# PRODUCT MODELING STRATEGIES FOR TODAY AND THE FUTURE

Chuck Eastman and Fried Augenbroe
Design Computing and Building Technology
Georgia Institute of Technology
Atlanta, GA USA

**ABSTRACT**

*Today, new technologies are being developed for information exchange in the construction industry. These include data exchanges based on formal representations, ranging from Aspect Models in specific product areas to large scale integrated Product Models, as well as new languages such as EXPRESS-X and EXPRESS-2, new data sharing technologies, ranging from flat file exchange to CORBA and Java enabled object access, and new forms of product data management to better manage the state of product data. The purpose of this paper is to sort out and review the various efforts in the field of product data technology (PDT) as a whole, from several different perspectives:*

- *in terms of new trends in each of the areas mentioned;*
- *in terms of what can be used now or in the near future in a production form;*
- *in terms of significant technical issues and limitations that may require generation changes in exchange technologies,*
- *and in terms of external business, political and other externalities that are affecting these efforts.*

*The survey will review major current research efforts and the problems and solutions they identify and the trends that they reveal, especially in regard to building product modeling issues. The result of these perspectives will be to identify scenarios of future evolution of building product modeling, with an assessment of their likelihood of coming to be, and the critical issues needed to accomplish them.*

## 1. INTRODUCTION

Product data technology (PDT) is used here as a unified perspective for information modeling of products and all associated life cycle processes, with the aim to share that information within and across engineering disciplines. PDT typically relates to concurrent engineering, project data management (PDM), data sharing, integration and product knowledge modeling in the broad sense. The increasing relevance of PDM in electronic commerce is an additional driving force behind PDT. Our focus is on the data exchange aspects.

The goal of product modeling and data exchange is to make the exchange and sharing of information among multiple applications easy and an everyday occurrence. After twenty years of product modeling development -- first in the area of building modeling and in the last ten years primarily centered around the ISO-STEP efforts, but more recently augmented by the IAI, we would like to review where we are in realizing these goals. Our intention is not a detailed survey but rather a more high level scan of the current intellectual landscape and to raise questions regarding progress toward them. Our focus is the construction industry. We wish to assess if the goals of product modeling and data sharing have been realized and if significant progress has been made. If not, given our current trajectory of efforts, will the goals of easy exchange be realized? If not realized, then we wish to identify what is lacking in current capabilities (if anything) and to identify why major benefits from product modeling research have not been realized. Possibly the lack of results are inherent in the problem and are not solvable. If this is so, then the practical limits need to be recognized. The intention of these questions are to identify research issues and to set the agenda for further discussion.

## 2. REVIEW OF EFFORTS

There has been a long history of building modeling efforts. Most of these have been research efforts, though an important early set of efforts were sponsored by the British government. The work of Bijl in estate housing (Bijl and Shawcross,75), the OXSYS and BDS efforts by Applied Research of Cambridge (Hoskins,73), the Harness System for hospital design (Meager,'73), the GLIDE work by Eastman's group (Eastman and Henrion,78.) and the Arch-Model work by Borkin's group at Michigan (Borkin and McIntosh,'81) were important early efforts in the1970s to early 1980s. All of these developed custom building models defined to support a variety of both design and analysis applications within a single integrated system.

Since this early generation there were a few commercial trials, including the RUCAPS and later SONATA effort and the GDS intelligent modeling system. These approached building design as the development of an integrated model operating through a set of views. Some issues of maintaining the consistency of views were addressed. However, these systems did not generate critical mass in the marketplace and failed financially. Proprietary research building modeling efforts have continued to the current time by various research groups. A valuable summary of recent work is available in (Carrera and Kalay,'94).

More direct data exchange efforts evolved first through the early efforts at CAD data exchange formats: IGES, DXF, SAT and others. These grew into the PDES effort in the US and ISO-STEP effort in Europe that were merged in the late 1980s. In addition to the exchange efforts, there has been the emergence of data management facilities for complex engineering projects, typically called Project Data Management systems (PDMs). These did not attempt to integrate the information carried in different design and engineering application files, but to provide a coherent way to manage heterogeneous files regarding access, ownership, sharing, change management, version control and review and approval procedures. Mature software products in PDM, EDM (engineering data management), and DMS (document management systems) are available in the market with a fairly large user base, with growing presence in the AEC field. The building industry however has been slow to absorb these new technologies.

