PRODUCT REPRESENTATION IN LIGHTWEIGHT FORMATS FOR PRODUCT LIFECYCLE MANAGEMENT (PLM)

Lian Ding
University of Bath
l.ding@bath.ac.uk

Alex Ball
UKOLN
a.ball@ukoln.ac.uk

Jason Matthews
University of Bath
j.matthews2@bath.ac.uk

Chris McMahon
University of Bath
enscam@bath.ac.uk

Manjula Patel
UKOLN
m.patel@ukoln.ac.uk

ABSTRACT

Currently, companies face the unprecedented challenges of the global marketplace, collaborative environments and the entire product lifecycle. There are new requirements for product representations, including: platform/application independence, support for the product lifecycle, rapidly sharing information between geographically distributed applications and users, and protection of commercial security. To meet these, some lightweight representations have been developed and applied in different industries. This paper highlights some limitations in the current applications, and presents a framework of lightweight representations for the product lifecycle. In the proposed framework, a markup method is applied to the whole product lifecycle. The approach is demonstrated with an industrial case study.

KEYWORDS

Lightweight representation, markup method, Representation Information Registry/Repository, PLM, CAD model

1. INTRODUCTION

For companies actively developing new products, the most important information and knowledge issues are stored in the product representation. With the current trends of global competition, companies now need to consider the entire product lifecycle and expand product developments, to support the global market place. This is likely to include collaboration across companies based in different geographic locations. To support such a scenario, and to strengthen product information management for the total product lifecycle, enhanced requirements are needed for product representations. Conventional product representations, e.g., geometric or feature-based Computer-Aided Design (CAD) models, are facing significant challenges in this respect:

- There is a vast array of commercial CAD software systems, each of which has its own proprietary format. Obviously, it is not a feasible or economically viable solution for every user to install a copy of each CAD system to view or manipulate product models in its representations.
- Current CAD models are too 'heavy', and restrict information transmission between geographically distributed applications and users.
- It is difficult for long-term users to retrieve information, due to the ephemeral nature of CAD file formats and the applications that work with them.
- Product models are among the most important intellectual property of a company. In a collaborative situation, companies need to work with each other but they are often unwilling to share all the details of their product models directly, to avoid leaking their commercially sensitive information to their competitors.

A new-generation product representation is needed to rise to these challenges. Recently, lightweight representations have attracted researchers' attention, for their particular characteristics. This paper gives a survey of the lightweight representations employed in product lifecycle management (PLM). Then, to address the problems of current lightweight representations, a new framework of lightweight representation for the entire product lifecycle is proposed. The framework integrates a markup method and the Representation Information Registry/Repository constructed by the UK Digital Curation Centre (DCC). Some experimental results with lightweight representations using the JT format and PLM XML are given to demonstrate the feasibility of the proposed method. Finally, conclusions and suggestions for further work are given.

2. LIGHTWEIGHT REPRESENTATIONS IN PLM

The aim of lightweight representations is to support users at different stages of the product lifecycle in rapidly browsing, retrieving and manipulating product information. Recently, it is an area that has attracted much attention from researchers. Currently, the main efforts focus on two aspects: 1) developing compression methods for the reduction of file sizes, such as domain-specific compression, various 3D graphics compaction algorithms, simplified visualisation methods, and reference mechanisms; and 2) integrating markup languages for cross-platform support. Some lightweight representations have been developed into commercial systems and have been successfully used in some parts of PLM. In addition, research is actively being carried out on desiccated formats [Kunze, 2005, UC Libraries] for long-term preservation and representation compatibility. The basic idea of a desiccated format is to "dry up" the representation of the original information by eliminating some special format features (e.g. fonts, graphics and colours). As the application of the concept of desiccated formats to the engineering domain is still not clear, it is not explored further in this paper. The principal lightweight representations for CAD are listed below:

- Universal 3D (U3D) [U3D: Information From Answers.com]

U3D has been developed by Intel and the 3D Industry Forum (3DIF), which aims to develop a universal standard for various 3D data for inter-exchange. To reduce the file size for quick Internet downloading and fast rendering on screen, most of the engineering data associated with the original model is eliminated. U3D describes a 3D object through a series of nodes and resources. The resources contain the majority of the information required to create an object and are referenced by nodes. Multiple nodes may use the same resource, therefore further reducing the U3D file size.

