INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

STRATEGIES FOR THE COLLABORATIVE USE OF CAD PRODUCT MODELS

Lian Ding[1], Alexander Ball[2], Manjula Patel[2], Jason Matthews[1] and Glen Mullineux[1]
(1) Department of Mechanical Engineering, University of Bath, UK (2) UKOLN, University of Bath, UK

ABSTRACT

Today, more and more companies are shifting from design and manufacturing to provide through-life support. Product models, as one of the most important types of product information, must be communicated, shared, and retrieved between distributed design teams and various users or partners at different stages of a product lifecycle. However, the conventional CAD models cannot satisfy these demands due to the ephemeral nature of the CAD systems and issues of proprietary formats, the protection of intellectual property, and the recording of domain-specific information. To overcome some limitations of CAD models, lightweight representations have been developed during the last decade to support collaborative product development. This paper aims to address the issue of long-lived product models using a combination of CAD models and lightweight representations. A new strategy, which integrates the techniques of the annotation of product models and the registration and storage of representation information, is proposed to support product models and other related information to be shared and reused over the long-term product life. Three key issues in the proposed strategy have been addressed: storage of extra information by the 'stand-off' annotation of product models; sharing information among CAD models and their derived lightweight representations through the consistent identification of geometry; and keeping product data readable and reusable in the future by long-term preservation of representation information. An implementation has proved that the proposed strategy can be applied using current software techniques.

Keywords: *product model, CAD, Lightweight representation, representation information repository*

1 INTRODUCTION

To survive an increasingly global economy and unprecedented competitive market, more and more companies are shifting from design and manufacturing to provide through-life support. Product information, especially the detailed information of physical products generated after embodiment design stage (i.e. product models) must be shared with users from across the whole product lifecycle, which could last over 20-30 years. Meanwhile, companies are expanding multinationally so that people, departments and partners involved in the product lifecycle tend to be geographically dispersed within various collaborative relationships. Evidence [1, 2] from such collaborative environments has shown up many issues concerning the communication, sharing and retrieval of conventional product models (i.e. Computer Aided Design [CAD] models) between separate design teams or different users at different stages of a product lifecycle, such as commercial confidentiality, proprietary formats, multiple viewpoints on the model, and large file sizes. Lightweight representations have recently been developed aiming to support all users of the product lifecycle in rapidly browsing, retrieving and manipulating product information. Compared to the full CAD model formats, the lightweight representations have several advantages, such as smaller file sizes, greater platform/application-independence, protection of intellectual property, and more openness. Thus, they have already shown benefits for collaborative product development with wide applications in various industries. However, lightweight representations are derived from the CAD models using techniques that can preserve some of the problems of the original. For example, much of the domain-specific information (e.g. non-geometrical manufacturing information) is still missing. In addition, most of the current lightweight representations focus on visualisations of 3D models, and little work has been done for application support, such as how to feedback information to designers. Thus, the applications for these representations are still limited to the purposes of marketing and service.

There are two active areas of research that show promise for overcoming the above challenges:

- The annotation of CAD models with extra information (metadata, application-specific information, etc.) allowing the same data to support multiple viewpoints;
- Collections of representation information that assist in keeping product data readable, understandable and reusable long into the future.

This paper proposes a new strategy to integrate the advantages of the two methods in order to support product models and related information, enabling them to be shared and reused over the long-term. The subsequent content is organized as follows. Section 2 reviews the current status of product models including conventional CAD models and lightweight representations, and the challenges for product models to support the whole product lifecycle in a collaborative environment. Section 3 proposes a strategy for long-term usage and preservation of product models. Three key issues within the strategy are described and discussed: the techniques of annotation of 3D product models, sharing information among the CAD models and their derived lightweight representations, and the representation information registry for long-term curation of product models. Section 4 presents the implementation with an example. Finally, conclusions are drawn and further work is presented.

