Unifying product definition and standards representation with a specification-orientated organisation of design information

PAT TOMS
Consultant
14 Grosvenor Terrace
Glasgow G12 OTB Scotland UK

ABSTRACT

This paper considers two apparently disparate problems to have the same conceptual issue at root. Firstly, the constitution of a model for building design standards and codes, from which different computer representations can be derived. Secondly, the content of a Generic Product Data Model, as a neutral representation on which to base the implementation of any data exchange technique. A specification-orientated organisation of design information is proposed as a basis for a unified approach to these models. The meaning of the concept of specification is considered in terms of representation as well as content, and the concept of a product in relation to specification. The extent to which computer object-orientated database and programming techniques can model specification-orientated information handling in design is considered. The scope of STEP is discussed in this context. Some features of a standard for the creation and use of specifications are outlined as computer database requirements for handling design information.
Keywords
design; product data model; standard; object-orientated; specification-orientated

Introduction

Interactive software is being developed by different sectors of the construction industry to process design information for different purposes. In particular, a model for building design standards and codes is sought from which various computer representations can be derived [CIB 1992]. At the same time a Generic Product Data Model to facilitate data exchange is being developed for STEP [STEP 1992, Danner 1992]. This paper proposes a unified approach to product definition and standards representation.

Unification is based on recognition of the links formed between the two different types of information when undertaking design check tasks, described previously (figure 1) [Toms 1988 and 1992]. Formulation and revision of standards, and exchange of product data may be regarded as discrete operations. But in the context of a design check the information being processed is related. As is information handled in other types of process, such as checks on the adequacy of design configuration and location, or routine selection and optimization of the configuration, each of which concern some of the information in figure 1.

