THE USE OF ANNOTATION IN DESIGN REPRESENTATION

C. A. McMahon, D. Davies

University of Bath
Department of Mechanical Engineering
e-mail: c.a.mcmahon@bath.ac.uk, d.davies2@bath.ac.uk

Keywords: computer-aided design, feature-based design, annotation, mark-up

Abstract: *This paper describes the use of annotation in computer-aided design (CAD) as a mechanism for recording the viewpoints of engineering specialists concerning the semantic interpretation of elements of CAD models, and then briefly presents suggestions on how annotation of text documents may be analogous to that of CAD models. Annotation, applied through markup of the elements of boundary representation solids, allows semantic information from multiple viewpoints to be recorded in CAD part models and then used for subsequent design evaluation. Markup may be introduced into a model in different ways. Three modes are described, based on approaches to feature-based design such as design-by-features and feature recognition. These modes consider markup incorporated in the course of model creation, post-hoc markup from different perspectives, and automated markup based on the model contents, including existing markup as appropriate. Similar modes may be identified in the annotation of text documents, and similar life-cycles may be envisaged in the development of annotation in both text and CAD models.*

1. INTRODUCTION

This paper is concerned with the use of annotation in engineering design, in particular with the possibilities afforded by the incorporation of annotation into computer-based information used in design. Annotation essentially involves the attachment of notes, labels etc to the elements of the information entities used in the design process. It has been used extensively in engineering in the past. Engineering drawings are essentially Mongian projections of geometry annotated with notes, labels and dimensions. Annotations are used to indicate comments and notes on drawings and diagrams in the process of their checking and revision, sometimes called "red-lining". Reviews of drafts of text documents are marked with the reviewer's annotations, and further annotations are used by proof-readers. Annotation may be used when the artefact is in manufacture and in use to mark comments and notes on design documentation. In these regards annotation is used to support communication and collaboration, but this paper will argue that when annotation is applied in a computational environment it offers far more. In particular it will argue that annotation can be used to assist in incorporating structure into design documentation and to support knowledge discovery and design automation.

The work reported in this paper began as an exploration of whether the ideas of annotations used in documents (in the form of "mark-up" explain here) might be applied in computer-aided design (CAD), as a mechanism to allow the viewpoints of different specialists in the design process to be represented in CAD models. By using annotation, we believed that the relevance of different parts of the models to different specialists might be indicated, and that this might be used to minimise the effort in model conversion in computer-aided engineering. As we progressed, we realised that the comparison could go further – that annotation could serve analogous purposes in documents and in CAD models, and that furthermore we could identify distinct modes of use and application of annotation that were similar in each of the two domains.

In this paper we will first introduce the use of model representations in design, and explain how manipulation of the models depends on their interpretation by specialists. The need to capture the meaning of different aspects of the design models from specialist points of view has been the basis of feature-based approaches to CAD for some years. After a brief introduction to the ideas of mark-up applied to documents we will suggest how it might be applied to CAD models, in particular boundary representation solids, to annotate the models in order to capture what are in effect multiple feature-based interpretations of the design. Annotation from different viewpoints may be achieved in different ways, and an exploration of these will be used to present a model of the way annotation could be developed in CAD. Simple “proof-of-concept” experiments in this regard will be described. The final part of the paper will then explore how the ideas presented in the context of CAD may be extended to consideration of the annotation of documents through markup.

As noted above, annotation has been used in design for many years for purposes other than the labelling of model entities to make them more computer-interpretable – it has been used widely as a mechanism for collaboration and communication in the design process, for example through red-lining of physical drawings and equivalent processes for computer-based models. We believe that such annotation is not incompatible with what we are describing here, and that similar computational mechanisms could be used for the capture and storage of each type of annotation.

2. 2. USE OF MODELS IN ENGINEERING DESIGN

During the engineering design process, different engineers take information about the emerging design, and manipulate it in order to generate the new information required for the development of the product. Various engineering representations of the product are created and used for different engineering tasks. These represent, for example, manufacturing engineering viewpoints, structural and other analysis viewpoints, process engineering viewpoints and so on. These representations for the various engineering requirements are obtained by translation between engineering models, augmenting them as necessary with information specific to the specialist viewpoint [1].