2 LITERATURE REVIEW

2.1 Product models

Conventional product models are usually developed within CAD/CAM applications (hence they are usually called CAD models). There are two basic types: Boundary Representations (B-Rep) that represent shapes by their external boundaries (structured collections of faces, edges and vertices); and Constructive Solid Geometry (CSG) that constructs models as a combination of simple solid primitives, such as cuboids, cylinders, spheres, cones, etc [3]. In addition, further modeling techniques such as freeform surface modeling and feature-based models, have been extensively used. Freeform surface modeling represents parts with complex surface curvatures using functions or approximations to represent those surfaces, such as Non-uniform Rational B-spline (NURBS) and Bezier surfaces. The feature-based models or parametric-based models aim to encapsulate the engineering significance of portions of the product geometry and, as such, are applicable in product design, product definition and reasoning about the product in a variety of applications such as manufacturing planning [4]. The conventional CAD models focus on geometric and topological depictions of the product, such as assembly relationships, shapes and dimensions. Thus, they are good at representing the design and engineering aspects of a product during the configuration design, assembly design or detail design stages, and therefore have been popularly used for product design, from kitchen utensils to modern automobiles and jet aircraft. However, with comprehensive collaboration, not only with the geographical distributed design teams but also the partners/contractors at different stages of a product's lifecycle, some technical problems and limitations of the CAD models appear. Firstly, the CAD models are too 'resource-heavy', and may not easily be transmitted between geographically distributed applications and users. A 3D CAD model of a relatively simple component could be multiple megabytes in size and hundreds of such components may be included in a product (e.g. a car), leading to very large storage requirements for models and hindering the options for communication of such models. Secondly, CAD models contain the details of the products. As these details are among companies' most important intellectual property (IP), they are often unwilling to share all the details of their product models directly to avoid exposing commercially sensitive information to their competitors. Thirdly, each CAD system 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 their native representations. Migrating models between CAD systems, and subsequently repairing any translation errors, is by no means a simple or inexpensive task, either: in 2007, almost 40 Tb of CAD data were migrated, creating a US$5.5 bn market for about 150 service providers worldwide [5].

To overcome the above limitations, lightweight representations, which do not contain the full CAD model data, and/or which exploit data compression in some way, have recently been proposed. Compared to full CAD model formats, lightweight representations have the following significant differences.

Smaller file sizes. Typically, lightweight representations have small file sizes and, when generated with approximate geometry by various compression methods, help to protect sensitive design information. Domain-specific compression, such as that used by U3D [6], eliminates most of the engineering data associated with the original model. Reference/instance mechanisms allow object geometry to be specified once and instantiated at many different points throughout the model by reference to that specification; examples can be found in formats such as 3D XML [7] and U3D. The method by which geometry is approximated can also have an impact on file sizes. For example, 3D XML adopts a single surface patch instead of a tessellated form to accurately approximate a portion of a CAD model's surface area.

Published specifications. Currently, most lightweight representations have freely available documentation. For example, the specification for U3D, the 4th edition of Ecma International's ECMA-363 Universal 3D File Format Standard [6], the JT file format reference [8], and the schemas of PLM XML [9] are all open to the public and can be downloaded and used freely.

Inexpensive viewing applications. In contrast with CAD models, it is normally possible to view lightweight representations without the aid of expensive CAD packages. For example, U3D is supported from Version 7 of Adobe Acrobat (PDF Version 1.6). X3D tools and applications are freely available (http://www.web3d.org) and there are several popular X3D plug-ins for Web browsers, e.g. Xj3D (www.xj3d.org), Flux™ (www.mediamachines.com) and BS Contact™ (www.bitmanagement.de). 3D XML can be used in office applications and Web browsers as well as Dassault's product developers' tools, and Virtools Dev 3.5 [10] is a software tool that uses 3D XML for creating real-time 3D applications with complex interactivity. A C++ library, JT Open Toolkit, has been developed, which is able to create, read and access JT formatted data on various hardware and operating systems. Ball, et al [11] and Ding, et al [12] provide a review of some of the more common formats. Lightweight representations have already shown benefits for collaborative product development, especially between geographically distributed applications and users.

2.2 Challenges for product models in long-term usage and preservation

With the increasing demands on digital information flow covering the whole product lifecycle, not only do 3D product models need to be preserved, they also need to remain usable in the long term. As reviewed above, the CAD models and the lightweight representations each have their own characteristics. The CAD models are coming to dominate as the definitive record of product data at the product design stage. The lightweight representations are derived from CAD models to support geographically distributed collaboration. The combination of CAD models and lightweight representations has the potential to address the issue of long-lived product models. However, there are still some technical issues that hinder an effective strategy of long-term product representation and the challenges are addressed as below:

- The CAD models are centered on geometric and topological depictions of the product, but lack the ability to model high-level design and engineering context and semantics. Product information includes not only the geometrical information of a physical product (i.e. conventional CAD models), but also the information defined at early stages of design (e.g. the specifications, required functions and sketches), the rationale evolved in the design activities (e.g. constraints and intention), and other non-geometrical information (e.g. performance capabilities and limits). The information, which is not recorded in the CAD model, is still greatly helpful for collaborating designers to understand each other and share knowledge correctly, especially when they are based at different sites. Meanwhile, such information could also be valuable for the support of the product lifecycle and design upgrades. Furthermore, various related product information generated throughout a product lifecycle, like manufacturing data and service feedback, can be an invaluable resource when repurposing an existing design, allowing evidence-based improvements to be made. Many issues need to be explored, such as: how to record and integrate the many structures of product information; how to embed such information into the product models; whether the current representation methods (e.g. CAD models) need to be modified; and how to retrieve the information if it is recorded well.