The specification for U3D, the 3rd edition of Ecma International's ECMA-363 *Universal 3D File Format* Standard [Ecma International, 2006] and the corresponding implementation software [Sourceforge.net] has been released as open source. In addition, U3D is supported from Version 7 of Adobe Acrobat (PDF Version 1.6). With the 3D tool in Adobe Acrobat 7.0, a U3D file can be embedded in a PDF document, in which its presentation properties, 3D canvas, and new views can be edited, adjusted and created.

The U3D format allows users to export high-quality on-screen visualisations of 3D CAD models, but without large source files. As U3D adopts a domain specific compression algorithm, it is particularly useful for industries with complex products, such as the automobile industry, for sales, marketing, customer support, online promotions, maintenance, training, and many other workflow areas. Recently, Adobe has announced that Adobe Acrobat 3D software has been adopted by Renault Group to extend its 3D visualisation and design collaboration capabilities across its extended enterprise of employees and supply chain partners [Dexigner, 2007].

- X3D

X3D is developed by the Web3D Consortium [Web3D Consortium] as a major upgrade from VRML (the Virtual Reality Modelling Language). The basic unit of the X3D run-time environment is a scene graph. The scene graph consists of several nodes in a hierarchical structure, with shape nodes representing the objects themselves, and transform nodes describing the spatial positions of the objects. X3D adopts multiple compression algorithms, including domain-specific, type-specific fields and Fast InfoSet, and therefore it is more lightweight than VRML. In addition, the nodes in X3D are represented using the eXtensible Markup Language (XML) so as to take full advantage of the potential of XML on the Internet. X3D Tools and Applications are provided on the website [Web3D Consortium], including various X3D viewers, browsers, plug-ins, developer toolkits and libraries. For examples, Xj3D is an open source X3D toolkit and X3D browser written completely in Java [The Xj3D Project]. X3D has become a free, ISO-ratified format to provide long-term stability for Web3D content and applications.

Using X3D, high-quality 2D, 3D and video information can be easily incorporated into technical publishing, maintenance manuals, websites, mashups, database applications, visual simulations, navigation systems and many other professional and consumer uses [Web3D Consortium]. Currently, X3D's applications cover many different industries, from engineering, healthcare and oil exploration to astronomy.

- 3D XML

3D XML is a lightweight and standard XML-based format that enables users to access and share accurate 3D data quickly and easily [Versprille, 2005]. To improve the speed of sharing 3D product data, a sophisticated 3D graphics compaction algorithm based on NURBS surface mathematics has been developed. The algorithm approximates a portion of a CAD model's surface area using a single surface patch instead of a tessellated form that could consist of hundreds or thousands of tessellated triangles. A reference/instance mechanism is adopted by 3D XML, in which a product structure consists of the 3D reference or standardised objects that can be reused in one or more products by instantiation [Dassault Systèmes, 2006]. As the object geometry is defined only once in a reference, the reference/instance mechanism minimises data duplication and reduces the size of 3D XML documents. 3D XML uses open XML schemas to communicate product geometry, product structure, and graphical display properties. Thus, it can be read, written and enriched by standard tools; and allows users to add extensions based on their own specific requirements.

3D XML is able to transmit and exchange complicated 3D graphics rapidly allowing companies to collaborate with their suppliers, partners and customers more effectively. Currently, Dassault Systemes' latest release of their CAD/PLM software suite, V5R15, has embedded 3D XML throughout so that it can be used in Dassault's product developers' tools worldwide, such as CATIA V5, DELMIA, ENOVIA and DMU V5. Virtools Dev 3.5 [Virtools], a software tool to create real-time 3D applications with complex interactivity, also adopts 3D XML. The Virtools 3D XML plug-in allows users to configure import settings of 3D XML files, including textures, lights, viewpoints/cameras, materials, meshes and scale; and to optimise the geometry details of the models, including Exact Geometry, Tessellated Geometry and Compressed Tessellated Geometry options. 3D XML Player [Business Wire, 2005] extends 3D XML beyond traditional PLM applications, offering integration with Microsoft Office applications and the Internet Explorer browser, or working as a standalone application. In addition, 3D XML has been supported by organizations such as the Toyota Motor Corporation.