The information that the participants in the engineering design process produce represents modelled properties of the emerging design. These properties may be divided into two main classes: design parameters that describe the shape, dimension, surface finish and other attributes of the product for subsequent manufacture, typically represented in drawings, diagrams and CAD models; and performance parameters that describe the characteristics and behaviour of the products subject to an external environment that applies loads and other boundary conditions to the artefact [2]. The design parameter model is what the design engineer produces, generally today in the form of a CAD model. Specialist engineers, generally using further models that are used to assist in their work – such as Finite Element and other computational models – estimate the performance parameters during the design process. The additional models, which we term auxiliary models, may be computer models, mathematical models, graphical models and even physical prototypes and test pieces. During the design process a large series of models – of specifications, constraints, loads and the various design and performance parameters – are developed in parallel, as shown in Figure 1.

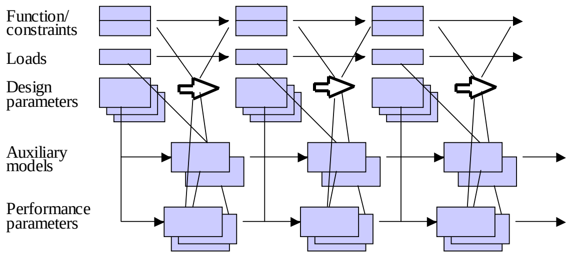

Fig. 1. *Model development in the design process* [3]

A significant issue in design is the amount of time and effnort that is spent first of all by design engineers in creating CAD models of design parameters and then by the specialist engineers in creating their auxiliary models, derived from the design parameter models, so that they may make their specialist judgements. These auxiliary models are produced by manipulating the elements of the design parameter model, usually based on the engineer’s semantic interpretation of the design model. For example, collections of faces and edges in a boundary-representation solid model may represent manufacturing features to be considered in process planning [4]. Loads and constraints may be applied to other collections of faces and edges of the same solid model for the purposes of Finite Element analysis. If suitable representations of the design parameter model are used, then it may be possible to generate specialist information (for the evaluation of performance parameters) more rapidly, thus allowing more thorough exploration of the design space. Ultimately, partial or full automation of design evaluation processes may be enabled.

An important approach in CAD is the representation of the design parameter model not as a collection of low-level faces, edges and vertices, but as a collection of higher-level features – geometric elements with some engineering meaning [5]. Features were originally used in particular in process planning, where features with a manufacturing significance were used. Significant effort was expended in this context on methods of automatically identifying features from solid models. This “feature-recognition” approach is one approach to feature-based design. Other routes are designing using features in the construction process (“design-by-features”) and manual identification of features from the CAD model (“interactive feature creation”).

The advantages of feature-based design are that (1) the designer can store in the feature model non-geometric information which is available at the design stage and can later be applied to various engineering domains, and (2) features can be used to access information associated with particular feature types during the design and design evaluation process. However, the geometric and other attributes of a component may be combined into features in a variety of ways that reflect the needs of different design and manufacturing applications at different product development phases. Different specialists may choose to use different feature sets for the same part because they assign different engineering meaning to the elements of the part. This is known as viewpoint dependency. In this regard there is a need for feature definitions, or for a method of mapping or converting features, that allows wide coverage of different viewpoints in the design process.

2.1. Representations for viewpoint-dependent models

There have been a number of approaches to the representation of multiple viewpoints in design by features, including the maintenance of multiple feature models with translation between them [6] and use of "master-models" from which viewpoint-specific models are derived [7][8]. The limitations with these approaches include the complexity in the number of product models that need to be maintained, and difficulty in maintaining multiple viewpoint models in the context of modern CAD systems, which base their representations centrally around the boundary-representation solid model. We propose that these issues can be resolved by using annotation to allow multiple features to be defined within the same CAD model, by using as a high-level representation computer-aided design models with their constituent faces and edges annotated with labels or tags to indicate view-point dependent features. In the next section we will describe some of the ways in which this annotation can be associated with CAD models, and then we will explore the application of similar ideas in text documents.