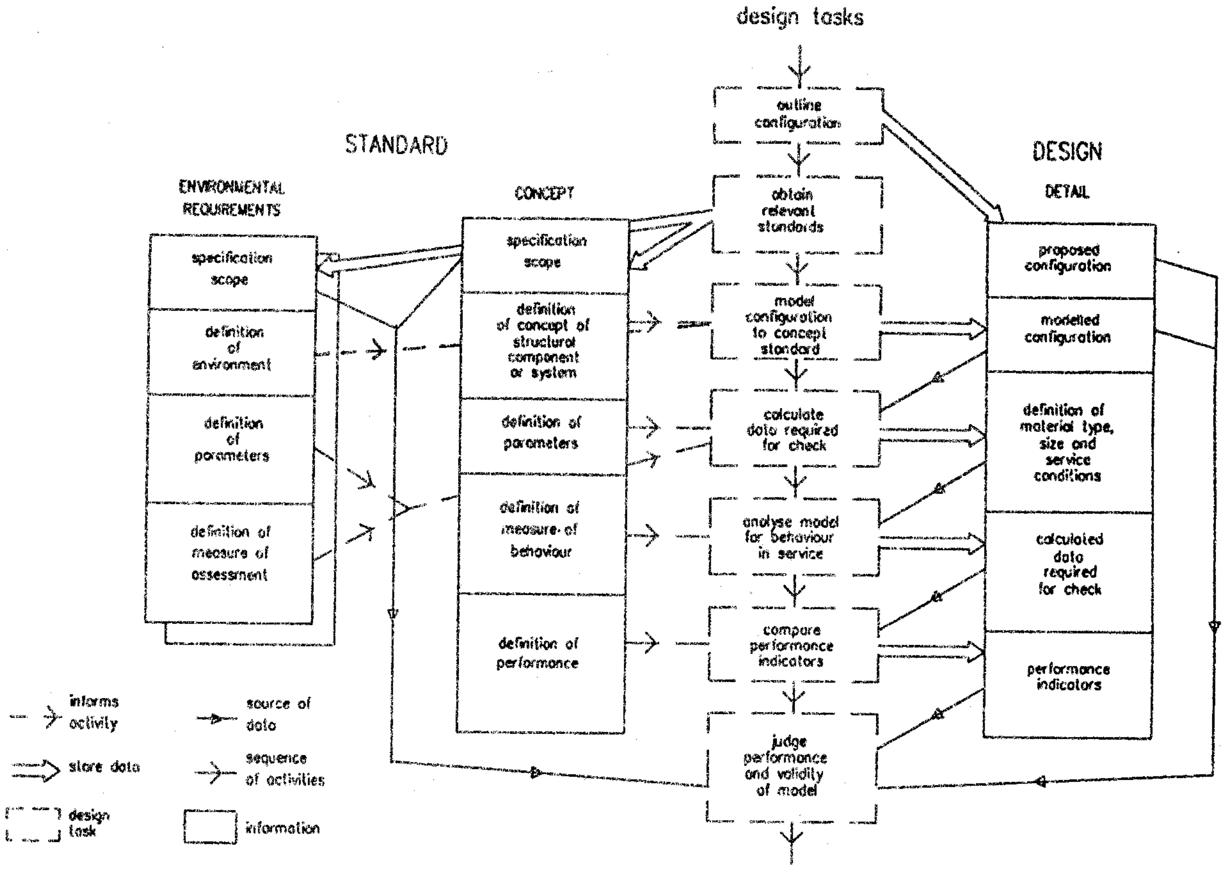

Figure 1 Activities and information involved in a design check.

All construction activity uses products in processes to satisfy particular requirements. All aspects of a products existence can be described in terms of specifications. Therefore, all design information can be categorized in terms of specifications. Both product definition and standards representation are regarded here as problems of specification representation. For a specifiation-orientated organisation of design information, an improved understanding and definition of specification is sought. This requires elaboration of the conceptual relationship between product and standard specification.

This paper argues that the function of a specification is not only to provide information, but also provide a representation that facilitates efficient designer interaction for design assessment. A specification exists to be compared to other specification information. Presentation of text and graphics in paper or computer-based specifications vary to suit different design tasks. A specification-orientated organisation of design information for computerization needs to be based on standardization of both specification content and representation. Research into the logical structure of information in standards provisions, such as SASE, being developed using object-orientated modelling, concerns data handling rather than representation [Fenves 1987, Garrett 1992].

Appropriate categorization of document data and effective manipulation techniques are needed for word, graphics and spreadsheet processors. Similarly, computer handling of specifications as product representation requires suitable

* definitions of specification information to enable any design information to be categorized in terms of specifications, and
* data handling techniques to peruse specification contents and assemble them into a product specification (ie product representation and definition).

Categorization of design information should not be concerned with product type, but generic characteristics of product form. The form of building products (whether a building or components) need to be described in generic terms using an appropriate concept of specification. Product definition and standards representation are considered here using concepts outlined previously [Toms 1988 and 1992]. These include

* a building product is a process
* design information should be categorized in terms of both product specification and design process
* specification types are grid, environmental requirements, concept and material
* specification content comprises five categories of design information (scope, definition, parameters, measure of behaviour, performance indicators)
* for product representation specifications are given locations in space
* symbols, often lines annotated with text, are used in a design sketch to represent proposed specifications as a design concept
* specification information is associated with the symbols.

Product characteristics and the meaning of a specification are considered in this paper. Comment is made on some current object-orientated approaches to handling design information from a specification-oriented viewpoint. Object-orientated concepts are related to specification-orientated techniques. Some semantics of product definition and application protocol outlined in STEP [STEP 1991,1992] are considered in relation to concepts of product and standard presented here. Finally, some requirements of a standard for the creation and use of specifications are outlined. As the basis of computer database requirements, it has to go beyond requirements for computer data exchange, to include requirements for designer-computer interaction.

Representing the product

To facilitate use of a product, characteristics are represented in terms of specifications. Essentially, a product has two states of existence: transition and utilisation (figure 2). An asset has been distinguished from a product [Toms 1991]. An asset is owned and has a lifecycle described as a succession of products, each produced by a process such as construction, maintenance or alteration. STEP could usefully adopt this distinction between product and asset.