- Many engineering companies are expanding their business from design and manufacturing to the whole product lifecycle, which means the original product models may be needed over 20-30 years, such as for aircraft, cruise ships and cars. On the other hand, given the pace of CAD software development, one may confidently predict that a CAD system will be replaced several times during the lifecycle of the product. There is a danger, not only that the software will become obsolete within a few years, but that compatibility with newer generations will prove to be unreliable. Although there are some strategies to be proposed to preserve product model data, such as emulation of old software and hardware (virtual machines), and the use of neutral standards (e.g. Initial Graphics Exchange Standard (IGES) [13] and the Standard for the Exchange of Product Model Data (STEP) [14]), few successful applications have been provided due to heavy costs, unreliability of data, constraining reuse, and so on.
- The persistent identification of geometry, also known as the "persistent naming problem", has been described by Mun and Han [15] among others and concerns the identification of the surviving entities of the pre-edit b-rep in a post-edit b-rep. Here, we address this issue in translation from native CAD models to lightweight representations, and information sharing among the users who use the native CAD models and the corresponding lightweight representations, respectively. Most lightweight representations currently use facet representation to reduce file size, but at the possible cost of losing the geometry identifiers within the CAD models (although sets of facets may correspond with faces in the original model). As a consequence, the information that is associated with the 'same' entity cannot be concisely transferred or shared between the CAD model and its derived lightweight representations. It is clear that the consistent and persistent identification of faces and edges of a product, and further identification of special-viewpoint features by combination of the faces and edges, must be in place, but little work has been done to explore this so far.

3 STRATEGY FOR THE LONG-TERM USAGE AND PERSERVATION OF PRODUCT MODELS

3.1 Annotation for product models

Annotation can be simply defined as adding any extra information for various purposes, such as further explanations, viewpoint interpretation, extra descriptions of or comments on the existing entity. To date, annotation has been widely used in different digital items, including text documents, structured data (e.g. databases and tables), 2D images, 3D models and multimedia (e.g. video and audio). For an important subtype of annotation – formally structured annotation (sometime called as markup) – there are five basic types, which are defined as below [16, 17]:

- Punctuational annotation: where word, phrase, and sentence boundaries are identified by spaces, commas, full stops, and other punctuation characters inserted into the text.
- Presentational annotation: where the visual form of the document is specified directly.
- Procedural annotation: in which presentational instructions (or commands) for some particular processing system are embedded in the text.
- Descriptive annotation: the author identifies the element types as tokens, as often found in applications of SGML and XML, which approach documents as structured objects containing semantically interpretable parts.
- Meta-annotation: provides a facility for controlling the interpretation of annotation and for extending the vocabulary of descriptive markup languages (e.g. macros).

It should be noted that the descriptive annotation allows users to focus on both the structure and content of document so that it has greater portability. Additionally, descriptive annotation separates how data is stored and how it is used, and therefore can potentially provide any additional information required when transforming the object to different formats (e.g. native word processor format, HTML, PDF) for the purposes of presentation or accessibility in different contexts, or when the content needs to be re-processed or re-organized for different applications and uses (e.g. raw data of experiments and graphs of the experimental results). Thus, descriptive markup has been considered as one of the best solutions for manuscript composition and distribution.

Annotation has been used to augment engineering depictions of products with information that is difficult to represent unambiguously pictorially such as information required for manufacturing a product – tolerances, machining processes, surface finishes etc. – using formally specified systems of symbols normally recorded in some level of standard, such as the American Society of Mechanical Engineers (ASME) Y14.41 [18] issued in 2003. Work on annotation and markup of product models has been carried out by some researchers. For example, Elinson and Nau [19] use a graph to represent machining features, whose nodes and edges correspond to features and its relationships of features. The nodes and edges are annotated with labels by various parameters that may be useful for classification purposes; Hoffman and Joan-Arinyo [20] proposed an architecture for a product master model federating CAD systems with downstream application processes for different feature views and especially addressed the need to make persistent associations of design information with net shape elements. Ding et al [21] has shown the knowledge retention capabilities of extending 'annotation' in CAD systems to aid constraint-based redesign activities; Davies and McMahon [22] explore the application of a markup approach to the attachment of information to CAD models and its subsequent organization and manipulation; Hisarciklilar and Boujut proposed an approach based on the concept of annotation for supporting design communication [23].

'Annotation' support has also been introduced by many CAD systems and lightweight representation systems. However, such 'annotation' functions provided are normally limited to a single type (e.g. free-text, URI or hyperlink), they cannot satisfy the requirements of annotation schemes supporting context-specific information and multiple viewpoints. Furthermore, the annotation strategy normally adopted is the *'inline' annotation method*, in which the annotation information is actually associated with product models, and therefore it is only reusable for invariant topologies. Thus, an extension to existing 'annotation' functions is really needed. Ding et al. [24] have explored the possibility of using *'stand-off' annotation* to extend the 'annotation' functions of the CAD and lightweight representation systems. This is where the annotation is stored in a separate external document utilizing a system of references or pointers to the element (such as a face, or a feature) to which the annotation refers. Compared to the 'inline' annotation, the 'stand-off' annotation method offers many advantages. Firstly, the same annotation file can be applied to any copy of the model in any format; this allows the annotation information to be shared by a wide variety of users. Secondly, multiple independent annotation files can be safely applied to the same CAD model. Thus, it allows context-specific information to be stored in a number of separate files and passed around only as needed. Thirdly, annotations can be edited, circulated, and processed independently of the model, and the CAD model remains unchanged. This is significant as it allows downstream processes to be independent of the CAD model by only reusing annotations.