2.2. Markup

Mark-up is about adding information to an entity to identify its constituents and how that entity may be used. For, example in text documents Goldfarb [9] identifies the purposes of mark-up as "Separating the logical elements of the document; and specifying the processing functions to be performed on those elements". The first structured mark-up language was IBM's Generalized Mark-up Language, developed in the 1960s by Goldfarb, Mosher and Lorie. This led in the 1970s to the Standard Generalized Mark-up Language (SGML), which was published as an international standard in 1985.

The Hyper Text Mark-up Language (HTML) was developed from SGML and became the standard language for document sharing across the World Wide Web (WWW). In HTML, the document mark-up is principally used to indicate how elements of the document should be presented – for example as headings, tables and so on. Today, other developments from SGML, including the eXtensible Mark-up Language (XML) and the Resource Description Framework (RDF) [10], are being used as tools to allow documents to be marked-up for the purpose of making them computer interpretable. In particular, the initiative of the Semantic Web seeks to "…bring structure to the meaningful content of Web pages" [11], to allow computers to manipulate and deal with the contents of the Web on a more sophisticated (abstract) level. It is suggested here that by applying annotation to computer-aided design (CAD) models as mark-up the same may be realised in a CAD environment.

3. ANNOTATION OF CAD MODELS

As noted, geometric models in modern CAD systems are generally represented as collections of faces, edges and vertices in boundary-representation structures. Many systems in addition store information about the steps used in construction (the construction history), and allow models to be constructed from features. However, features are often used largely for the purposes of rapid construction and their data is not very accessible in the CAD model, and construction history contains little information about the engineering significance of the model. The use of technologies in CAD that are similar to the Semantic Web technologies of XML and the RDF may rectify this by allowing a higher layer of abstraction to be coded into CAD models. By using appropriate annotation, through mark-up, information relevant to manufacturing and analysis from many viewpoints may be embedded into the CAD models, because in principle multiple mark-up instances might be associated with any entity in the B-rep model. And, just as ontologies allow marked-up documents to be manipulated more intelligently by computer, so ontologies in CAD are needed to define the entities in the domain (e.g. CAD model entities and how they are used in applications such as computer-aided manufacture and finite element analysis) the relationships between entities (including pieces of process that manipulate the entities) and rules of 'inference' (process sequences) for moving between the entities.

It is argued that all three established feature creation methods outlined in section 2 (design-by-features, automatic feature recognition and interactive feature creation) have potential for CAD model annotation, and that each approach is appropriate at different stages of the design process and for different design applications. Design-by-features (DBF) may be used in a conventional way, with the entities associated with each feature being marked up as such. DBF may be used to mark-up features (the incorporation of a feature into a model would incorporate a collection of geometric elements with appropriate annotation), but may also be used to allow other annotation to be used in the course of model construction and use – for example standard parts (motors, bearings etc.) could be incorporated with mark-up to describe their attributes, and annotation might be added during manufacturing to record feedback to design.

Automatic feature recognition (AFR) may be used to process the geometric model and to label those entities corresponding to features and other meaningful parts of the model. In addition to feature recognition, the processing of an annotated model may be used to derive further information from a part with or without mark-up. For example a part marked with information about its density could be processed to compute its mass properties, and these attached as mark-up to the part. Automatic processing will allow easier expansion of the evaluations covered as the model may be passed through the processing software any number of times to identify different feature sets or other information.

Interactive feature creation (IFC) allows labels to be attached to a model by the user interactively selecting entities for mark-up. Annotation can be used to identify conventional features, and also to label entities in the part that have particular significance. It can also be used for the annotation of the CAD model for the purpose of embedding comments or opinions in the model for communication within the design process, and to record in the model the results of design evaluations carried out by the human participants in the design process.