During the state of transition a product is stored, transported and incorporated into a process. During utilisation a product performs. For purposes of transition the arrangement of material, as mass, is a paramount consideration. For the purpose of

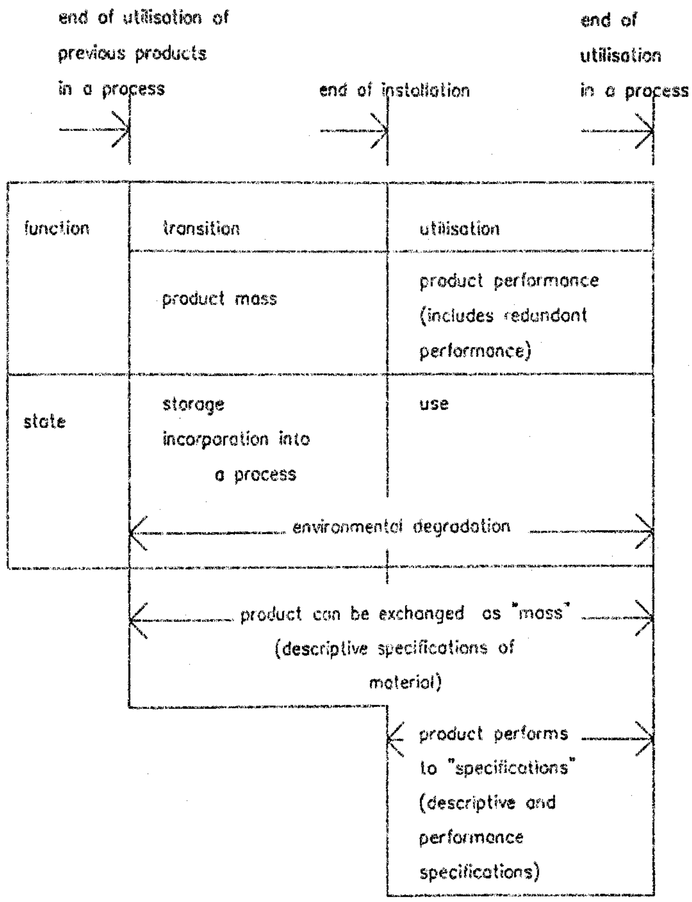

Figure 2 Product life cycle: two states of existence.

utilisation the product is considered to embody specifications, governing both description and performance. The concept of a building comprising compatible processes can be used to unify information representing the transition of products into structural form and utilisation of structure as an activities environment [Toms 1991].

Product concept is described in terms of descriptive and performance specifications. During design a set of appropriate product specifications are selected to suit chosen specifications for environmental requirement and concepts. Conformity is checked by comparing design configuration to specifications adopted as standards (figure 1). Descriptive specifications can be explicit and quantitative (such as dimension or mass). Conformity to a performance specification, however, has to be described in terms of the comparison of design configuration and standards. It is the generic definition and classification of the information required to represent this comparison that constitutes a key problem of product representation.

Applying object-orientated techniques

product model permits "inquiry and modification" and a process model defines certain activities operating on the product model.

In contrast, this paper argues such a separation is inadequate as

* a product representation cannot be understood without an integral interpretive process.

A single conceptual framework of specification handling is sought as the basis of one model, to subsume both product model and standards representation. Figure 1 shows how a design configuration (the product) cannot be assessed without reference to standards. Thus, interpretation of a drawing assumes an understanding of standards of mass and dimension, and an interpretive process, such as use of a ruler, to facilitate comparison.

Rigopolous and Oppenheim consider design synthesis to be "the assembly of substructure and elements" [Rigopolous 1992]. A structure model is created and updated by instantiating Design Objects associated with 'local knowledge', and connecting them. Design objects are instantiated from existing classes: "the engineer ... instantiates the appropriate number of objects (floor panels, planes of support, and so on) to assemble the model". Design objects are described as essentially structural components. In contrast, it has been argued that environmental requirement and concept specifications are instantiated first, then specifications of design material and layout [Toms 1991]. The product itself is the representation of all required specifications, not simply material (see above).

