Practical Problems in the Development of Product Models

A S WATSON, A J CROWLEY, A D BOYLE and G M KNOWLES*

ABSTRACT

This paper is concerned with the practical difficulties encountered when developing Product Models for information exchange within the construction industry. It is based on experiences gained since 1987 from involvement in the Pan-European Eureka CIMSTEEL Project (within which a model for constructional steelwork is being developed) and from a more recent national research project (CI-PM) concerned with Product Models and construction (within which a reinforced concrete model will be developed). The paper addresses the questions of what a Product Model is, and what its role is in the management of information at a project level. The paper then moves on to consider the practical problems faced by the developers of such models; considering such aspects as the industrial context, modelling methodologies and software tools, differing viewpoints and validation. The authors highlight areas where progress in addressing these problems is being made.

Key Words

product model; management of information; construction industry

INTRODUCTION

Researchers in the construction sector are increasingly using the term "Product Model" (PM), but they do not always use this term with the same meaning. The intuitive interpretation - a structured representation of some engineering reality - is valid, but is some distance from the formal meaning. The term originated in the field of data exchange where the aim is to facilitate the transfer of engineering information between different applications software. Existing data exchange standards define exchange file formats (such as DXF and IGES) which enable users to transfer data between CAD systems provided that they have the necessary export and import "translators". These first generation standards are primarily concerned with the transfer of graphical representations, rather than the transfer of the underlying engineering intent.

The concept of a PM evolved among those researching future data exchange standards. These standards are intended to enable application software to share engineering information, not just the graphical representations. In this context the following description is applicable:

* *Computer-Aided Engineering Group, Department of Civil Engineering, Leeds University, Leeds LS2 9JT, UK.*

A Product Model is concerned with standardising the way engineering information is held, to facilitate the unambiguous transfer of such information between computer systems. It is a formal "information model" of how the totality of information, concerning a particular type of artefact, can be coherently represented to facilitate information sharing.

The use of the word "product", which is not normally applied in a construction context, underlines the fact that these ideas evolved in other sectors of engineering. The concern is with how data is held when it is *between* applications software (and not, necessarily, how it should be held within a given application). The words "data" and "information" can be regarded as interchangeable since the PM gives the data being exchanged an unambiguous context; and information is simply data with a meaning. And finally, the scope of a PM potentially spans the whole life-cycle of the type of artefact in question.

We can regard a PM as a specification for information structures with little intrinsic value. However, agreed PMs will play a key role in enabling a more integrated approach to engineering by facilitating the sharing of information between different computer systems. In this paper we briefly look at this wider context before looking at some of the practical problems encountered when developing PMs for the construction industry.

THE CONTEXT PROVIDED BY ISO/STEP

Launched by ISO in 1983, ISO/STEP (Mason, 1992) is a very ambitious long-term project to develop co-ordinated second-generation data exchange and information sharing standards that span all sectors of engineering. STEP envisages that engineering information will be transferred between application software (and thus between the users of that software) by the use of STEP format data exchanges files and import/export translators. It also envisages that information will be shared more intimately among groups of applications linked to a STEP compliant database. Whether the applications are to be "interfaced" or "integrated", STEP seeks to provide a PM based standard plus vendor independent technology for implementing the data sharing. Thus, STEP should provide engineering with an open architecture approach to information technology.

Although STEP seeks to establish a suite of compatible PMs, which address all the needs of engineering, the term PM is no longer formally used within STEP. Instead, STEP is establishing "Integrated Resource Models" (IRMs) - both Generic and Application - from which particular "Application Protocols" (APs) will be derived. This terminology reflects the way in which STEP is addressing the problem of compatibility between different PMs. An AP formally defines how STEP should be applied in a given context. One of the components of an AP is an agreed PM for that particular domain. This

PM is specified in a data definition language called EXPRESS (Spiby, 1992) which, being computer parsable and human readable, forms the cornerstone of the STEP implementation technology.

Put simply, the EXPRESS form of the PM defines the structure of the data exchanged between applications. The EXPRESS form can also be used to generate much of the code needed to build the required data exchange translators. Similarly, it can be used to generate the database schema needed to "integrate" applications and configure the STEP Data Access Interface (SDAI), (Fowler, 1992). The STEP technology required to help the implementation of information sharing is still immature, but this technology is now becoming commercially available. Because the market for this technology goes beyond the construction industry, rapid evolution is anticipated.