- JT Format

JT is a 3D product visualisation data format, which was originally developed by Engineering Animation, Inc., and is now the product of UGS Inc. Differing from the above formats, JT format consists of a combination of facets and B-Rep (Boundary Representation) geometry along with Product and Manufacturing Information (PMI) and textual attributes [Wiki: JT]. JT files adopt two compression methods: standard and advanced. The standard method uses a simple, lossless compression algorithm, and the typical compression ratios average is about 2:1 over non-compressed JT files. In contrast, the advanced method applies a more sophisticated, domain-specific compression scheme to support lossless geometry compression. The typical compression ratio average is about 2.5:1 over JT files that have been compressed by the standard method. Thus, with the combination of standard and advanced compression methods, the smallest JT file size can be achieved.

At present, the JT format is supported by JT Open, which is an organization including software vendors, users, and interested parties in the PLM industry that have adopted the JT format. JT Open Toolkit is a C++ library, which is able to create JT formatted data and read and access JT data on various hardware and operating systems, such as Windows, SUN, HP, SGI and AIX. JT2Go [UGS] enables users to view JT files and embed 3D JT data in Microsoft Office documents. In addition, JT files can be created by translating data from all major MCAD applications, such as UGS NX, SolidEdge, Catia and Autodesk Inventor. The JT format is a mature lightweight data format, and can be used as a predominant, lightweight visualisation format for PLM, a CAD-neutral exchanging format for collaborative product development, or a consistent 3D visualization up and down the supply chain. At present, the JT format is being widely used in the automobile, aerospace and various manufacturing industries, such as Siemens [UGS: Newsroom, 2005].

- PLM XML

PLM XML [PLM Component - UGS, 2005] is an XML-based PLM format created and supported by UGS. The objective of PLM XML is to integrate collaborative product lifecycle processes by offering a standardised protocol for data interoperability. Thus, PLM XML is an incorporation of product, part, and process information, rather than the "geometry-only" approach of other open formats. It consists of three main aspects: product structure, shape information and process information. The product structure includes not only the typical product structures in a CAD system (e.g. hierarchies of parts and sub-assemblies), BOM (Bills of Materials) and visualisation applications, but also the information that is required for PDM (Product Data Management) systems, such as revisions and effectivity. Multiple representations for shape information are supported by PLM XML, for example, B-rep for CAD models and a facet representation for visualisation applications. Process information refers to any type of process-specific (non-geometric) information associated with the product to support, for example, manufacturing processes, features identification and exchange, and visualisation applications.

In PLM XML, instance graphs, which use instances to describe the relationships between all sub-assemblies and parts in an assembly, are applied so as to reduce the amount of duplication of shape information. In addition, instance graphs allow one to reference external representations of part shapes, using URI and pointer mechanisms, and therefore decrease the file sizes further.

So far, PLM XML has been used extensively in UGS applications, for example, Teamcenter products are able to communicate with other applications by externally generated PLM XML files; and UGS can use PLM XML in their internal translator development. Furthermore, PLM XML is defined by a set of open XML schemas, therefore it is able to transport externally attached data while complying with the core of PLM XML.

Table 1-Summary of current Lightweight Representations

	Compression method	Developer	Supporting tools	Applications	Characteristics
U3D	Domain specific Node/Resource mechanism	Intel and the 3D Industry Forum (3DIF)	Adobe Acrobat Version 7.0 Open source	Sales & marketing Customer support Online promotions Maintenance Training	Level-of-detail Progressive streaming Rigid-body & skeleton-based animation File format and run-time extensibility
X3D	Domain-specific Type-specific field Fast InfoSet	Web3D Consortium	X3D Tools and Applications	Technical publishing Maintenance manuals, Websites, Database applications, Visual simulations Navigation systems	No heavy browser XML-based open profile/ components - based architecture Integration of advanced 3D techniques
3D XML	3D graphics compaction algorithm Reference/instance mechanism	Dassault Systemes/ IBM	V5R15, CATIA, Delmia, Enovia, Novia, Spatial, SmartTeam, SolidWorks, Virtools Dev 3.5 3D XML Player	Technical documentation Maintenance manuals Marketing brochures Websites Email	Level-of-Detail Multi-file architecture Easy to adopt Extensibility
JT Format	Lossless compression algorithm Domain-specific	Engineering Animation, Inc./ UGS Inc	JT Open Toolkit JT2Go	Lightweight visualisation format for PLM CAD-neural exchanging format Consistent 3D visualisation	Neutral exchange format Support of multiple files
PLM XML	References to external files	UGS Inc	UGS applications Open XML schemas	Connection UGS PLM Solutions products and third party adopter applications	Incorporation of product, part and process information Open source Support of multiple representations for shape definition Extensibility