Howard has described a hierarchy of information for design synthesis, with primitive classes being assembled into composite classes [Howard 1992]. "A beam combination of specific forms, functions and behaviours" is a composite object. "Using composite classes, an engineer can assemble special purpose objects without being constrained to the list anticipated in a data standard". However, the contents of the primitive schema are not classified and include different types of information: material, cost, scheduling, stress, load, location, dimension, etc. Sause identifies a need to research into "design product abstractions" [Sause 1992]. This paper considers design information in terms of specification and seeks improved understanding of the concept.

"Correct automated interpretation of the standard" is a key objective of an object-orientated modelling of standards [Garrett 1992]. Four types of information have been identified: design object heterarchy (shape, function, material), data item hierarchy, performance limitation hierarchy (strength, serviceability) and data item instance network (requirements). Whether such objects can facilitate interactive design, rather than automated interpretation, requires further investigation. Particularly the creation of sets of specifications rather than nested objects.

What constitutes the meaning of a specification?

A specification is a category of information used to take decisions about the definition and control of the productive processes of product transition and utilisation. It is not a rule

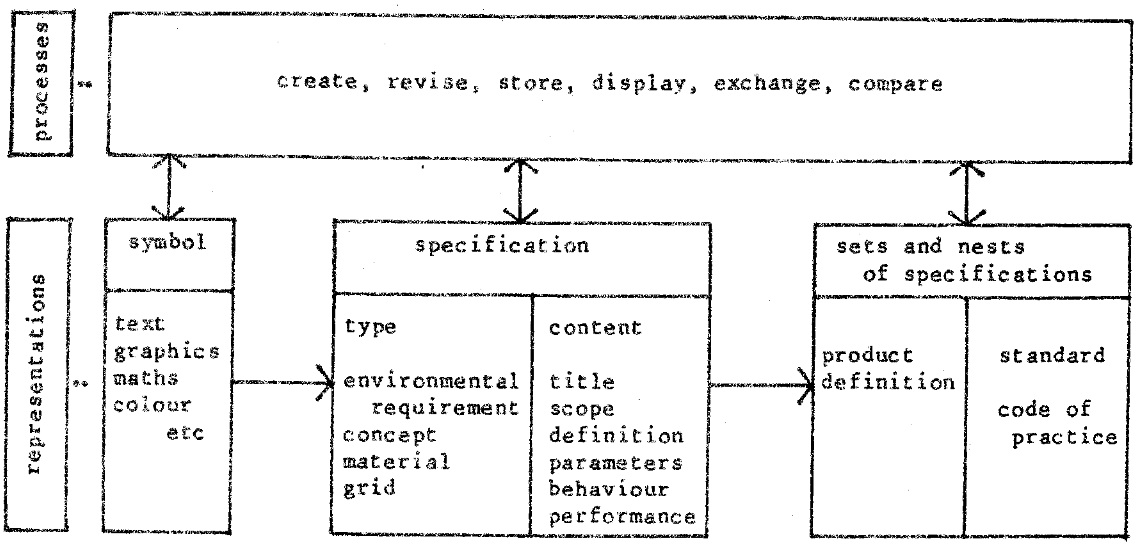

Figure 3 The meaning of specification:
symbolic representation and processing.

[Toms 1988 and 1992], though information from it may be used in a routine fashion. A specification is used to define a product for consumption or as a standard for process control (figure 3).

Specifications are assembled into further specifications, as sets or nests. A set declares the association of selected specifications each located in space [Toms 1992]. Symbolic representation of a specification with reference (perhaps implicitly) to other explanatory specifications each having a related means of processing, enables meaning to be given to a specification (figure 3). Reference may be given to a specification title or a graphical representation annotated with text on a design concept sketch. Standard symbols are used in AEC graphics software for standard products. However, for a design sketch of a unique structure a structural engineer has to devise symbols for project-specific specifications [Toms 1992].