Early efforts to apply the data exchange technology in the construction industry include the General AEC Reference Model (GARM) (Gielingh,'88) and RATAS (Bjork,'95). Significant R&D efforts in the first half of the 1990s were building on these efforts by implementing some of the general ideas in a specific application area , e.g. COMBINE (Augenbroe,‘95), or expanding the underlying capabilities of semantic representation, e.g. EDM (Eastman, Bond and Chase,'93). Both projects also attempted to deal to some degree with both the data exchange issues and the PDM issues.

In the mid 90s two EU funded projects were slated to contribute to harmonizing ongoing PDT research projects: PISA (Gielingh et al,'96) and ATLAS (Böhms and Storer,'93). PISA

(Platform for Information Sharing by CIME Applications) was to provide an approach to better capitalize on the potential of STEP and CORBA (Common Object Request Broker Architecture)/OMG (Object Management Group). While STEP concentrates on product information, CORBA provides a framework for the integration of applications. The PISA development effort supported the potential to integrate and leverage these technologies. The project was influential on its successors, as we will see in the next paragraphs. Performance and scalability results of the proposed architecture have been a concern and are being addressed in current projects.

ATLAS (Architecture, Methodology and Tools for Computer-Integrated Large-Scale Engineering) introduced a modeling strategy for the support of data exchange between different disciplines and on different levels of abstraction. Another challenge was the support of life cycle data. The project came up with a layering structure based on four levels: project, sector, discipline, application. It also introduced a general structuring paradigm based on View Type Models for actor specific system views of the building. Many ideas in the ATLAS modeling approach went into the early versions of the building construction core model (BCCM), aimed to eventually produce STEP-AP106. ATLAS can be regarded as an exemplar of a search for new paradigms that would allow a systematic approach to large scale model development. This topic was the main subject of the IRMA electronic discussion forum in 1993 (Luiten et al,'93), which at the time was considered a beacon for future directions.

## 2.1. Ongoing European R&D projects.

This section explores modeling issues addressed in recent and ongoing research projects, with an attempt to identify mainstream trends in international R&D. The survey is based on a cross section of A/E/C related efforts in the wider scope of PDT. Most collaborative European efforts are part of the 4$^{th}$ framework ESPRIT program, in the Integration in Manufacturing theme (ESPRIT, 1994-1998). It is expected that the fifth R&D framework, starting in 1999, will support continuing efforts in the same areas. Several of the concluded 4$^{th}$ framework projects have contributed substantially to STEP in other areas than A/E/C (e.g CAD*I, CADEX, PRODEX, MARITIME, PISA). Ongoing projects are expected to advance the STEP effort further, both in product and process modeling methods as well as supporting IT tools for these activities. Special incentives are underway to create awareness in SME's through STEP centers and dedicated infrastructure support projects such as ESCN. User group involvement is enforced through a number of projects that target industrial needs, such as PRIMA for the process industry and ELSEWISE in the construction and engineering industry. Other projects are aimed at pushing the envelope in different industrial sectors, i.e. ELECTRO-NET in power distribution, FUNSTEP in furniture, PIPPIN in the process industry, CIREP in electronics. Other projects deal with specific product domains that span more than one industry, such as GEM for generic engineering analysis, OPAL for business process integration and PRODNET II for virtual enterprise management. The commonality between these projects is their focus on modeling of product and process semantics, access to and sharing of product data and the (standardized) use of data in inter- and intra-enterprise projects.

Information about all of these projects can be found on the EU research web site: *www.crdis.lu*, by using the search engine and look for the acronyms by which the projects are referred.

### 2.1.1 ToCEE (ESPRIT 20587)

ToCEE (Towards a Concurrent Engineering Environment in the Building and Engineering Structures Industry) started in 1996 (Amor, 1997). The project aims to develop a framework that ties together all information types and supports processes in concurrently engineered projects. The framework incorporates product, process and document models.