A summary of current lightweight representations is given in Table 1. From the reviews, it can be seen that lightweight representations have been successfully developed into commercial systems, and their applications cover many different industries, such as engineering, healthcare, oil exploration and astronomy. Secondly, compared to full CAD model formats, current lightweight representations have several advantages, such as smaller file sizes, platform/application-independence, enhancement of progressive streaming, and multiple levels of detail (LOD) for rapid display. Thus, they have already shown benefits for collaborative product development, especially between geographically distributed applications and users.

However, as shown in Table 1, the main compression methods adopted currently are approximate or simplified geometric representations with domain-specific compression, while the compressed information or domains are out of the control of users. Thus, most of the representations developed so far (e.g. U3D and 3D XML) can only be regarded as lightweight 3D visualisations, and their applications are limited to later in the product lifecycle, e.g. customer services, sales/marketing tools, repair guides, assembly instructions and catalogues. JT Format and PLM XML are the lightweight representations that are announced to support the whole product lifecycle, but there are still some problems that decrease their effectiveness when applied to applications in PLM. For example, users at different stages throughout the product lifecycle need different information, and obviously a single lightweight representation can not satisfy this requirement.

X3D, 3D XML and PLM XML are all XML-based formats. XML is a generic markup language that allows users to define their own tags based on the specific needs of a document. XML has recently become popular because: 1) it is related to HTML (HyperText Markup Language), which is conventionally used to create Web pages, thus it is a good compromise between being easy for humans to read and write, and being easy for computers to interpret; 2) it is extensible, which allows for tag definition reflecting the tailored structure of information for various applications. Although the nature of XML makes platform and application-independence much easier, various viewers, toolkits and plug-ins are still needed for the XML-based lightweight representations (i.e. X3D, 3D XML and PLM XML) developed so far. For example, a test has been performed for PLM XML. The results show that PLM XML can be successfully converted and read in Solid Edge V16, but failed to be read in NX3 and NX4 without the support of the PLM XML Software Developer Kit (SDK). Thus, there is still a risk of losing access to the information in current lightweight representations.

With the above discussion in mind, it is necessary to propose a strategy that generates different lightweight representations for different users and partners, and "lightens" the CAD model while at the same time maintaining the necessary information.

3. LIGHTWEIGHT REPRESENTATIONS FOR MULTIPLE VIEWPOINTS

To address the above problems, a framework of lightweight representations for the product lifecycle is proposed, which is shown in Figure 1. The proposed framework integrates a markup method and a Representation Information Registry/Repository with a CAD model technique.

As the foundation of lightweight representations, CAD models have three main limitations that hinder the application of current lightweight representations in covering the whole product lifecycle.

The first limitation is that CAD models cannot embody the various degrees of complexity for a product. This means that users cannot identify the compressed parts in a CAD model according to different user requirements. For example, the fuse shown in Figure 2 consists of four parts: the fuse body (tube), fuse element (wire) and two end caps. The lightweight representation for assembly engineers should retain the assembly relationships between the caps and the tube, and the caps and the fuse, and the product length and tolerance (shown in Figure 2 (a)), but eliminate the detailed manufacturing information, such as the detailed dimensions and tolerances of the tube, wire and caps (shown in Figure 2 (b) – (d)). On the other hand, the lightweight representation for part manufacturing engineers needs to retain the manufacturing details, but may omit the assembly relationships.

The second limitation is that the information that is embedded in the current CAD model is not enough for the all the stages during product lifecycle. At present, most CAD systems implement a hybrid-modelling strategy, such as integration of sheet/surface/solid representations with parametric and feature-based design. The geometrical models (e.g. sheet/surface/solid representations) mainly describe geometrical and topological information of a product. Although feature-based models or parameter-based models have tried to encapsulate information of engineering significance, extra information is still needed for differing users at certain stages. For the example shown in Figure 2 (e), the packaging company needs additional information, such as the quantity of fuses in each packet. The manufacturing engineers also need extra information, such as surface finish (e.g. Ra 0.2 in Figure 2 (c)) and material treatment (e.g. tin clad in Figure 2 (d)).