A specification is referenced by title and contains five necessary categories of information for use [Toms 1988 and 1992]. Even a ruler, a standard specification for length, contains information in these five categories. As a convenient mobile graphical presentation, it contains lines on plastic annotated with text for units and numbers for lengths. Its title and scope may be implicit ("a ruler"). Definitions of length are contained in other standards referenced by "mm" or "inch". Its mobility permits manipulation and comparison of lengths on drawings or building products, from which comparative performance indicators are derived.

Thus, the meaning of a specification is established with both information content and a useful representation. Computer representation of specifications must provide not only content (and associated logic processing) but also techniques for designer interaction with information. Identification of generic processing techniques will enable any specification to be incorporated and processed routinely, to create unique designs.

Limitation of object-orientated techniques

Object-orientated programming techniques were devised for efficient computer processing of data. Programming and computer operation may be more efficiently implemented using OOP, but the relationship of the computer software user to the screen output does not change. It is the sophistication of this relationship which enables specifications to be handled by a designer to create unique designs.

For the limited problem of modelling structural systems as connected components, Rigapoulos and Oppenheim identify limitations of OOP and have suggested additional functionality [Rigapoulos 1992]. "Multiple inheritance is too liberal in jointly bequeathing the behaviour of all superclasses to the subclasses". An additional message handling technique "delegation to connectors" is proposed to cut across the tree structure of class hierarchy, which is "unacceptably restrictive".

Each instance of a class of objects has a unique identity, but has the same set of data properties and responds to the same set of methods as the class. For a specification, content can mean something different depending on the particular context in which a designer uses it (how else is a unique design achieved?). A specification is not a rule [Toms 1988]. So, can a specification be regarded as an object? Hardly, as it cannot be understood without an interpretive process (as discussed above). Why not include the interpretive process in the object? This cannot be done, as it can vary from project to project. Only certain aspects of the use of a specification can be handled as objects.

For software programming, an object is conceptualised to deliver data according to predetermined functions, on receipt of a message. A specification exists to be scanned and compared by the designer (not the computer), then referenced and associated with other specifications. Information from a specification is often modified or qualified before being associated with other specifications as sets or nests to create unique designs, using criteria not held by the computer. Only some information can be processed by the computer as objects. Where standard configurations of material have been selected to suit standard requirements, certain specification information may be associated with a design without modification. However, every design has unique characteristics which makes generic handling of design information by computer problemmatical.

Can a product be regarded as containing specifications? No, it conforms to specifications, for which a comparative assessment process has to be declared (perhaps implicitly for simple specifications such as length). Description of products as objects inheriting specifications as objects is not sophisticated enough. Effective specification-handling depends on designer interaction. It is not a question of object inheritance but application, where application involves comparing chosen specification information to a design configuration, which may be altered to suit, then checking appropriateness. A specification cannot be regarded simply as an object which delivers data on command, but as information available for modification and use, in a particular design application.

Trying to extend OOP concepts and techniques, within an OOP framework is inappropriate. New concepts are needed to represent generic interactive representation and processing of design information as specifications. Some design information can be handled in terms of object-orientated techniques, such as nests of specifications. Association of specifications into sets to represent products requires specification-orientated concepts. A specification-orientated approach is needed to model design synthesis and standards processing. Object-orientated techniques can be regarded as a subset of more sophisticated specification-orientated information handling techniques.

What does STEP provide?

STEP purports to support exchange of product information throughout the lifecycle of a product (part 1 clause 3.1) [STEP 1992]. Part 1 clause 2.1.28 defines product as "a thing or substance produced by a natural or artificial process". A draft of part 41, clause 4.6.1, gives another as "a thing that is, or is intended to be, produced or consumed in an industrial process" [STEP 1991]. Neither explicitly acknowledges specification with respect to a process, say,

* "a thing specified in association with a process".

Whilst dictionary definitions are vague, "thing" must include a process as a type of product. The definition of product in STEP part 41 concerns material parts and their relationships only. Requirement and concept specifications are not considered. In STEP the concept of a product is confused with an asset (their distinction has been discussed above).