A five-layered model architecture is used. The five layers are: Meta, Kernel, Neutral Partial, Aspect and Application Model. The Meta model defines the logical schema of the basic modeling paradigm(in fact an extension of EXPRESS) for all other models. The Kernel model defines the links between the underlying subsystems (layer 3) in the concurrent engineering framework (product, process, document, conflict, regulation). The 4th layer consists of Aspect models from external sources (COMBINE, CIMSteel, IFC, ..), whereas the 5th layer contains the actual instances. The project develops mechanisms to provide interoperability between the Aspect models, at different levels of coherence, i.e. 'strong' and 'weak' integration.

### 2.1.2 RISESTEP

RISESTEP (EP 20459) develops a platform for product data access in a concurrent engineering framework Industrial participants are from the car and airplane manufacturing industries. The project targets the prime need that companies have in concurrent engineering projects: access to each others data. For that purpose, the project provides a specialized middle-ware layer that provides distributed access to so-called 'Digital Mock-up Units'. The heterogeneous data access builds on STEP AP214 (Core Data for Automotive Mechanical Design Processes) for the storage mechanism and uses CORBA (OMG, 1995) for distributed access to sources of different storage formats (JAVA classes, VRML files, Relational Database). The resulting product falls in the category of PDM middle ware with support for a variety of view filters (HTML, VRML, ..) and object retrieval based on SDAI, IDL and CORBA.

### 2.1.3 CIMsteel

This pan-European research project started under the EUREKA umbrella in 1987 (Watson, 1993). It has provided the first stable implemented AEC industrial strength model: CIS (CIMsteel Integration Standards). CIS/1.0 was launched in 1995; interfaces have been built to approximately 12 structural steel engineering applications, using commercial SDAI toolkits. Many lessons have been learned, and the development of a formal STEP AP230 from the CIS is underway.

### 2.1.4 CONCUR

This project, started in 1996 in the Brite-Euram program focuses on managing information flows across the boundaries of different industrial partners. Industrial case studies will be performed on large scale engineering projects. The project will identify the technologies to support the semantic sharing, access and management of the information flow in these projects. There is no apparent emphasis on product modeling as a separate target.

### 2.1.5 GEN

The Global Engineering Network (GEN) is a European initiative to extend PDT, CAD and communication technologies into the electronic market place accessed over Internet. The emphasis is on suppliers of engineering products and services. GEN comprises a set of linked projects, e.g. GENIAL (GEN infrastructure) , PROCAT-GEN(multi-media catalogues as marketing instrument) and MATES (multi-media assisted tele-engineering). For our purposes, GENIAL is the most relevant to study.

GENIAL defines GEN objects in a so-called Common Semantic Model.(CSM). The CSM contains a product metamodel for product representation on a meta level. Each particular representations (CAD file, PDF, STEP file) of a product is an instance of the product metamodel. For domain specific information a product description is associated with an ontology model which can be extended by STEP-AP's as they emerge. GEN extends the functionality of current CAD and PDM systems by adding functions for authoring, searching, and managing large dispersed collections of product data.

### 2.1.6 PIPPIN (Esprit 20506)

PIPPIN (Pilot Implementation of Process Plant Information system) concentrates on integration of all as-built data for the facility, emanating from different sources in different formats and expressed in different languages. The project ties in strongly with the work of WG10 in the ISO-TC184/SC4 (STEP and other) standardization effort. WG10 looks at ways to improve the architecture and of the standards. One idea is to base this on Generic Data Models, as proposed in (West, 1996), which in effect changes the scope from product data to enterprise data and would undoubtedly.  Keeping a complete record of a constructed facility in a coherent 'electronic signature', i.e. a single consistent set of engineering data, is the goal of the PIPPIN data warehouse. The warehouse delivers information on demand to operations systems during the service life of the asset. The check out/check in mechanism is controlled through a data integration layer expressed in a common business data model, in this case the EPISTLE model.