The different approaches have different application at different stages in the design process, suggesting a life-cycle for the development of annotated part models in CAD. Initial construction of the model will be from a combination of unlabelled part geometry together with marked-up features and other elements incorporated as required in the course of construction. The model would be interactively annotated to identify important elements or aspects of the part – for example the part material or manufacturing process, or faces to be loaded or constrained – and it would be automatically annotated to record the results of the application of feature-recognition algorithms or of algorithms that derive further information by processing the part model and its existing annotation. As an example of the application of this life cycle, consider the following scenario for the stages in the creation of the information for an engineering part:

1. The initial part geometry is created from a parametric solid, with the parameters recorded as mark-up, and with additional manually created unmarked geometry. Marked-up features representing machining operations are subtracted.
2. Annotation indicating the part material is added interactively.
3. Mass property data (mass, centre-of-mass, moments of inertia) are computed based on the part volume (computed) and density obtained from an external information resource (using the material data added in (2)). This data is added as annotation to the part using mark-up.
4. Annotation indicating manufacturing processes to be employed is added interactively.
5. A process plan is created automatically or by guided manual activity, based on the machining features. The process plan data is stored in a separate information entity and a reference to this included in the part file. Cost information is derived from the process plan and added to the part as annotation to the CAD model.
6. Faces to be constrained and loaded are indicated and marked-up interactively, together with information about the load case to be applied.
7. An analysis model is created automatically or by guided manual activity, based on the mark-up created in (6) and external records of material properties and load case data.

Further interactive or automatic annotation steps are taken as required. The process of adding information in this way through the life-cycle of an information entity is in principle unlimited, and the various modes of annotation can be combined in many different ways.

3.1 Experimental exploration of annotation approaches

Experimental work is being carried out using the Unigraphics™ CADCAM software and its applications programming interface Open C++ [12]. Similar capabilities are afforded by other major CADCAM tools. Mark-up of the CAD model is achieved by associating Unigraphics' user defined objects (UDOs i.e. user-defined data structures) with elements of the B-rep. The UDOs can be attached at whatever geometric level is most appropriate – for example identifying the external boundary of a model would involve labelling all faces, the part density can be attached at the solid level, faces can be labelled as being subject to loads and manufacturing features can be generally indicated by the labelling of faces and edges (e.g. a cylindrical surface, conical surface and a centre-line for a hole). Figure 2 shows an example of annotation that might be associated with a simple part.

Annotations are added interactively to the part model using an Open C++ program. Derived mark-up, and processing of the CAD model based on the attached annotations is also applied using Open C++. In the next section three examples are given of proof-of-concept experiments in this regard. We suggest that in future independent computing agents could be used to produce the derived information. The execution of these agents would be conditional on the presence of appropriate mark-up in the CAD models and external information as required. Our current research is also exploring the use of ontologies to describe legitimate markup , and the use of process modelling languages to describe the way in which the models should be manipulated.

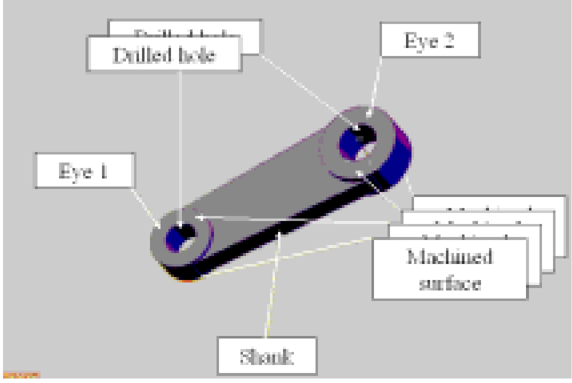

Fig. 2. *An annotated model of a link.*

4.3. Proof of Concept Experiments

In this section the use of annotation of a single artefact model as a basis for multiple downstream processes is described through three simple examples – design evaluation using FEA, generation of a manufacturing model, and automatic generation of additional markup.

Design evaluation using FEA

The task of producing and executing a finite element model for a part was chosen as the example case for producing auxiliary models from a marked-up geometric model. The NX3 software includes a structural Finite Element (FE) analysis application, which can be controlled using the API. A program was written within the API to execute an FEA process based solely on annotation attached to a model as markup – for example to indicate faces that have a significance in the analysis (e.g. to be loaded of constrained). Once the model was annotated, the FE analysis was then executed automatically, using an API program, based on the information contained in the markup. Figure 3 includes an example results visualisation from execution of the analysis. This demonstrates that markup of CAE models can be added to a model in an existing commercial CAD environment and that the information encoded in the annotation can be utilised to drive automated processes within the same environment.