The third limitation is related to the protection of intellectual property. All companies have recognised that CAD models are important intellectual capital, especially in a collaborative working environment. A water-marking method has been proposed to deal with security issues [Chou and Tseng, 2006]. However, no solution has so far been devised that allows for levels of security within a CAD model to vary, so that the CAD model displays different subsets of information according to the security privileges of the user. Ideally, the same CAD model should be able to display the product length and diameter to the packaging partner, whilst reserving the detailed tolerances, surface finishes and material treatments for the eyes of internal designers and manufacturing engineers.

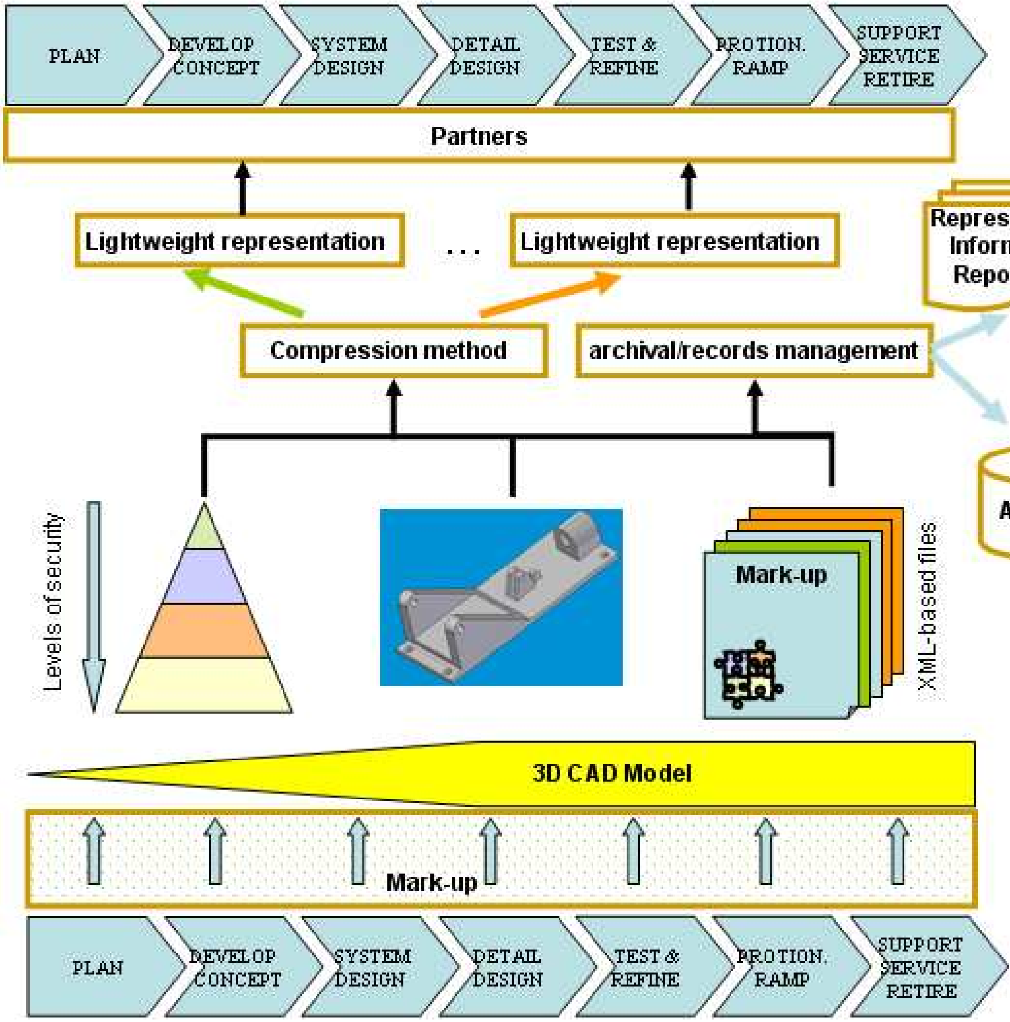

Figure 1 – Framework of lightweight representations in product lifecycle

3.1. MARKUP METHOD

The markup method initially appeared as a way of adding descriptive information, such as logical structure, to text or word-processed documents by means of inserted characters or symbols. The markup method was extended to 3D CAD models by Davies and McMahon [2006], as part of an exploration of approaches to multiple-viewpoint representation. The basic method is to embed extra information (such as machining surface) into CAD models by annotation. This method has demonstrated that, using markup techniques, a CAD model can be integrated with additional engineering and non-engineering information not currently supported by the established formats. Thus, it can be seen as a potential solution for the extension of a CAD model, according to the requirements of lightweight representations in the product lifecycle.