The restricted definition of product matches the limited scope of the STEP standard. It deals with descriptive aspects of product only, such as shape, and management of versions. It envisages two levels of resources, generic and application, and does not provide a conceptual basis for defining and handling all types of specification information (such as definition of product behaviour using mathematical relationships). Instead of treating specifications as an integral feature of a product, an External Specification Schema "enables links to be established between aspects of product data and references to particular formal standards or documents that are outside the domain of ISO 10303" (part 41 clause 7.1). The schema is limited in scope

specification-type
specification
specification-usage-context
specification-relationship,

to enable reference to non-STEP specifications only. Examples given of "specification-type" are descriptive only ("material, surface finish"). "Specification-relationship" enables two specifications to be associated, but how is not elaborated, merely being "defined in the application protocol". If the character of specification information, including sets and nests, were addressed the basic distinction between generic and application resources (specifications) would not be critical. It would not matter whether a resource (specification) declared was inside or outside the standard, as its information content could be categorized and handled in a standard way. No comment is made on definition of the term "specification". Inadequate definition of specification in other standards has been discussed previously, and improvement proposed [Toms 1988].

In STEP an Integrated Resource provides "representation of product information" (part 1 clause 3.2.1) and is currently limited to shape and some information management issues. An Application Protocol "specifies the representation of product information for one or more applications" (clause 3.2). Integrated Resources are defined and contained in the standard, each having the characteristics of a specification as discussed in this paper. APs are to be defined in other ISO standards. They define sets and nests of STEP Integrated Resources (specifications) for applications. An application is defined as "a group of one or more processes creating or using product data" (part 1, clause 2.1.4) and relates to computer information processing, not processes as products. An AP describes the use of integrated resources for a specific application context (part 1, clause 2.1.9). Thus additional activity models can be defined, but using STEP resources only.

The conceptual basis for organisation of resources and APs is explained in STEP in terms of the assembly of parts of products, rather than specifications. APs are regarded in this paper as a particular type of specification giving "definitions of scope, context and information requirements of a [computer data exchange] application" (STEP part 1 clause 7.1),

Application protocols were introduced into STEP fairly recently as

"* clauses were at a different stage of maturity
* it was clearly inappropriate for CAD vendors to implement all of the standard and uncontrolled 'subsets' of the whole" [Owen 1990].

In addition, it could be argued that a standard intended to represent any product, made up of nested Resources (specifications) could potentially include every product-related specification in existence. Each would have to be drafted and approved for STEP as an AP. Introduction of APs into STEP can be regarded as recognition of the nested character of different standard specifications making up a product definition. Now it seems appropriate to propose a general standard framework for handling specifications.

A proposed Generic Product Data Model reflects deficiencies in the concept of product discussed above (see figure 4 reproduced from [Danner 1992]). It is constructed around a hierarchy of information representing "product definition", with "context" above and "property definition and representation" below. Data types are linked without reference to decision-making processes and provide a limited framework for representation of all product data. It is intended to be independent of implementation methods and does not indicate how project-specific modifications of standards, including STEP Integrated Resources modified using EXPRESS, are held.