### 2.1.7 VEGA (ESPRIT 20408)

The VEGA (Virtual Enterprises using Groupware tools and distributed Architectures) project aims to integrate business and technical processes. It targets the Large Scale Engineering (LSE) industry (i.e. construction, process, shipbuilding and other one-of-a-kind sectors) particularly in communication and information supply. In these sectors, distributed teams carry out concurrent activities. VEGA will contribute towards solving some of the problems by developing models and tools within a Computer Integrated Construction (CIC) environment. In particular, the concept of a coordinated project database is retained, but it is decentralized through distribution mechanisms and workflow, facilitating information flow between computer systems. To implement CIC in virtual enterprises, distribution and workflow technologies are required for information sharing and to manage the sharing process. The challenge is the delivery of a redesigned architecture to publish product data not in a client/server mode but in an all distributed product data approach.

### 2.1.8 ELSEWISE (ESPRIT 20876)

The ELSEWISE Project is a User Reference Project concentrating on the IT and Product Data Technology (PDT) needs of the European Large Scale Engineering (LSE) industry. ELSEWISE has a bias towards building and construction and is the pilot study phase of the ELSEWISE User Reference Project Concept. Future phases will monitor the results of the development projects, and establish a validation community in support of the implementation of new technologies. A wide range of construction partners mainly , but not exclusively, industry based.

### 2.1.9 GEM (ESPRIT 8894)

GEM (Generic Engineering Analysis Model) was recently concluded. It addressed analysis methods such as finite element analysis (FEA) or computational fluid dynamics (CFD) to be used more effectively within the design process throughout the product's life.

The project developed a generic engineering analysis model (GEM) for the exchange, data sharing and archiving of engineering analysis models. The GEM supports a range of industrial applications, a variety of design and analysis methodologies, and facilitates the use of analysis results in the design model. This work is accomplished by first classifying the data types which can be employed in the analysis process and hence defining the way in which the results of analyses can be represented in a general manner for reuse as input to other processes.

## 3. PERCEIVED TRENDS

Reviewing the current efforts from different perspectives, a number of new trends can be distinguished.

The semantic integration challenge is no longer dominant:
Except CIMsteel, there seems to be no particularly concern about the detail definition of AEC product models. This appears be based on the belief that somehow the semantic integration problem will go away. This apparently disregards many of the lessons reported by the CIMsteel team.
The emphasis on modeling methodologies exhibited by previous generation of projects (ATLAS, PISA) seems to have evaporated, although some projects are using frameworks established in previous projects (GEM). Other projects indicate a particular framework approach but do not seem to spend any resources to 'new' building models (ToCEE, CONCUR)

A new pragmatic approach to the integration problem has emerged:
The new focus seems to be on middle ware layers giving access to heterogeneous representations in a uniform presentation but without an explicit focus on semantic integration (VEGA, PIPPIN, GEN). There is an apparent shift in the granularity of integration activities. A "zoom-out" in focus has taken place: the semantic integration problem on design team or individual application granularity has been replaced by the inter-enterprise product data management. In doing so, many of the items hitherto on the research agenda seem to left over to the PDM products of the individual enterprises.

The majority of projects are dealing with 'Check out/Check in' issues, but mostly on coarse granularity and mostly on the scale of complete coherent subsets of the data, which are self contained and have no or very little overlap with other subsets. The interoperability problem is reduced to simple bookkeeping, using a data integration layer similar to the ones defined in the CAD Framework Initiative (CFI, 1995) , (van Leuken, 1996), which is a dominant effort in electronic circuit engineering but seems to be getting very little attention in the EE/C projects.

Data sharing technology is converging:
EXPRESS is overwhelmingly accepted as the modeling tool, whereas SDAI specification is used to access the data. The SDAI-CORBA/IDL binding is the favorite approach in most prototype implementations, although performance problems persist. The general belief is that these problems will be overcome in future releases of SDAI toolkits and binding specifications.
The harmonization of multiple standards is getting more attention involving links between SGML, STEP, VRML etc, hyperlinks between documents and product data, and workflow and product data.