3.2 Sharing annotations between a CAD model and its lightweight derivations

Although 'stand-off' annotation has the advantage of supporting any copy of the model in any format, the annotation files can be shared only if the references or pointers used by the annotations are concisely transferred between the CAD model and its derived lightweight representations. However, most lightweight formats currently use facet representations to reduce the file size, and therefore lack bi-directional linkages between the entities in the original CAD models and those in their lightweight derivations. Assuming all the faces and edges of a product model can be persistently identified, features based on a specific viewpoint can be constructed as combinations of the faces and edges of the model. Correspondingly, the annotations that are added to the faces, edges or the features can also survive within both the CAD model and the lightweight representation. Thus, to investigate the possibility of preserving the names of faces and edges during the translation between CAD models and its lightweight representations, an experiment was carried out based on UGS NX3 and Adobe Acrobat 3D 8.1. The original CAD models (i.e. NX part format: .prt) were produced within UGS NX3; all faces and edges were given unique names (e.g. TEXT_FACE_1). The lightweight representations (PRC format) were generated by importing the NX3 CAD models into Acrobat 3D 8.1, with options set to import the maximum possible amount of information from the original file. Next, both NX3 and Acrobat 3D were used to output a STEP AP203 B-rep solid model. The Acrobat 3D export options were set to match as close as possible the behaviour of NX3 converter; for example, analytic shapes such as circles and cylinders were exported as analytic shapes rather than NURBS surfaces. These two STEP files were then compared for consistency. Finally, the Acrobat STEP file was imported back into

NX3 then exported to a further STEP AP203 B-rep solid model. This model was compared with the other two STEP files, and the differences noted. The detailed transfer route is given in Figure 1.

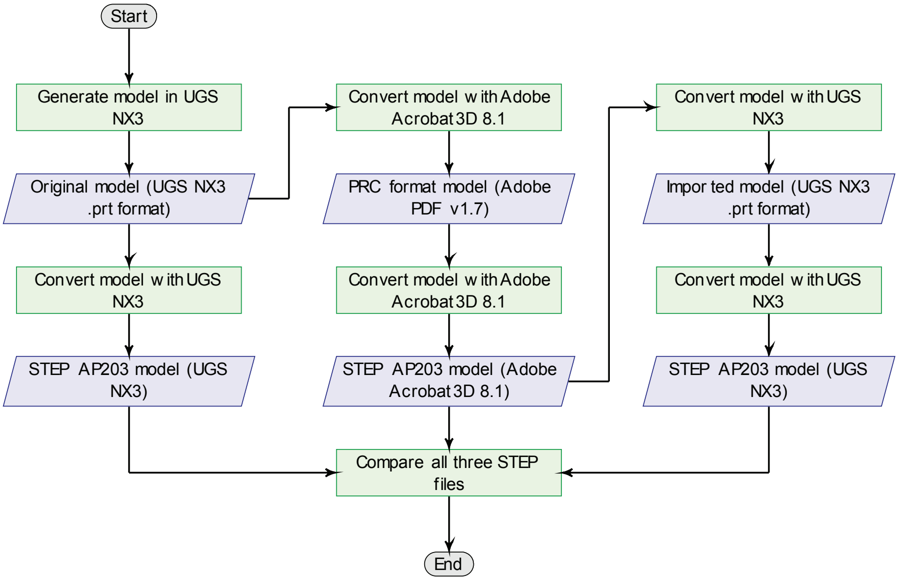


Figure 1. Flowchart indicating the stages of the experiment

In terms of geometry, the STEP file written out from the PDF file was syntactically quite different from the one generated by NX3. The majority of these differences, though, could be mapped and thus the changes wrought by one system could be undone by the other. As NX3 and PDF treat the geometric entities in different ways, it is unrealistic to expect that no information would be lost during the translation between the representations. However, the labels for faces and edges survived the translation from NX3 to STEP, but only the face labels survived into the Acrobat STEP file. This could be related to the fact that, under the Acrobat 3D 8.1 implementation of the PRC format, the only topological reference supported is to faces; PRC itself allows for topological references to multiple vertices, unique vertices, wireframe edges, B-rep edges, loops, faces, shells and areas bounded by shells. The face label used to describe the single cylindrical surface in the NX3 version was used to describe both semi-cylindrical surfaces in the Acrobat version. If these results are representative, then it offers the potential of tracking annotations that have been attached to named faces throughout the whole product life cycle.