Firstly, in the proposed framework, the generation of a CAD model is extended from the design stage into the whole product lifecycle. The users throughout all stages are allowed to mark up the CAD model according to their different requirements and experiences. Secondly, the markup information can be embedded into CAD models, but most of the information will be recorded in a series of separate markup files written in XML format. Each markup file is based on a certain view for users or editors and linked to the CAD model through a specific element of the model. The elements of the CAD model cover different levels of details from the assembly and parts to detailed parameters and tolerances. Such linkage mechanisms can avoid the CAD model becoming too heavy and help to speed the process when several users mark up the same CAD model at the same time. Thirdly, the markup scope is expanded in three respects: 1) The insertion of the extra information that is needed for a certain point of view; 2) The embedding of a commercial security level for a certain partner or user; 3) The identification of the parts of information in a CAD model that are necessary for a certain point of view. Finally, for a different point of view or partner, users can compress the CAD model into a lightweight representation according to the corresponding level of security, accompanied by one or more markup files.

Figure 2 presents an example from the fuse manufacturing industry. It shows the markup needed for different points of view, such as manufacturing engineers, the assembly partner, the packaging partner and the marketing staff.

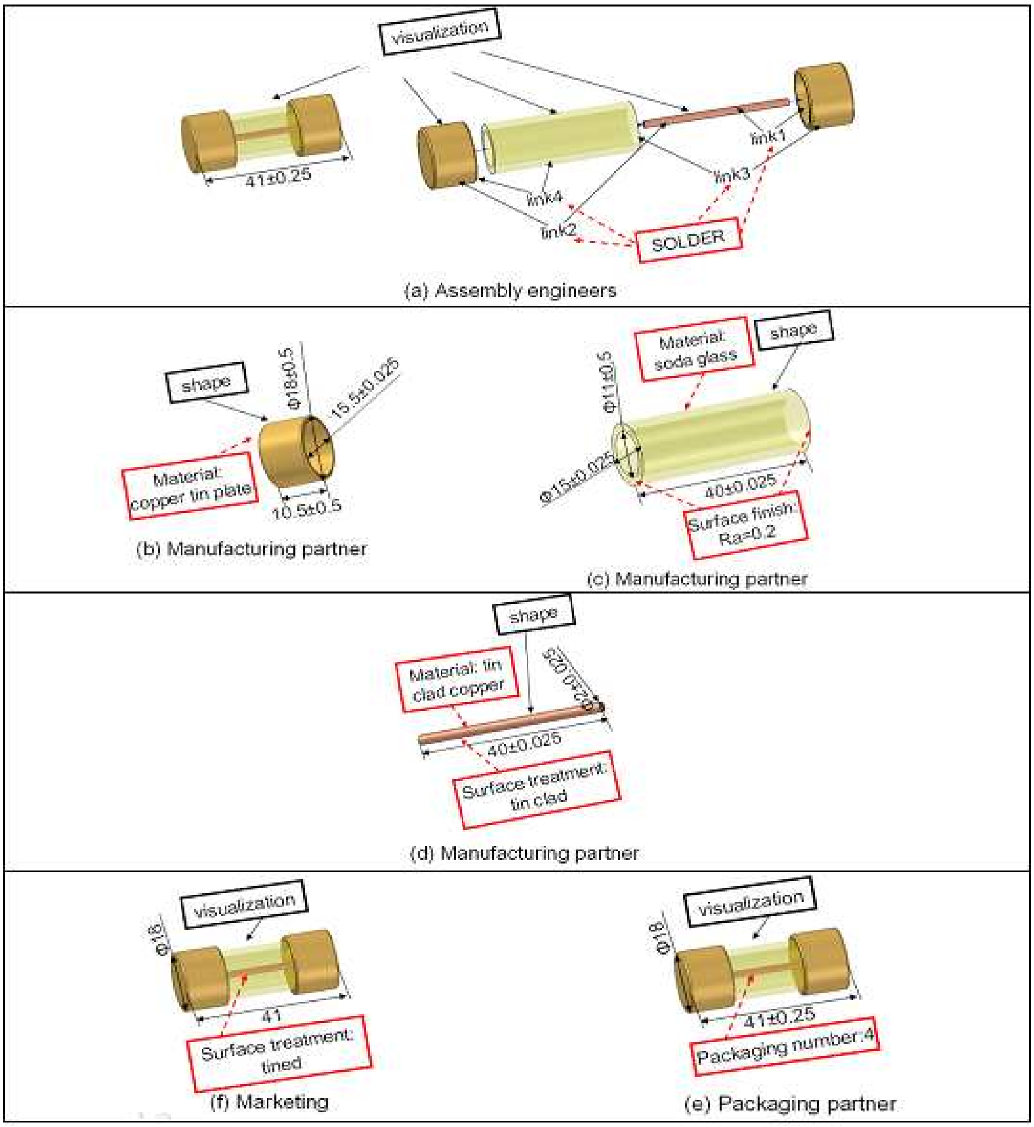

Figure 2 Markups for the fuse product

3.2. REPRESENTATION INFORMATION REGISTRY/REPOSITORY

Representation information' is a term that comes from the Open Archival Information System (OAIS) Reference Model [ISO 14721], an international standard model for describing the activities of data repositories and other long-term stores of information. It is defined as the information required to turn a data object (commonly a stream of bits) into something meaningful, and therefore includes such things as format specifications, data dictionaries, ontologies and sets of hardware and software known to be relevant to a format. In short, it is the information needed to keep a data object perpetually understandable.

While digital data objects typically need some representation information peculiar to themselves, they also share representation information with objects of a similar format or type. To avoid having to rediscover and store this information every time it is needed, it can be stored in a persistent registry and linked to whenever needed. The UK's Digital Curation Centre (DCC) and the European CASPAR Project are working on a Representation Information Registry/Repository for objects from across the spectrum of culture, art and science. The framework proposed in this paper would use the Registry/Repository to store relevant representation information for CAD model and lightweight visualisation formats, as well as XML schemata for markup documents.