The scope of integration emphasis is changing dramatically:
The emphasis has shifted from a data centric approach to a process centric approach, and is moving towards an enterprise centric one. Projects are concerned with the correct positioning of PDT in the overall business process, recognizing the relationships to document management, workflow management, and enterprise resource planning

These observations lead to some concerns regarding progress in the building industry. It is obvious that we have not reached the level of model stability (or even a model for that matter) that other industries seem to have reached. So the question is whether we can afford to widen the scope to the areas explained above before we have reached the level of an AP221 (process industry) for buildings. Ultimately, this is a question of an analysis of the building industry needs for the near and long term future. In other industries these needs have been implicitly expressed in business plans of the leading firms in the industry. The lack of such major players in the building industry and the small size and relative short duration of each project combines with very little market wide evidence as to what the industry needs really are.

**Some observations**

There seems to be little consolidation of project results between different 'waves' of projects. The funding agencies seem to endorse an attitude to 'move on' and fund the next stage -- whether the previous stage is resolved or not. This leaves 'hard' and often hidden problems unresolved. In many of the recent and ongoing projects there is a tendency to go from one 'fancy' prototype to the next, with increasing but narrow functionality without any guarantee that the approach, scalability, performance will allow general application.

Most projects explicitly or implicitly propose strategies to develop frameworks for product models, dealing with:

- decomposing at a high level a building model into a set of subsystems
- abstraction of certain behavioral models allowing these to be embedded in various objects having that desired behavior
- the intermediate abstractions of a building -- at the assembly level -- varies for different types of behavior, for different types of design, and for fabrication. Thus the intermediate abstractions seem particularly diverse and open-ended.
- most issues of geometry, materials, ownership, versions and releases and other management coding of information can be dealt with through general libraries that can be shared within different aspect models.

The A/E/C research community at large does not seem to be able to make a concerted effort to consolidate proven approaches in these areas and reuse and test these solutions in the next funded prototype development. This is a necessary step to incorporate the backing for these developments from the industrial partners in the large collaborative projects. As a result, both STEP standardization and IFC efforts are not embedded in a consolidated strategy. Hence the foundation of any model will be small and the lack of progress of these efforts is evident.

## 4. ISSUES

In this section we attempt to articulate some of the major issues facing the AEC product data exchange field, based upon the previous review and interpretation of trends. Our concern is not to address detail issues regarding good or bad practices in the development of specific models, but rather to review the larger domain of all modeling efforts in the AEC and especially the construction area. The issues are not meant to be complete, but rather to raise for discussion topics that by attending to them, will allow us to make the best possible progress.

## 4.1 Do We Need Building Models?

This question is easy to answer: every time we express a fact about a building (design) we do this based on a mental conceptualization of the building. We do this using personal abstractions, but keep only the end result and soon lose any explicit recollection of the process used in definition. During communication, the receiver/interpreter of our information does not necessarily have the same mental concepts, which gives rise to common misinterpretation problems that may arise in any communication. So, very early on, designers and builders have started to use more formal language and expressions of the information they want to convey/ interpret. These expressions vary from informal sketches and notes to rule-based drawings and precise data definitions. The understanding of the semantics in human communication is based on shared mental models and definitions.

Recently the use of computers has forced much more attention on the use of formal expressions based on a rigid syntax and predefined semantics, enabling (at least part of the) meaning to be derived from a shared semantic model. Whereas humans are adapt at re-conceptualizing parts of information that does not fit with the rest, computers are quite poor at such error corrections (we have not figured out the instructions that would allow computers to do this on their own). Thus computer communication requires exact definition of the translation or conversion rules for exchanging information between different models. In the case we are exchanging information between computers (e.g. between two software applications) we cannot rely on any existing mental capacity to fill in the gaps so we have to be 100% formal.

