, the change of geometry mainly relies on the text description and knowledge indication. But this will not restrict the capability of the proposed modelling method. New standards and advanced geometry editing tools will accelerate the formation of a well-developed product modelling environment in the future.

VII. Conclusion

Design automation is a crucial demand from the industry to achieve a reduction in time and manpower cost of a product design. Capture, store, share and reuse the knowledge of experienced designers through a generic product model turns-out to be a challenge in product modelling. In order to provide non-engineers a product modelling environment where the existing knowledge can be used to guide and support their product modelling, a generative product model that integrates the geometry information witch the associated knowledge that applied by experienced designers is required. The product model must have well-defined classes that can represent the entire product information and its associated knowledge. Also, the product model needs to be represented in an interoperable schema to ensure a steady data exchange between different product modelling platforms. This paper has presented a VPM which consists of geometry and non-geometry information of a product. Further, this work presented a knowledge-based product modelling framework and explained how that captured data from a use case are modelled and visualised for enhancing the re-use of knowledge in product modelling. The future work includes improvement of the knowledge capture environment and further development of geometry editing functionality in the knowledge-based product modelling environment. As a continuation of the use case presented in this paper, a more complex product model will be tested in enhancing the performance of the proposed framework.

References

[1] Y. Haik and T. Shahin, Engineering Design Process - second edition, no. December. 2011.

[2] C. X. Feng, C. C. Huang, A. Kusiak, and P. G. Li, “Representation of functions and features in detail design,” CAD Computer Aided Design, vol. 28, no. 12. pp. 961–971, 1996, doi: 10.1016/0010-4485(96)00027-9.

[3] F. P. Tolman, “Product modeling standards for the building and construction industry: past, present and future,” Autom. Constr., vol. 8, no. 3, pp. 227–235, 1999, doi: 10.1016/S0926-5805(98)00073-9.

[4] W. Z. Yang, S. Q. Xie, Q. S. Ai, and Z. D. Zhou, “Recent development on product modelling: a review,” Int. J. Prod. Res., vol. 46, no. 21, pp. 6055–6085, 2008, doi: 10.1080/00207540701343895.

[5] V. Shapiro, “Solid Modeling,” Handb. Comput. aided Geom. Des., vol. 20, pp. 473–518, 2002, doi: DOI: 10.1016/B978-044451104-1/50021-6.

[6] I. Stroud, Boundary Representation Modelling Techniques, 1st ed. Springer-Verlag London, 2006.

[7] Y.-M. Chen and C.-L. Wei, “Computer-aided feature-based design for net shape manufacturing,” Comput. Integr. Manuf. Syst., vol. 10, no. 2, pp. 147–164, 1997, doi: 10.1016/S0951-5240(97)00006-2.

[8] O. W. Salomons, F. J. A. M. van Houten, and H. J. J. Kals, “Review of research in feature-based design,” J. Manuf. Syst., vol. 12, no. 2, pp. 113–132, 1993, doi: 10.1016/0278-6125(93)90012-I.

[9] Michael J. Pratt, “Synthesis of an optimal approach to form feature modelling,” in Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition, 1988, vol. 1, pp. 263–274.

[10] L. Wingård, “Introducing form features in product models: a step towards CADCAM with engineering terminology,” PhD Dissertation, Computer System for Design and Development, 1991.

[11] W. Van Holland and W. F. Bronsvoort, “Assembly features in modeling and planning,” Robot. Comput. Integr. Manuf., vol. 16, no. 4, pp. 277–294, 2000, doi: 10.1016/S0736-5845(00)00014-4.

[12] W. Lawrence, “Using Knowledge-Based Engineering,” Production, p. 74, 1989.

[13] A. Jurit H., Saia, A. and De Pennington, “Reasoning about machining operations using feature-based models,” Int. J. Prod. Res., vol. 28, pp. 153–171, 1990.

[14] L. W. Rosenfeld, “Solid modeling and knowledge-based engineering,” in Handbook of solid modeling, McGraw-Hill, Inc. New York, USA, 1995, pp. 91–911.

[15] E. J. Reddy, C. N. V. Sridhar, and V. P. Rangadu, “Knowledge Based Engineering: Notion, Approaches and Future Trends,” Am. J. Intell. Syst., vol. 5, no. 1, pp. 1–17, 2015, doi: 10.5923/j.ajis.20150501.01.

[16] I. O. Sanya and E. M. Shehab, “An ontology framework for developing platform- independent knowledge-based engineering systems in the aerospace industry,” Int. J. Prod. Res., vol. 53, pp. 1–27, 2014, doi: 10.1080/00207543.2014.965352.

[17] E. M. Shehab and H. S. Abdalla, “Manufacturing cost modelling for concurrent product development,” Robot. Comput. Integr. Manuf., vol. 17, no. 4, pp. 341–353, 2001, doi: 10.1016/S0736-5845(01)00009-6.

[18] I.-S. Fan and P. Bermell-Garcia, “International Standard Development for Knowledge Based Engineering Services for Product Lifecycle Management,” Concurr. Eng., vol. 16, no. 4, pp. 271–277, 2008, doi: 10.1177/1063293X08100027.

[19] M. Cederfeldt, F. Elgh, and I. Rask, “A Transparent Design System for Iterative Product Development,” J. Comput. Inf. Sci. Eng., vol. 6, pp. 300–307, 2006.

[20] A. B. F. Kc Morris, STEP, the grand experience. Gaithersburg: National Institute of Standards and Technology, 1999.

[21] S. Suresh, C. Egbu, and B. Kumar, “Key issues for implementing knowledge capture initiatives in small and medium enterprises in the UK construction industry,” no. January, 2006.

[22] F. A. Salustri, “A formal theory for knowledge-based product model representation,” Manuf. Syst., no. 519, pp. 1–19, 1996.
[23] S. J. Fenves, “A core product model for representing design information,” Tech. Rep. No. NISTIR 6736, Natl. Inst. Stand. Technol., 2001, [Online]. Available: http://www.mel.nist.gov/msidlibrary/doc/ir6736.pdf.
[24] F. Wang et al., “Towards modeling the evolution of product families,” ASME Comput. Inf. Eng. Conf., 2003.
[25] K. W. L. Mehmet Murat Baysal, Utpal Roy, Rachuri Sudarasan, Ram D. Sriram, “Product information exchange using Open assembly model: issues related to representation of geometric information,” 2005.
[26] Z. Lou, H. Jiang, and X. Ruan, “Development of an integrated knowledge-based system for mold-base design,” J. Mater. Process. Technol., vol. 150, no. 1–2, pp. 194–199, 2004, doi: 10.1016/j.jmatprotec.2004.01.037.
[27] G. La Rocca, L. Krakers, and M. J. L. van Tooren, “Development of an ICAD Generative Model for Blended Wing Body Aircraft Design,” in 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 4-6 September 2002, Atlanta, Georgia, 2002, no. September, pp. 1–10, doi: 10.2514/6.2002-5447.
[28] J. Groß and S. Rudolph, “Generating simulation models from UML - A FireSat example,” in Proceedings of the 2012 Symposium on Theory of Modeling and Simulation - DEVS Integrative M&S Symposium, 2012, pp. 1–8.
[29] J. Gross, A. Reichwein, S. Rudolph, D. Bock, and R. Laufer, “An Executable Unified Product Model Based on UML to Support Satellite Design,” in Proceedings of the AIAA SPACE Conference, 2009, no. September, doi: 10.2514/6.2009-6642.
[30] Unity Technologies, “Unity User Manual,” Unity Documentation, 2020. https://docs.unity3d.com/Manual/index.html (accessed Jun. 18, 2020).