In the short term, the principal usefulness of representation information is to enable informed decision-making on which formats would be most suitable for particular users, viewpoints and purposes, and which tools would be most reliable for which types of processing. Clearly, for these purposes the value of the information is in having a sizeable collection that can be cross-searched, rather than in individual pieces of information. Later in the lifecycle, though, the same information can be used to support file recovery procedures and other interpretive activities. In this case, the persistence of the relevant, individual pieces of representation information is critical, with preserved software tools taking on perhaps more significance than specifications and other more descriptive types of information, though descriptions of former practice and terminology will certainly be useful.

4. CONCLUSIONS AND FUTURE WORK

Nowadays, companies face the unprecedented challenges of the global market, collaborative environments and the entire product lifecycle. There are new demands on product representations, including platform/application independence, support for the product lifecycle, rapidly sharing information between geographically distributed applications and users, and protection of commercial security. To meet these requirements, some lightweight representations have been developed and applied in different industries. From the survey, it can be seen that current lightweight representations have been successful in some areas, such as the reduction of file sizes, cross-platform support, and enhancement of progressive streaming. However, current lightweight presentations adopt approximate or simplified geometric representations and domain-specific compression methods, and therefore their applications are limited to the later stages in the product lifecycle, such as customer service and marketing. Meantime, though XML is widely used, there is still a risk of losing access to the information in current lightweight representations.

Aiming to address the problems of current lightweight representations, a framework of lightweight representations for the product lifecycle is presented in this paper. In the proposed framework, a markup method is applied to the whole product lifecycle. More engineering and non-engineering information, commercial security information, and viewpoint-specific information, are attached to the CAD model by a series of separated markup files. Based on these markup files, a system of levels of security, and the original CAD model, various lightweight representations for different viewpoints during the product lifecycle can be generated. Similarly, an archival function can store the product information and its representation information into a private archive and a representation information registry (or network of private and public registries) respectively. As all CAD models are application dependent and "too heavy", lightweight representations are a good solution for archival purposes. Meantime, the markup files linked to the CAD model can also provide the evidence needed by the archival function to decide which information is worth preserving for the long term.