So, we need to rephrase the question: *do we need coherent and complete building models?* The intellectual conflict encountered in main stream building modeling is to defend the seemingly endless effort of constructing a ‘complete’ semantic representation of a building against the current situation where we witness the use of disparate, poor and sometimes incoherent representations. But, as we all know from personal experience, the deficiencies of current representations, associated with different actors and at different stages in the building process, are currently overcome by experience, improvisation talents, craftsmanship encoded in established working practices, procedures and teamwork. It seems that we have a highly trained work force that is very capable of exchanging and augmenting the information content. The result is that integrated buildings with a high level of complexity and novelty are being constructed without them ever having been fully represented in a single consistent representation. The ability to achieve integration, though with significant levels of effort, gives confidence to current working styles. There has been little in the way of demonstrations for practitioners offered by the research community that shows that a significant improvement is possible rather than some incremental efficiency improvements. At the same time, the effort required to achieve these ends bring many practitioners to the conference table, for example at IAI Domain meetings. The lack of confidence that significant improvements can be made is the most important barrier for the industry to support current exchange technology, that is by its nature more constraining than the present methods and tools.

Another pitfall that is hard to avoid in any approach to building models is the semantic ‘start from scratch’ trap. Modeling efforts have been handicapped by taking the absolutist and closed domain approach. It is quite amazing to see how people start building models from atomic bits of ‘information’ (comparable to the old and never fully conclusive classification of what ‘things’ in a building mean, what function they have or role they play), hoping (in the end) to represent all possible manifestations of a design or building that one could ever encounter. Not only is this impossible to accomplish (there are probably an infinite number of interpretations of some building part), but also one can wonder about the need to accomplish it. After all, as soon as the human enters the picture, there is necessarily a continuous mapping from the computer representation to the mental model held by the human user. The goal is not to realize all human interpretations of a building in a computer representation, but to realize mappings between all computer-based representations.

Electrical engineering has a rich base of applications for specifying goals of a circuit design, for defining its logical structure, its behavior, its physical layout and for automatically fabricating the circuit. Piping design also has a rich set of applications. However, in general, the suite of applications used in architecture and construction are meager. Compared to most other engineering or design disciplines, the range of serious AEC applications is quite small. Few applications exist that could take advantage of advance building models. It is sometimes argued that the lack of integration is the reason the applications have not been developed and that such applications will 'blossom' once an integrated building model comes into being. The net effect, however, is that the building model/integration must develop both the application base that will effectively use such a model and also simultaneously develop the building model integration standards. It is reasonable to ask if this is a practical way to proceed.

## 4.2 The Sequence of Development of a Building Exchange Architecture

The basic approach of ISO-10303 (STEP) has been to define Application Models for a functional product domain. The direction of most STEP efforts has not been to develop models of whole products, but distinct application protocols (AP) for "aspects" that require integration of a set of applications supporting design, analysis and fabrication. The idea behind application APs is to develop a strong set of middle level integration platforms providing support for a suite of applications, allowing incremental refinement and addition the different APs. As more is learned about the semantic issues in the various APs, the potential for developing larger models of complete products will become possible. We might call this a bottom-up approach. It does not attempt to resolve an overall product model for automobiles or airplanes, for example. Rather it focuses on supporting the development of smart applications, but without making strong commitments to an overall product model that may turn out to be short sighted.

These issues are especially important in the architecture and building area. There are few large suites of intelligent applications based on complex product models. Rather there are only a few tools, most general purpose, that do not yet demand complex object exchanges. The few major efforts to develop aspect level integration have been relatively successful. Most notably, this includes the CIMsteel work from Leeds University (Watson,'93). This aspect model is in wide use in Europe and is being reviewed for use in the US and to be adopted as an ISO Part Project. A prototype aspect level integration that received wide acclaim earlier was the COMBINE II project. But these two efforts are exceptions. We should remember that each performance area -- structures, energy, lighting, acoustics, fire safety, for example -- has multiple application needs for use at different levels of development or abstraction within the overall building design and construction process.

On the other hand, the building product modeling community has made repeated efforts to develop a large-scale "framework model" that integrates all functional subsystems in a building. There have been three STEP efforts: the Version Str100 developed in 1995 (Wix,'95) , the BCCM model of early 1996 and the Part 106 BCCMT100 Model of early 1997 (Wix,'97). Separately, the EU funded the ATLAS effort. Recently, another large scale framework model has received much attention and effort: the Industry Foundation Class model of the International Alliance for Interoperability (IAI,'97). These different versions of a building model vary significantly and seem based on somewhat different philosophies. While each has added useful modeling constructs and understanding to the framework modeling problem, none as yet have received wide support or endorsement. While the aspect models have had a clear set of applications to integrate, providing a strong definition of what is functionally required, the functional specifications for the different framework models are much less clear with regard to their measure of effectivenss.