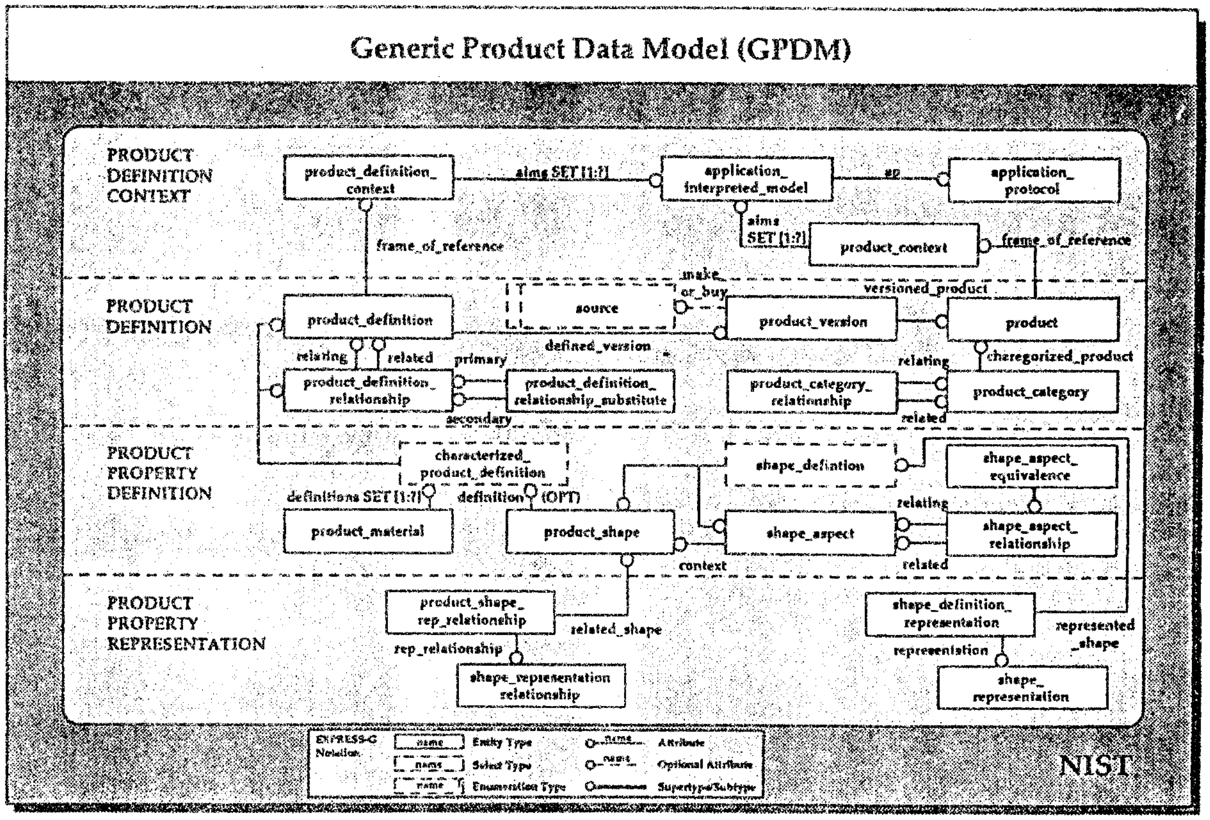

Figure 4 STEP Generic Product Data Model reproduced from [Danner 1992].

Alternatively, data could be classified in a specification-orientated product data model in terms of standard and design product specifications separated by a design task interface (figure 5). Generic links need to be made to design task processes (compare figure 1). On the standards side, ISO, national and industrial standards, Integrated Resources in STEP and Application Protocols all require categorization as different types of standard specification in terms of both product specifications and design process.

DESIGN TASK INTERFACE

Standards	Product Design Specifications
ISO, national and industrial (external) STEP Integrated Resources Applications Protocols	represented by sets and nests of selected standards plus project-specific specifications
categorized as sets and nests of standard specifications	with context of use and particular parameter values.

Figure 5 Product data as specifications related to design processes

On the product design side, information selected from standards and project-specific specifications, with design assumptions and parameter values, require to be classified similarly in terms of specifications.

Design information can be handled in a specification-orientated, not product-based, manner. This requires improved definition of specification and product as discussed above. It seems necessary to outline requirements for the creation and use of specifications. STEP data exchange resources can then be presented in a specification-orientated context.

A standard for the creation and use of specifications

The standard is required to facilitate computer classification and assembly of specifications into product representations. It would not be concerned with particular product types or models. It is required to assemble specifications into further specifications to represent a product (figure 3) for

* consumption (product definition) or
* use as a standard or code of practice (for control of a productive process).

Task control should provide interactive access to and control of processes for

* creation and combination of symbols (text, graphical shapes (as in STEP), mathematical, etc) each with an associated specification
* combination of symbols to represent specifications, each able to be given a location in space
* assembly and use of specifications into sets and nests to represent products.