Producing a derived CAD model

The task of producing the blank casting geometry for a part was chosen as the example case for producing manufacturing data from an annotated geometric model. Text labels in the form of UDOs were attached to entities in the part, indicating what kind of operation was to be used to produce them. An API program was created which searched a part file for these labels and applied processes to the entities identified by the labels. In this simple example any surface found with the label “machined surface” was extruded through a fixed distance and filleted around the edge; any feature found with the label “drilled hole” was deleted. The resulting geometry was saved as a new part file with a name that indicated the part file it was produced from and identifying it as a casting blank. This file was placed in a subfolder in the directory containing the original part file and marked with the name of the part file. The original geometry is based on that shown in Figure 2 and the derived casting blank geometry is shown in Figure 3.

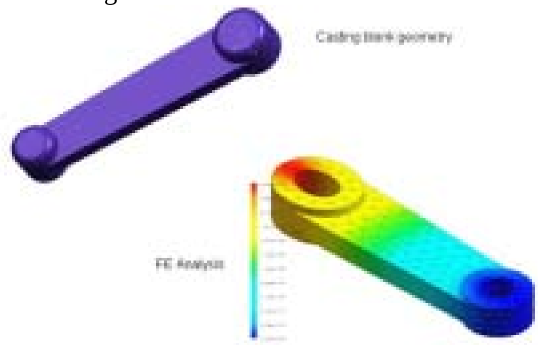

Fig. 3. *Models derived from the annotated model.*

Markup from process

An API program has been created which reads the material name in the markup of an object in the model, calculates its mass properties and then adds this information as further markup of the object.
The name of the material and its density is added interactively to the “body” in the model as markup. The user then selects an option from the menubar to trigger the program, though this could be triggered automatically using an event in the NX3 environment such as the opening or saving of a part. The program searches for markup in the part and filters it for a string identifier that indicates a markup structure containing a material name. Once this markup has been identified the density value it contains is used with a native NX3 mass property calculation function to produce the mass properties of the body. A new markup structure is then created and attached to the body in question to which are added the calculated mass properties. This new markup structure includes a text string identifying it as the container of the mass properties of the body. This demonstrates the practicality of progressive markup, i.e. new markup being generated by processing existing markup.

4. ANNOTATION OF DOCUMENTS

Our approach to the use of annotation in CAD was based on our observation that the techniques of mark-up could be transferred from a document context to a CAD context. We believe that learning can also take place in the other direction, and that there are analogies in what we have just described in the annotation of documents. We do not wish to stretch the analogy too far – we appreciate that documents and CAD models are different information entities. Nevertheless the general principles – that annotation may be created in different ways that are similar in different domains, and that there is a life cycle in creating and manipulation of this annotation – may be helpful in informing the way annotation is used.

The annotation of text documents that is being considered here involves the addition to plain text of XML tags or RDF triplets. In this context it is suggested that the three modes of annotation seen in CAD have the following broad equivalents in document mark-up:

1. Equivalent to design-by-features is the creation of a document from a template or by incorporation of standard paragraphs or glossary elements. The markup would either be incorporated in the template, or in the standard element. A simple form is to an extent analogous to a parametric part.
2. Equivalent to automatic feature recognition is the generation of document annotation by processing the document in some way. Examples of this are the use of a constraint-based (rule-based) classifier to identify classification categories for a document that are incorporated into the document by means of metadata, or incorporation of complex computer document attributes such as readability or judgements of value into a document.
3. Equivalent to interactive feature creation are the manual addition of metadata to a document, and manual markup of document elements with RDF.

Table 1 summarises the comparison of annotation approaches in the two domains.

As an example of a document life-cycle equivalent to that presented for a CAD model, consider the way in which annotation could develop in a report of a design analysis activity:

1. Initial creation of the report based on a standard report template. This initial population might be through a wizard-driven process or through a process that constrains the form to be completed in a particular way. Standard paragraphs might be incorporated, annotated with metadata, to describe the (standard) methodology employed in the work, the software tools used and the material and manufacturing data.
2. The design analyst adds further text, describing the results of the analysis and a discussion of these results. Annotation is added interactively indicating lessons learned and key results from the analysis.
3. A constraint- (rule-) based classification system is used to identify classification categories for the text file, and these are added as markup to the file if not already included in the existing metadata.
4. A senior engineer adds further annotation indicating a commentary on the results of the analysis.
5. A "value" rating is computed for the file, based on the importance of the analysis, the seniority of those who have contributed and approved the work and other related information. This is added as metadata in the file.
6. Further interactive or automatic mark-up steps as required – for example at a later date the part that has been analysed is tested, and annotation is created to indicate the results of the test.