It is expected that the initial version of ISO/STEP will be published as an incremental standard during 1993. Unfortunately, this version will not contain any APs, or IRMs, that are specifically related to the construction industry. It is against this background that we turn to some of the problems encountered in developing PMs.

THE CONSTRUCTION INDUSTRY CONTEXT

The construction industry as a whole is fragmented, with many players having transient project based relationships. The complexity of the processes requires good information flows for effective working. Diverse application software is increasingly being deployed within those processes, while the end product, typically, has considerable value and can be extremely complex. The need for product information standards is evident.

However, most product information is still conveyed by traditional paper documents. The use of exchange files to transfer drawings between CAD systems is growing but, as yet, the construction industry is not a major information technology user. This is not the case in several other sectors of engineering where significant use is made of the first-generation data exchange standards. Although, researchers have been active in product modelling for several years (*eg*, Pentilla and Bjork, 1989 and Leppanen, 1990), the term "PM" is generally unknown to those involved in construction. To date the construction industry has had little involvement in ISO/STEP and hence the absence of relevant APs or IRMs. However, this picture is beginning to change.

A number of major research projects are now established and can be expected to provide construction related inputs into STEP. These include Process Base (Model for Process Plant Structures), CAESAR Offshore (Standardization of information exchange in Offshore Projects), ATLAS (Application of advanced integrated Information Technologies in large scale engineering) and the CIMSTEEL project. Much of this paper draws on experience gained from CIMSTEEL - a Pan-European project within which a PM for constructional steelwork has been under development for several years.

*Within the CIMSTEEL project, the term Logical Product Model (**LPM**) is used, in preference to PM, to specify how information should be structured to facilitate information sharing. This convention accommodates the more natural tendency of applying the term PM to real data relating to a particular steel structure, and emphasises the fact that the LPM exists at a logical level. The group developing the LPM initially drew on ideas from the RATAS project (Bjork, 1989) and from the GARM model (Gielingh, 1988). The latter was the source of the functional unit - technical solution constructs and the generic, specific and occurrence levels that both featured in early versions of the LPM. Broader attempts were also made to draw on resources and conventions from ISO/STEP but, at the time, these proved to be of little practical value. More recent appraisals have suggested that IRMs now available (in draft form at least) may offer some useful constructs. However, the benefit of their adoption has not yet been proven.*

MODELLING METHODOLOGIES

Even given the current state of the art, developing a PM is a complex and time consuming task. There are no correct solutions; one modelling idea can be judged to be "less good" than another, but the criterion for making such a judgement is usually subjective. Because a number of people are generally involved in the development of a model, different viewpoints will arise (the representation of information is subjective). Thus when developing a PM, the aim should be to make best creative use of these differences.

The intial phase of developing a PM is to create an activity model, typically using IDEF0 (Marca, 1988), to help define the scope of the required PM and to better understand the context it is to serve. The activity model provides a representation of the activities that use product data within the area of interest, and, being a graphical representation, allows experts in the field of interest to review it.

Ideally, once the activity model has been agreed upon, the PM can be developed. Although PMs can be created directly in EXPRESS, current practice is to develop the model in one of three graphical information modelling representations: IDEF1X (Dacom, 1985), NIAM (Nijssen and Halpin, 1989), or EXPRESS_G (Spiby, 1992). The latter, which is of recent origin, provides a graphical representation of EXPRESS. Unfortunately, the current version only supports a subset of EXPRESS and has no formal modelling methodology associated with it. The other two, which were developed in the first instance to design database schema, both have associated information modelling methodologies that are mature and well understood. Thus, the PM can be progressively developed in a graphical form which can easily be assimilated by the developers and by other domain experts who may provide inputs.

Another essential element in the development of a PM is validation, both to ensure that the model is sound and unambiguously defined, and to maximise the probability that others will agree with the proposed model. Once a PM has been developed, the procedures within ISO/STEP impose further stages of international validation and integration before the model becomes part of the standard. However, it is during the evolution of a PM that its periodic review by the third parties can be most valuable. Validation is a time consuming task, which requires appropriate engineering expertise and the ability to correctly read the model. This is a major factor in favouring the use of a graphical representation of the information model.

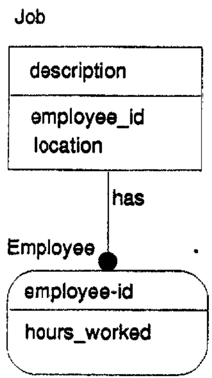
The CIMSTEEL Project opted to use IDEF0 for activity modelling and IDEF1X for information modelling. Pragmatic guidelines also govern the evolution and validation of the LPM. Within the CI-PM project the authors are currently seeking to develop modelling guidelines and procedures which could be more widely applied.