This requires classification of types of symbols and specification information, and generic processing requirements. Design information is to be categorized in terms of product specifications and the design process. Basic processes facilitate design tasks linked sequentially in figure 1. The standard will not describe particular design checks or routine processes for product configuration, such as shape optimization. The following needs to be included.

Definition of product and specification. Definition of product types

* material, space and compatible process.

Define product representation as

* a specification set that includes grid, environmental requirement, concept and material specifications, where
* a concept specification set includes specifications for compatability, integrity, functional performance and cost.

Categories of symbol and associated process requirements to

* devise and combine symbols representing specification information
* select symbols for display
* exchange and reproduce symbols at other locations

(eg text, hypertext, graphical shapes, mathematical equations, photographs, etc).

Classification of specifications, and definition of generic information content in terms of product specification and design process

* type: grid, environmental requirement, design concept and material (includes STEP resources)
* descriptive, performance or both
* type of concept specification: compatability, integrity, functional performance or cost
* design process: scope, concept, parameter, measure of behaviour, performance indicator.

Generic requirements of processes to handle these categories of information and associate them with symbols representing a specification [Toms 1992]

* using the categories of symbols defined previously (eg to produce tables, layout of computations, diagrams, annotated photographs)
* spacial association using graphical representation.

Requirements of interactive processes to handle both content and representation of specifications, to represent products

* as sets, nests and combinations of specification
* enabling allocation of a title and graphical symbol to an assembled specification.

Categories of standard and product specifications require to be held such that they may be

* viewed and revised
* compared to any other information.

Summary

A specification-orientated organisation of design information is proposed to unify representation of design standards and product data models. A definition of product is given which acknowledges the specification necessary to associate a product to a process. The meaning of a specification is considered to constitute both information content, a useful representation and an associated interpretive process. Only certain aspects of specification representation can be handled in terms of objects. Interpretive aspects require a specification-orientated approach, in which object-orientated techniques are considered to be a sub-set.

The limited scope of STEP stems from an inadequate definition of product. The concept of specification is not adequately defined. An improved Generic Product Data Model can be based on specification, rather than product, classification. Proposed contents of a standard for the creation and use of specifications to represent products include requirements of generic specification processing tasks, as well as information classification.

References

CIB (1992), Task group objectives Joint CIB Workshops, Montreal, Canada

Danner, WF, Yang, Y and Reed, K (1992), STEP Integration and interpretation methods, Joint CIB Workshops, Montreal, Canada

Fenves, SJ et al (1987), Introduction to SASE: Standards analysis, Synthesis and Expression, National Bureau of Standards, Gaithersburg MD, USA

Garrett, JH and Hakim, MM (1992), Object-orientated model of engineering design standards, ASCE Journal of Computing in Civil Engineering, Vol 6, no 3, pp 323-347

Howard, HC, Abdalla, JA and Douglas Phan, DH (1992), Primitive-composite approach for structural data modelling, Journal of Computing in Civil Engineering, Vol 6 No 1, pp19-39

Owen, J and Woodall, A (1990), The current status of STEP Application Protocols, CADDETC University of Leeds, UK, June

Rigopoulos, DR and Oppenheim, IJ (1992), Intelligent objects for synthesis of structural systems, ASCE Journal of Computing in Civil Engineering, Vol 6, no 3, pp 266-281

Sause, R, Martini, K and Powell, G (1992), Object-orientated approaches for integrated engineering design systems, ASCE Journal of Computing in Civil Engineering, Vol 6, no 3, pp 248-265

STEP ISO 10303-1 Part 1 (1992), Overview and fundamental principles, International Organisation for Standardisation, Geneva, draft September

STEP ISO 10303-41 Part 41 (1991), Fundamentals of product description and support, International Organisation for Standardisation, draft version 5, Geneva, August

Toms, P (1988) Standards and the design process, Design Studies, Vol 9 No 2, pp115-122

Toms, P (1991) Defining structure in terms of standard specifications, Journal of Engineering Design, Vol 2, No 2, pp151-162

Toms, P (1992), Categorization of design information in terms of design process and product specifications for a standard for computer representation of specifications, Joint CIB Workshop, Montreal, Canada.