In addition, the experiments have also investigated the issue of the long-term preservation of the metadata of the product model files. It has been shown that the metadata, like the status of files (e.g. approved or not), author, and authorisation, are not stable during these translations. Furthermore, there is a high risk that the contexts behind a product, such as design features and intents (e.g. the features' sequence) could be lost. Because of this, it is worth re-considering where such metadata and the contexts should be stored for long-term objectives. It would be a better solution to store various metadata and contexts in one or more separated annotation files rather than embedding them inside the original geometrical model directly, if a system of references can consistently indicate to which entities of the product the annotation refers.

3.3 Registry/Repository of Representation Information for Engineering (RRoRIfE)

Representation information is a term defined by the Open Archival Information System Reference Model [25] as 'the information that maps a Data Object into more meaningful concepts.' In the digital context, it typically covers file formats, rendering software and instructions on how to interpret the rendered contents. RRoRIfE is a tool that uses representation information to explore the options available for converting CAD models into other formats, whether for contemporaneous exchange or for long term archiving; it is intended primarily as a planning tool for information managers. Given this purpose, RRoRIfE does not use a comprehensive set of representation information, but rather a small subset describing the capabilities of various formats and processing software with respect to certain potentially significant properties. These properties cover 2D geometry, 3D geometry, modelling paradigms, compression, identification and metadata, and are intended to represent the syntax, semantics and properties that an information manager may consider important to preserve or discard in a format migration, depending on the context.

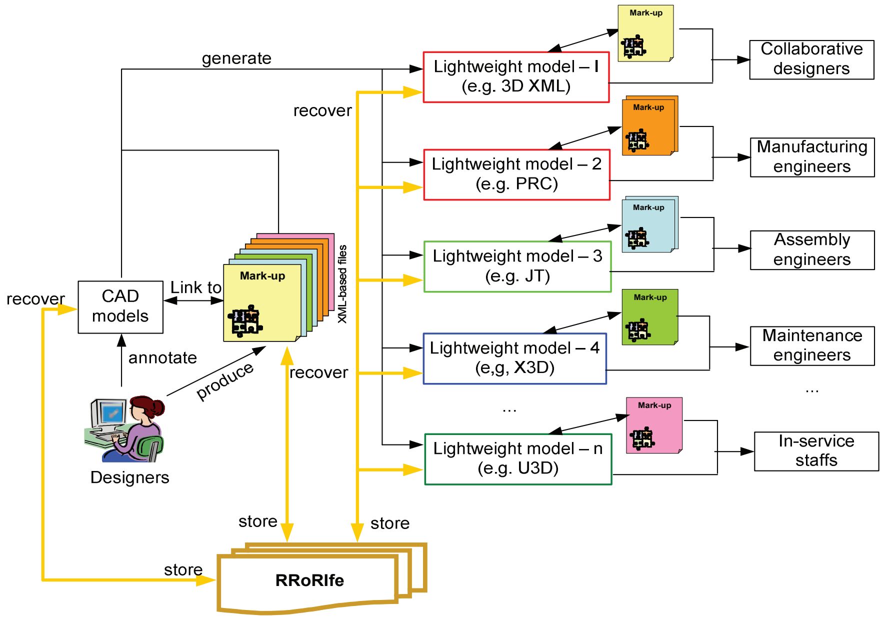
As well as simply allowing one to browse through the information contained in the self-contained repository, RRoRIfE allows one to perform three different types of search on it. The first allows one to search for all the (known) formats that fit a chosen set of criteria with respect to significant properties. For each property, one can specify that it should be fully supported, fully or partially supported, or not supported at all; otherwise it is not considered in the search. The second type of search calculates the possible migration paths between two formats, in a given number of steps or fewer. The third type of search allows one to specify a starting format and, as in the first type of search, a set of criteria for the final destination format. RRoRIfE will then calculate a set of suitable migration pathways with the specified number of steps or fewer, and perform some simple ranking on them.

The primary usefulness of RRoRIfE lies in managing the loss that occurs when translating a CAD model from one format to another. We anticipate it being used to identify what information a lightweight representation would lose from the original CAD model, thereby informing a) which lightweight format is most suited to a particular application and context, and b) whether additional information from the CAD model needs to be saved as 'stand-off' annotations. It could also be used to determine the tool or service with which to perform the migration. Given sufficient representation information, it could also inform the procurement process for a replacement CAD system.