The AEC research community has only limited resources to expend in bettering its situation. Where should these limited efforts be directed now? Toward a large integrated model for which applications do not yet exist to support it? Or to smaller scale applications of a scale of CIMsteel or reinforced concrete design, mechanical equipment simulation and design, etc.? Do we know enough to define an effective overall building product model? If such a building model was developed, what applications would use it? Wouldn't we have much better chance of success once a set of aspect models have been defined, giving the framework model explicit integration criteria to realize?

## 4.2. The Degree of Automation that can be Accomplished in Building Model Exchange

The ideal of data exchange is the completely automatic transfer of data from one application to another. The issues raised by this goal are quite broad. At one level is the translation of building elements from one intelligent CAD system to another. Because in-place construction, when complete, is unitary, the various decompositions used in applications are abstractions of the building. The decomposition of building fabric into parts can be done in many ways and the parts are likely to be organized differently in different design or construction systems. Are the wall intersections separate elements in wood construction or part of one of the walls? Are window and door jambs and headers part of the window assembly or part of the wall? These differences likely depend upon means of fabrication and cost estimation, not just arbitrary decisions of software developers. The differences must be addressed in building model translation and it is unclear if these can be accomplished automatically or require manual interpretation.

Other kinds of translation more clearly require human intervention. In many applications involving analysis, it becomes an open issue which objects are to included in the analysis. In a large window wall system, it calls for judgment which elements are to considered structural ones. In complex analyses, such as in-door air flows using computational fluid dynamics, what are the boundary conditions at the edges of the modeled region? Such questions will require human judgment for the foreseeable future.

Another aspect of translation requiring human intervention is the idealization of a design for many forms of analysis. When should the size reduction of columns as they go higher in a building, while keeping some of their edges aligned, be treated as eccentricities? When should notches in a structure be explicitly considered as structural changes to the element? These also are judgmental issues that rely on human expertise to deal with.

Due to the nature of loosely organized design management in current practice, it is only in rare occasions that no human intervention will be needed in the transmission of data from one actor (and his tool) to another actor (and his tool) in the design team. Efficiency can be gained if we can keep these interventions to the minimum and in any case delegate the laborious routine like mappings between applications to the computer. Eliminating human intervention will most likely only be possible in exchanges between not very dissimilar disciplines and in well-controlled project work flows. By realizing this as a fact, we should devote efforts to those areas where efficiency losses occur in current practice. The problem is that we have no clear understanding where data exchange efficiency sinks occur.

Many (and probably the most important) efficiency sinks occur at the 'disconnects' of humans rather than at the disconnects of applications. Design iterations hardly ever take place without the intervention of the expert consultant, which makes the mapping between the computer representation and the human conceptualization unavoidable. Failing to support the augmentation during this mapping process (which involves user experience, expert knowledge, etc.) will lead to rejection of the computer representation. Major disconnects exist to date at the transfer points between different disciplines, e.g. for commission, estimating, construction planning, construction and sub-contracting. The ultimate challenge is to remove the inefficiencies at these major transfer points. Of course there is much debate now whether we need these transfer points at all and cannot avoid them altogether by changing the process. This calls for a balanced data and process centric approaches.

## 4.3. Integration of PDT Issues with Product Modeling

The main focus of STEP and IAI has been to focus on the building model and to demonstrate it in the easiest possible way. This has been by batch file transfers. Recent research building modeling efforts have accepted the partial results of building models and integrated them into higher level efforts addressing data sharing, exchange and collaboration. ToCEE, RISESTEP, CONCUR, GEN and VEGA have this level issue as a major focus. The earlier efforts of COMBINE and EDM have shown that product data management is crucial capability for the effective practical use of product model data.