CONVERSION TO EXPRESS

Figure 1 contrasts the representation of a model in IDEF1X, NIAM and EXPRESS. Model developers face significant practical difficulties when translating their models from the two graphical representations into EXPRESS. For example, the IDEF1X methodology leads to a model in the third normal form and NIAM to the fifth normal form, while EXPRESS does not require normalisation to be applied. To better understand these difficulties a summary of the main differences between IDEF1X and EXPRESS follows:

IDEF1X has the entity, the attribute, and the relationship as its basic constructs. A relationship is a verb-phrase that connects one entity with another entity, and specifies the cardinality between those entities. Imposition of the third normal form is an intrinsic part of the overall IDEF1X modelling methodology (*ie,* many-to-many relationships must be resolved).

EXPRESS is an alpha-numeric representation that has only the entity and the attribute as its basic constructs. An entity is defined as a set of attributes, which specify a data type or an aggregation type, and a cardinality. The relationship between entities is implicit in the *entity* data type of an attribute (*ie,* Relationships are expressed as entity attributes). As there is no formal modelling methodology associated with EXPRESS, or requirements for normalisation, there is a degree of flexibility within the EXPRESS language that enables the same thing to be represented in different ways.

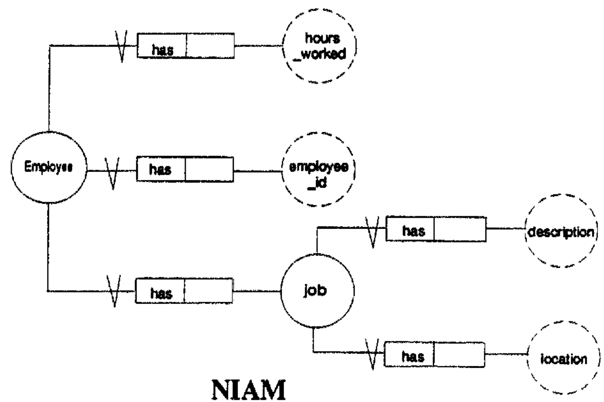


IDEF1X

```
ENTITY employee;
employee_id : STRING;
hours_worked : REAL;
has : job;
END_ENTITY;

ENTITY job;
description : STRING;
location : STRING;
END_ENTITY;
```

EXPRESS



NIAM

Figure 1. Three Alternative Representations
(Note: Due to the different Methodologies, the above representations are only approximately equal).

Table 1. A Brief Comparison of IDEF1X & EXPRESS

CONSTRUCTS	IDEF1X	EXPRESS
Entity	Entity defined by attributes.	Entity defined by attributes.
Attribute	Single valued attributes.	Single valued attributes or aggregation types.
Relationships	Expressed as *Foreign Keys* and relationship names.	Expressed as *entity* data types.
Derived Attribute	Not supported.	A function of explicit attribute(s) which appear after *DERIVE*.
Unique Identity	Key attributes which may be either *Primary Key, Alternate Key, or Foreign Key.*	All attributes which appear after *UNIQUE*.
Cardinality	Applied to relationships. Cardinalities may have values ranging from N:M, where N=1 M=>0. The inverse cardinality is always 1.	Applied to attributes. Cardinalities may have values ranging from N:M, where 0<=N<=M. The inverse cardinality may be defined within the referenced entity after *INVERSE*.
Aggregation	Not supported.	Supports aggregation types: e.g. *set, bag, list, & array*.
Data Types	Not supported.	Supports data types: e.g. *integer, real, string, entity, boolean, & defined.*
Identity Dependence	Defined by solid black relationship line. The *Primary Key* of the Child entity inherits the *Primary Key* of the Parent entity as a *Foreign Key*.	Not supported.
Existence Dependence	Child-Parent relationship with non-zero cardinality .	Non-zero cardinality of an *entity* attribute.
Categorisation	Generic entity has mutually exclusive Category entities which have discriminating attributes. Complete and in-complete categorisation supported. Categories have only one Generic entity.	Subtype Supertype constructs in which subtype has discriminating attribute(s). Subtypes are not inherently mutually exclusive, and can have several supertypes.