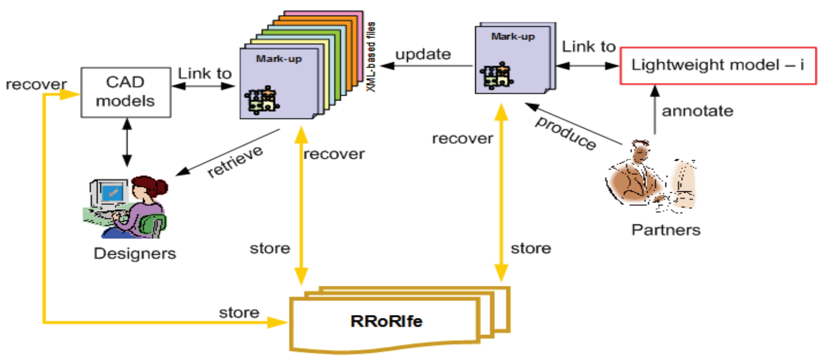
3.4 Proposed strategy

A long-term strategy for the use and preservation of product models throughout a product lifecycle is proposed in Figure 2:

- Designers are able to 'stand-off' annotate CAD models with various pieces of product related information, such as high-level design context, design constraints and intentions, performance limits and other domain-specific information (e.g. manufacturing requirements), etc.
- Collaborative teams and partners throughout the whole product lifecycle can share the CAD model or a lightweight derivation along with related annotations according to different viewpoints, purposes and security levels.
- The collaborative teams and partners can update related product information and share their particular expertise through 'stand-off' annotations associated with lightweight representations they are using. The 'stand-off' annotations are then able to link back to the original CAD model through persistent identification of entities in the CAD model and the derived lightweight representations. Therefore, the designers are able to retrieve these annotations through CAD models directly.
- RRoRIfE covers the whole product lifecycle to successively store relevant representation information for CAD models and lightweight representations, as well as the schemata for annotation files. Also, RRoRIfE can inform users which formats would be most suitable for viewpoints and purposes and support file recovery procedures.



(a) Sharing design information throughout a product life



(b) Sharing product lifecycle information

Figure 2. Strategy for long-term usage and preservation of product models

4 IMPLEMENTATION AND TESTING

An implementation has been developed to demonstrate the proposed strategy. Two commercial software packages have been used, which have been widely applied in both academia and industry – Unigraphics NX3 and Adobe Acrobat version 8.1. The contents of the annotations are recorded into separate XML (Extensible Markup Language)-based documents, and therefore it is possible to define different schemas and freely manipulate the associated annotations between different versions according to different requirements and security/IP profiles. A program to write/read the XML documents has been developed using Visual C++ and Microsoft XML Core Services (MSXML). The NX3 Open C API of User Defined Objects (UDO's) is employed to link the annotation documents to the specific entities to which the annotations refer. In addition, the UIStyler dialogue in NX3 has been utilized to develop a friendly user interface. A design for a shelf-ready packaging machine is chosen for a testing example. Figure 3 (a) shows the interface that allows a designer to annotate the CAD

model with extra information, such as design rationale (e.g. that the distance dimension for the 50MM shaft is designed according to the carton constraints) so that other designers or users at other stages of the product lifecycle can share the information. Figure 3 (b) gives the interface that allows a designer to retrieve extra information from the CAD model, which is produced by other team members and partners. For example, the engineers feed back that the surface treatment of part of the side_plate_end has a detrimental effect.
The JavaScript API provided by Acrobat 8.1 has been adopted to develop the 'stand-off' annotation in PDF. In the API a node – an object with a 3D representation within a scene hierarchy – has the properties 'info' (Acrobat 7.07+) and 'metadataString' (Acrobat 8.1+), which give access to information associated with the node, presented as strings [26]. In addition, Acrobat 8.1 introduced a new property of the scene object, selectedNode, which enables one to query which node in the tree currently has focus. This is used to provide alternative annotation tools. As shown in Figure 3 (c), the annotation interface in PDF allows engineers to select a specific node (e.g. a face and a part) in the model that they want the annotation to link to, insert free-text comments, attach related supporting files, and export an external XML-based annotation file as shown in Figure 3(e). Figure 3(d) gives an example of retrieving an annotation inserted in the CAD model by selecting a specific node in the PDF and reading the external annotation file. In addition, a transfer identifier is written by Visual C++ and the NX3 Open C API to transfer the annotation files from PDF environment to specific entities of the original CAD model. The interface is shown in Figure 3(f).