A prototype implementation for marking up CAD models has been developed using the NX3 API for which Figure 3 illustrates the interface. Obviously, the lightweight representations for product lifecycle are still at an early stage and further work is needed. The next stage will focus on the following tasks:

1) A friendly interface to allow rapid, collaborative marking up of CAD models. The developed implementation for marking up CAD models only supports a single user at present, but for the future it will need to support a collaborative environment and allow multiple users to mark up a CAD model and to generate separate XML files at the same time.

2) Partial geometrical compression methods. Current geometrical compression methods simplify the whole assembly/part model, but are unable to optimize according to users' requirements, such as compressing part of a CAD model but retaining other parts of the model uncompressed.

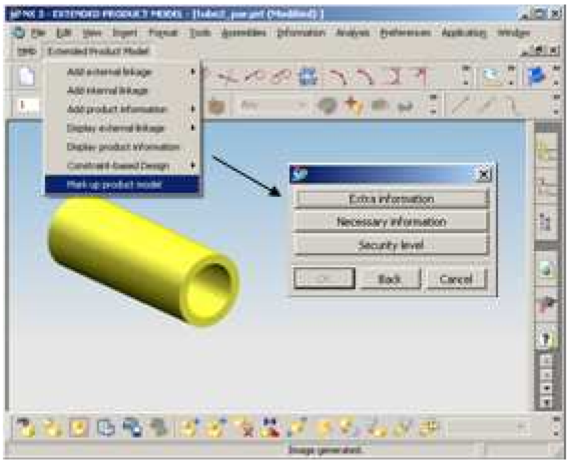

Figure 3 Interface of markup for single user

5. ACKNOWLEDGMENTS

The work reported in this paper has been supported by a number of grants for Engineering and Physical Sciences Research Council (EPSRC), involving a large number of industrial collaborators. In particular, current research is being undertaken as part of the EPSRC Innovative Design & Manufacturing Research Centre at the University of Bath (reference GR/R67507/01). The authors gratefully express their thanks for the advice and support of all concerned.

REFERENCES

Business Wire, Dassault Systemes Delivers 3D XML Specifications and Player – Universal Lightweight Format Enables 3D Democratization, June, 2005, http://findarticles.com/p/articles/mi_m0EIN/is_2005_June_15/ai_n13814506

Chou C. and Tseng D., "A public fragile watermarking scheme for 3D model authentication," *Computer -Aided Design,* 28, 2006, 1154-1165

Dassault Systemes, 3D XML User's guide, Version 2 Edition 0, January 2006

Davies D., and McMahon C.A., 2006, Multiple Viewpoint Design Modelling through Semantic Markup, *Proceedings of IDETC/CIE 2006,* ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Philadelphia, Pennsylvania, USA

Dexigner, Acrobat 3D Software to Accelerate Design Collaboration Processes at Renault, 21/03/2007, http://www.dexigner.com/digital/news-g10553.html

Ecma International, Universal 3D File Format, 3rd Edition, June, 2006, ECMA 363

Kunze, J., Future-Proofing the Web: What We Can Do Today, 16 September, 2005, *International Conference on Preservation of Digital Objects*, Göttingen, Germany

ISO 14721:2003, Space data and information transfer systems — Open archival information system — Reference model.

PLM Components, UGS, Open product lifecycle data sharing using XML, Write paper: PLM XML, www.ugs.com, October, 2005

Sourceforge.net, Universal 3D Sample Software, https:// sourceforge.net/project/showfiles.php?group_id=133107

The Xj3D Project, http://www.xj3d.org/

UC Libraries, Overview of the UC Libraries Digital Preservation Repository (DPR), University of California, 2006

UGS, JT2Go – A New Level of Customer – Supplier Collaboration, http://www.jt2go.com/

UGS: Newsroom, Siemens Joins JT Open Program, Adding its Global Manufacturing Leadership to Growing Momentum of Standard JT 3D Data Format, February 14, 2005, http://www.ugs.com/about_us/press/press.shtml?id=3936

U3D: Information From Answers.com, http://www.answers.com/topic/u3d

Versprille, K., Dassault Systèmes' Strategic Initiative: 3D XML for Sharing Product Information, July, 2005, Technology Trends in PLM

Virtools, 3D XML Virtools™ Plugin, http://www.virtools.com/solutions/products/virtools_3dxml_plugin.asp

Web3D Consortium, www.web3d.org

Wiki, JT (visualization format – Wikipedia), http://en.wikipedia.org/wiki/JT_(visualization_format)