Again, the process of adding information in this way through the life-cycle of an information entity is in principle unlimited, and the various modes of mark-up can be combined in many different ways. Capturing information in this way locally to the design artefact provides a basis for information completeness control and for design dependency and rationale capture and error tracking.

Table 1. *A comparison of annotation approaches in text documents and CAD models.*

Annotation Mode	**Application in text documents**	**Application in CAD models**
Incorporate mark-up during model/ document creation	Use of template-driven documents; incorporation of standard text fragments (e.g. paragraphs) in documents	Design by features; incorporation of standard patterns and design elements
Interactive mark-up	Post-creation addition of metadata. Interactive mark-up of document text e.g. against an ontology	Labelling/mark-up of model elements
Automatic mark-up	Generation of metadata through application of a classification scheme; generation and embedding of document readability measures etc.	Generation and embedding of derived information (e.g. cost estimates, mass property data)

5. CONCLUDING REMARKS

In this paper we have presented an overview of an approach to annotation in computer-aided design in which markup is added to the elements of a geometric model to indicate their engineering significance, and we have suggested how the approach has analogies in the annotation of text documents. We suggest that by combining the two approaches it may be possible to achieve an integrated and coherent approach to the incremental development of engineering information in design that addresses the long-standing issue of viewpoint dependency.

Acknowledgements

The authors gratefully acknowledge the UK Engineering and Physical Sciences Research Council for their support for Mr Davies via a doctoral studentship. The authors are both part of the Bath Engineering Innovative Manufacturing Research Centre, funded by the UK Engineering and Physical Sciences Research Council.

References

[1] MG-IT Coordonnateur: Lèon, J.C., Contribution to a multi-views, multi-representations design framework applied to a preliminary design phase, In J-L Batoz and P Orsero (Eds) *IDMME '98, 2nd International Conference on Integrated Design and Manufacturing in Mechanical Engineering,* Compiègne, France, May 27-29, 1998

[2] Suh, N. P.. *The Principles of Design*, New York, Oxford University Press, 1990

[3] McMahon, C A, Meng, X, Brown, K N and Sims Williams, J H, A parallel multi-attribute transformation model of design, In A C Ward (Ed) *Proc 7th Int. Conf. Design Theory and Methodology*, ASME Design Engineering Technical Conferences, Boston, MA, 17-21 September, DE-Vol 83, 2, 341-350, 1995

[4] McMahon C A and Browne J, *CADCAM: Principles, Practice and Manufacturing Management*, 2nd edition, Wokingham, Addison Wesley Longman, 1998.

[5] Shah J J and Mäntylä, M, *Parametric and feature-based CAD/CAM: Concepts, Techniques, Applications*, New York, John Wiley and Sons, Inc. 1995

[6] Jha, K. and Gurumoorthy, B., Automatic propagation of feature modification across domains, *Computer-Aided Design*, 32, 691-706, 2000

[7] Hoffmann, C.M., and Joan-Ariyo, R., Distributed maintenance of multiple product views, *Computer-Aided Design*, 32, 421-431, 2000.

[8] Kugathasan, P and McMahon, C A, Multiple viewpoint design models for automotive body-in-white design, *International Journal of Production Research*, 39, 1689 – 1705, 2001

[9] Goldfarb, C F, *The SGML Handbook*, Oxford, Clarendon Press, 1990

[10] Berners-Lee, T., Hendler, J. and Lassila, O., W3C, *Resource Description Framework (RDF) Model and Syntax Specification*, http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/, 1999

[11] *The Semantic Web*, Scientific American, 284, 34-43, 2001

[12] UGS, *UGS: Products: NX: Programming & Customization: Application Programming Interfaces (APIs)*, http://www.ugs.com/products/nx/prog_custom/api.shtml, 2005