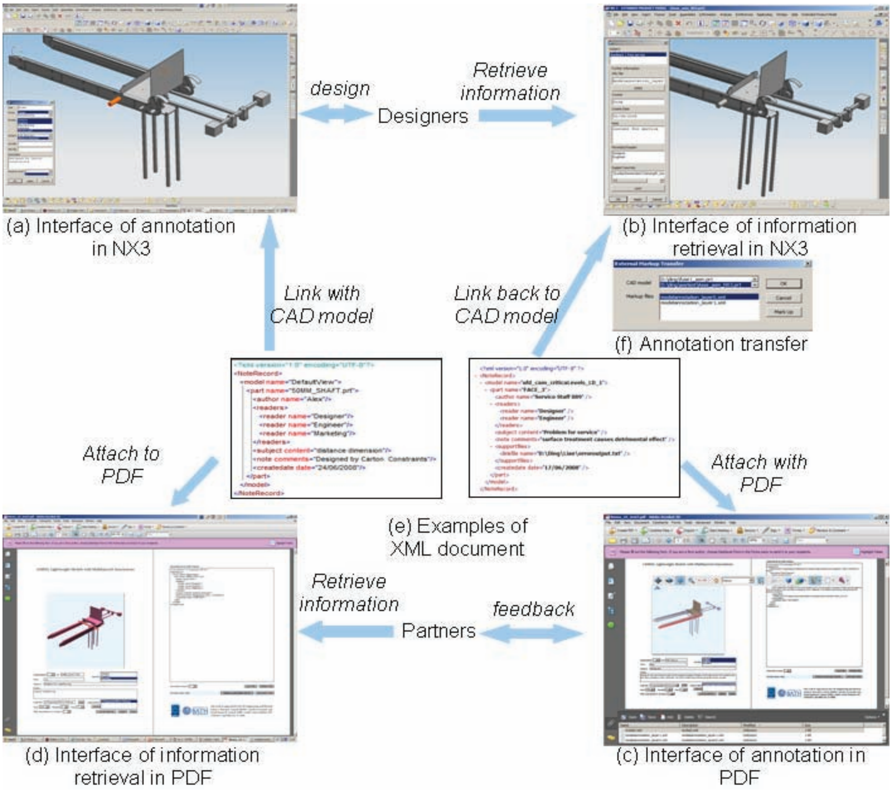


Figure 3. Interfaces for the implementation

RRoRIfE has been written as a Java application for use with version 6 of the Java Standard Edition Runtime Environment. The installation consists of a JAR file and a repository directory, the latter holding the representation information used by the application. The representation information itself is held in XML files; each one holds information about a single file format or piece of software. File format XML files record some identification information, and how well the format supports a particular property: 'full' and 'none' have obvious meanings, while 'partial' indicates the support is limited in a way explained in an accompanying note. Software XML files record identification information, and then detail the different format migrations of which the software is capable; the migrations are differentiated not only by the input and output formats, but also by the user-configurable migration options. For each migration, the XML file records how well file format properties are preserved under the migration: 'good' indicates sufficient fidelity to support a lossless round trip, 'none' indicates complete loss, 'poor' indicates that loss or corruption is at least equally as likely as fidelity with any realistically complex model, while 'fair' indicates a state between 'good' and 'poor' which must be explained in an accompanying note. For properties where degradation rather than corruption or loss occurs, the XML file also records whether that degradation is configurable, fixed (i.e. the same each time) or unpredictable. The interface to RRoRIfE is shown in Figure 4.

Figure 4. RRoRIfE interface

5 CONCLUSION AND DISCUSSION

With more and more companies considering through-life support, there is an increasing demand for sharing, using and preserving product models in a collaborative environment for a long-term period. However, the conventional CAD models cannot satisfy these demands due to the ephemeral nature of the CAD systems and issues of proprietary formats, the protection of intellectual property, and the recording of domain-specific information. After reviewing the status of product models, including the CAD models and the latest product representations, lightweight representations, a new strategy was introduced that integrates the techniques of the annotation of product models and the registration and storage of representation information, was proposed to assist product models and other related information to be shared and reused covering the whole product life. The proposed strategy has three key characteristics: storage of extra information (domain-specific information, high-level design context, metadata, etc.) by 'stand-off' annotation of product models; sharing information among the CAD models and the derived lightweight representations through consistent identifying geometry; and keeping product data readable and reusable in the future by long-term preservation of representation information. An implementation has proved the application of the proposed strategy by the support of current software techniques.
However, in comparison with other methods, such as "in-line" annotation, the proposed method needs to answer two major questions for practical applications: how to maintain the annotations when a

CAD model is changed; and how to re-organize information that is linked to the CAD model for efficient retrieval. Obviously, future work will concentrate on 1) the expansion of the applications of the proposed strategy to industry; 2) the reorganization of annotations for knowledge management using techniques like data mining, and 3) the exploration of how annotations should cope with changes to underlying model.

ACKNOWLEDGMENTS

The work reported in this paper has been supported by the UK Engineering and Physical Sciences Research Council (EPSRC) Innovative Design & Manufacturing Research Centre at the University of Bath (reference GR/R67507/01), the EPSRC and the Economic and Social Research Council (ESRC) under Grant Numbers EP/C534220/1 and RES-331-27-0006.