Table 1 illustrates some of the intrinsic differences in both the semantics and syntax of the IDEF1X and EXPRESS modelling languages. It can be seen that EXPRESS is a far more powerful modelling language than IDEF1X, in both semantics and syntax. For instance, IDEF1X does not support the notion of derivable information, rules, or data types and such information must be included as footnotes, if it is included at all. EXPRESS, on the other hand, provides the syntax and semantics required to derive attributes from other attributes, and incorporate rules and procedures within entities and schemata. Data types are strongly supported, including simple types, aggregation types, and other referenced entities. IDEF1X is intended to be used for the design of relational databases while EXPRESS does not support the notion of keys. These, and other differences can, and do, cause difficulties when translating from one modelling language to the other.

It is possible to adhere closely to the full IDEF1X methodology and to faithfully translate the resulting PM into EXPRESS. However, a better (EXPRESS) model results from a more selective application of the IDEF1X methodology. STEP provides no guidelines for translating models into EXPRESS. Indeed, although an EXPRESS Usage Guide has been written, there is no agreed style of using EXPRESS. The lack of a formal modelling methodology associated with EXPRESS is a major omission.

Within the CIMSTEEL project the interim solution to these problems was to develop of a set of IDEF1X to EXPRESS translation guidelines (de Gelder, 1991).

The EXPRESS data model is used to generate the syntactical definition of the STEP physical file, using the mappings defined in the standard. It forms a neutral computer processable conceptual schema, by which aspects of product data can be specified. EXPRESS does not provide for the definition of database formats, file formats, or transfer syntax's. Other parts of STEP provide for mappings from EXPRESS schemata to implementation forms; *eg* Part 21 (STEP Physical File), (van Maanen, 1993).

SOFTWARE TOOLS

The scale of the PM development task is such that the methodology used may be influenced by the availability of supporting software tools. IDEF1X models, for example, can be developed using graphical tools such as ModelPro by D.Appleton Company and Design/IDEF by Micromatch. Tools to help automate the generation of an EXPRESS representation are also emerging. For example, the South Bank University has a tool both to create NIAM diagrams and to output the corresponding EXPRESS. In time, such tools may be rendered obsolete by the uptake of EXPRESS_G.

The authors have recently developed a tool to automate the translation of ModelPro models from IDEF1X into EXPRESS. The development of this tool led to the development of an updated set of translation rules and the adoption of a more controlled usage of the IDEF1X methodology.

Many EXPRESS parsers exist, which are able to read EXPRESS versions of a model and check its syntax. However, a new generation of tools able to support the direct development of a PM in EXPRESS is beginning to emerge. Some, such as DECexpress by CADDETC, has the knowledge of the complex syntax of EXPRESS and can show the user what constructs may be used at each point. Others, like ExWare by Canonical Systems, allow the model to be specified *via* tables, thus avoiding the need for the user to input EXPRESS directly. Tools such as Extool by Association GOSET have the useful feature of including all the STEP IRMs in library form.

A discussion of tools that support the implementation of information sharing (Boyle, 1993) lies beyond the scope of this paper. However, a notable problem with many translator development tools, is their deficiency in supporting some of the more complex EXPRESS constructs, *eg* INVERSE and the WHERE rules. That the currently available implementation technology is deficient is a reflection of the relative infancy of STEP and EXPRESS. The development, testing, validation, and the subsequent exploitation of PMs may be inhibited by these deficiencies for some time.

WIDER ISSUES

Once a valid PM has been defined in EXPRESS it is relatively easy to use this specification to implement information exchange between selected applications. For example, we are currently using the LPM to implement such exchanges within the CIMSTEEL Project (using the CAESAR Systems toolkit, which is based on the emerging SDAI specification). This apparent ease of implementation does however mask a number of wider issues.

Implementation itself raises the question of the relationship between the structure of a PM and the operational efficiency of a particular implementation. Although general modelling guidelines can be defined which will limit the size of a populated PM, operational efficiency considerations depend upon the implementation technology employed. As this implies, this can lead to conflicting objectives. The pragmatic advice is to avoid the danger of focusing too much on question of efficiency when developing a PM (which should in theory be independent of implementation).