REFERENCES

[1] Li, W.D. and Qiu, Z.M., State-of-the-art technologies and methodologies for collaborative product development systems, *International Journal of Production Research*, 44 (13), 2006, 2525–2559
[2] Lang, S.Y.T., Dickinson, J. And Buchal, R.O., Cognitive factors in distributed design, *Computers in Industry*, 48(1), 2002, 89-98
[3] McMahon C. and Browne J., *CADCAM, From principles to practice*, Addison-Wesley Publishing Company, 1996, ISBN 0-201-56502-1
[4] Shah, J.J., and Mäntylä M. Parametric and Feature-based CAD/CAM, John Wiley & Sons, Inc., 1995
[5] Prawel, D., CAD Migration – The "Killer App"? *MCADCafe Weekly*, June 30, 2008, http://www10.mcadcafe.com/nbc/articles/view_weekly.php?articleid=551766 (accessible by 13/01/09)
[6] Ecma International, Universal 3D File Format, 4th Edition, June, 2007, ECMA 363
[7] 3D XML User's guide, Version 2 Edition 0, Jan., 2006
[8] JT Files – File Structure, http://www.jtopen.com/ technology/jt_files_overview.htm
[9] PLM XML SDK V6.0.1 - PLM XML Schema Functional Description, November 2005, UGS
[10] Virtools (2007), 3D XML Virtools™ Plugin, http://www.virtools.com/solutions/products/virtools_3dxml_plugin.asp
[11] Ball, A., Ding, L., Patel, M., An approach to accessing product data across system and software revisions, *Advanced Engineering Informatics*, 22(2), 2008, 222-235
[12] Ding, L., Li, W.D., McMahon, C.A., Patel, M., Light-weight Representations for Product Lifecycle Management, *Virtual Concept* 2006, Mexico, 26th November – 1st December
[13] US Product Data Association. *Initial Graphics Exchange Specification (IGES) v5.3*, ANSI, 1996. Retrieved July 12, 2008, from http://www.uspro.org/documents/IGES5-3_forDownload.pdf/view
[14] ISO 10303. (n.d.). Industrial automation systems and integration – Product data representation and exchange. *Multipart standard*
[15] Mun D. and Han S., Identification of Topological Entities and Naming Mapping for Parametric CAD Model Exchanges, *International Journal of CAD/CAM*, Vol 5 (1). 2005
[16] Coombs J. H., Renear A. H., and DeRose S. J., Markup systems and the future of scholarly text processing. *Communications of the Association for Computing Machinery*, 30:11. 933–947, 1987, issn: 0001-0782. url: http://xml.coverpages.org/coombs.html (2008-02-01).
[17] Khare R. and Rifkin A. The origin of (document) species, *Computer Networks and ISDN Systems*, 30 (1-7). 389–397, 1998. url: http://www.ifindkarma.com/attic/papers/www/origin-of-species.html (2008-02-01).
[18] ASME Y14.41 -2003, http://en.wikipedia.org/wiki/ASME_Y14.41-2003
[19] Elinson, A., Nau, D. S. and Regli, W. C., Feature-based similarity assessment of solid models. *Fourth Symposium on Solid Modeling and Applications*. New York, NY. Ed. By C. Hoffman & W. Bronsvoort. ACM Press : Atlanta, GA. 297–310, 1997.
[20] Hoffmann, C. M. and Joan-Arinyo, R., CAD and the product master model. *Computer-Aided Design* 30(11). 905–918, 1998. doi: 10.1016/S0010-4485(98)00047-5.
[21] Ding, L., Matthews, J., McMahon, C. and Mullineux, G., An Extended Product Model for

Constraint-based Redesign Applications, *International Conference on Engineering Design*, *ICED'07*, 28 – 31, August 2007, PARIS, FRANCE
[22] 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
[23] Hisarciklilar, O. and Boujut, J., An annotation-based approach to support design communication, *International Conference on Engineering Design*, ICED'07, 28-31 August 2007, PARIS, FRANCE
[24] Ding, L., Davies, D. and McMahon, C.A., The integration of Lightweight Representation and Annotation for Collaborative Design Representation, *Research in Engineering Design* (http://dx.doi.org/10.1007/s00163-008-0052-3)
[25] Consultative Committee for Space Data Systems, Reference model for an open archival information system (OAIS), Blue Book CCSDS 650.0-B-1. Also available as ISO 14721:2003. http://public.ccsds.org/publications/archive/650x0b1.pdf
[26] Adobe Systems, JavaScript for Acrobat 3D Annotations API Reference. Adobe Acrobat SDK version 8.1, 2007. http://www.adobe.com/cfusion/entitlement/index.cfm?e=acrobat_sdk

Contact: Lian Ding
University of Bath
Department of Mechanical Engineering
Bath, BA2 7AY
UK
Tel: Int +44 1225 386138
Fax: Int +44 1225 826928
Email: ld218@bath.ac.uk

Dr. Lian Ding is a researcher at the Innovative Design and Manufacturing Research Centre in Bath. She received her bachelor degree (1993) from Anhui Institute of Technology and master's degree (1997) from Southeast University in China. She became a lecturer at Southeast University after graduation, teaching a variety of courses in mechanical engineering and conducting research in mechanical engineering. She then undertook doctoral research at the University of Luton, gaining a PhD in 2003. Her current research interest is in the area of computer integrated manufacturing systems. She has published one book and over 40 refereed journal and conference papers.