More fundamental problems are posed by the need to gain widespread industrial acceptance for the PM. ISO/STEP provides a context for gaining international approval but the PM needs to pass through the STEP integration cycle during which the model is effectively rebuilt using entities drawn from the STEP IRMs. However, because of the limited involvement of the construction industry in STEP, suitable IRMs do not really exist as yet. This chicken-and-egg situation will take time and resources to overcome. The authors believe that there are similar problems relating to the STEP concept of APs when applied to the construction sector (Watson and Boyle, 1993). The main problem being the large raft of heterogeneous applications used within the industry and the consequential difficulty of scoping suitable APs. This compounds the problem facing all PM developers, the desirability of defining in advance precisely what the scope, role and form of the model should be. For instance, should the model only hold shared data between particular (known) applications or should a generic model be sought. These are issues which have not yet been fully resolved and continue to exercise the minds of the STEP community in trying to develop more complete and effective modelling guidelines.

The absence of a coherent overall methodology is made more acute in the construction sector by the limited experience of product modelling. Deficiencies which we collectively need to address. The best current advice is to be methodical, to define a method of working, even if it must be adjusted incrementally, and to maintain accurate definitions of all the terms used in a model.

The construction industry has been slow to take up innovations in information technology (NEDO, 1992) thus the attitude of the professionals currently involved in the exchange of information will be a barrier to the industrial use of PMs. Additionally, the contractual and legal implications need to be addressed (NEDC, 1992).

SUMMARY

PMs and ISO/STEP offer the construction industry the basis for a long term solution to the need to exchange and share technical information. Several problems must be overcome first, not least of which is the major task of developing the required PMs. Other than resources, the major difficulties facing PM developers in the construction sector are:

- The lack of awareness of STEP, product modelling, and data exchange technology within the construction industry, and the consequential absence of a body of experience in the construction industry in creating PMs.
- The absence of guidance to modellers and appropriate STEP Resource Models, together with difficulties anticipated in applying Application Protocols to construction, and the consequential difficulty in specifying a clear scope and context for a particular PM.
- The limitations of the available methodologies and tools; the fact that graphical modelling techniques (such as IDEF1X/NIAM) do not map easily into EXPRESS. There is also limited guidance on the usage of EXPRESS.

ACKNOWLEDGEMENT

The authors wish to acknowledge the indirect inputs to this paper from other CIMSTEEL collaborators, the funding of that project by the DTI and other sponsors, and the more recent SERC funded project "the creation and use of PMs within the construction industry" (CI-PM) which is enabling the authors to take a broader view.

References

Bjork, B (1989), Basic Structures of a Building Product Model, *Computer Aided Design*, Vol. 21, No 2, March, pp 71-78.

Boyle, A D (1993), STEP Tools Review Phase 2, Department of Civil Engineering, University of Leeds, CIPM/TP/LU/7.

Dacom (1985), *Information Modelling Manual IDEF1X*, D.Appleton Company Ltd.

Fowler, J et al (1992), *ISO 10303-22 STEP Data Access Interface Specification*, ISO TC184/SC4/WG7 N262.

de Gelder, J T et al (1991), *Guidelines for Translating IDEF1X Data models into EXPRESS*, TNO Building and Construction Research, TNO report B-91-0706.

Gielingh, W (1988), *General AEC Reference Model (GARM)*, TNO Institute for building Materials and Structure, TNO report BI-99-150.

Leppanen, P (1990), *A development effort on product data models for Reinforced Concrete structural systems*, VTT Technical Research Centre of Finland.

van Maanen, J et al (1993), *ISO 10303-21 Clear Text Encoding of the Exchange Structure*, ISO TC184/SC4 DIS.

Marca, D A (1988), *SADT: Structured Analysis and Design Techniques*, McGraw-Hill.

Mason, H (1992), *ISO 10303-1 Overview and fundamental priciples*, ISO TC184/SC4/PMAG.

NEDC (1992), *Liability Issues Arising under English Law*, National Economic Development Council (NEDC) publications, UK.

NEDO (1992), *A Simple Guide on How to do 'IT*, National Economic Development Office (NEDO) publications, UK.

Nijssen, G M and Halpin, T A (1989), *Conceptual schema and relational database design: a fact-oriented approach*, Prentice Hall.

Pentilla, H and Bjork, B (1989), *A scenario for the development and implementation of a Building Product Model standard*, VTT Technical Research Centre of Finland.

Spiby, P et al (1992), *ISO 10303-11 Description Methods: The EXPRESS Language Reference Manual*, ISO TC184/SC4/WG5 N35.

Watson, A and Boyle, A D (1993), *Product Models and Application Protocols*, under preparation for CIVIL-COMP 93, Edinburgh, UK.

---

c. 1993, Management of Information Technology for Construction, K. Mathur et al (Eds), World Scientific Publishing Co., Singapore.