In many cases, the desired scenario is for data exchanges between users to be two-way, with changes made to existing data, to the addition and deletion of existing data. Some practical updating methods have been developed (Eastman et al, 97):

- two-way mapping facilities between application models;
- support for incremental updates, allowing attributes or subsets of an assembly to be replaced with others;
- change detection and propagation, allowing any of the users of the repository to be notified when the data they are using or used earlier changes;
- communication and collaboration facilities, supporting effective work coordination between team member, possibly working at a distance in space and time; these facilities include process and well as product coordination;
- support for model and process evolution, so that new applications/ tasks can be integrated into the design/engineering/construction process;
- information access and coordination through new generic interfaces, esp. based on WWW interfaces..

All such facilities go beyond the structure of the data model itself. They determine how effective the use of the data model will be.

## 4.4. Process Modeling is Integral with Product Modeling

As experience grows in the PDT level of management of product data, the need for explicit representation of the processes that use the data become clearer. When an application is added to a building or aspect model, under what conditions can that application be used? What other applications must be run prior to the new one? How can we track the dependencies among different applications?

In the end, these questions can only be answered through explicit representation of the design and construction planning processes. As we proceed in the development of product model implementations, the need for process modeling to be used as a base for coordinating operations on the product model will become mandatory. This is a crucial part of PDT technology that will eventually affect how design and engineering are practiced.

## 4.5. The Long Term Adequacy of Current Product Modeling Technologies

Product data exchange has developed important technologies. Probably the most is the widespread adoption and use of EXPRESS and EXPRESS-G for representing product models (Schenck,'91). A number of small software companies are developing commercial software based on EXPRESS and EXPRESS-G, for graphical representation, parsing of models and for integrating with Integrated Resource Models. Another technology of significant importance is the SDAI software libraries for writing, instantiating and for later reading of model data, also developed by some of these companies. A third important technology is the definition of the various Integrated Resources that allow common parts of models to be quickly defined in a standard fashion. The geometric modeling resources of Part 042 is a particularly example of such technology. It is being relied upon, for example, in the development of the IAI Building Model. These efforts has encouraged standard representation and shared use.

A fundamental concept of the ISO-STEP architecture is the separation of reference model and interpreted model; requirements and draft model are proposed as one model, the interpreted and fully integrated representation as another. Initially a graphical language -- NIAM, IDEF1x or EXPRESS-G --is used to express requirements by human users. After careful design, this model is interpreted and turned into a (textual) EXPRESS model, allowing effective machine interpretation. The interpreted model is file format or database implementation independent, allowing multiple implementations (file, relational database, object-oriented database, expert system). Supporting multiple implementations was an important lesson from the IGES experience.

The current implementation of EXPRESS supports automatic generation of data structures supporting a product model. Rules embedded in DERIVE and WHERE clauses are parsed but not implemented. EXPRESS is a fairly generic object-oriented language, with a rich set of constructors and language features for controlling multiple inheritance to support different uses. Application models generally carry literal values for depicting shape.

We believe these facilities should be relied upon by all members of the product modeling community for communication and sharing of data. At the same time, we must embark on work to strengthen them so they can fulfill their role more effectively in the longer run.

### 4.5.1. Adequacy of EXPRESS and EXPRESS-G

After extensive use, a number of issues have arisen regarding the current languages for defining reference and interpreted models:

- A role of the three reference modeling languages was to facilitate definition of semantic relations[1] by domain experts. However, the three Reference Model languages were not designed for this purpose (except possibly EXPRESS-G) and are limited in defining complex building models (and also many other kinds of product models). The semantic range of EXPRESS-G is a subset of EXPRESS. It cannot depict Rules or distinguish different constructors, for example. Thus the interpreted model can be richer than the reference model from which it was derived. This negates somewhat the role relationship of the two models. In the long term, a richer and more extensible reference language will be needed if we are to make significant progress in product modeling.
- While the EXPRESS language facilities for defining object classes are well developed, the language facilities for defining relations between entities are weak. Only the INVERT constraint between relations and arity properties of relations are well supported.. Relations cannot be defined as subtypes of other relations, because they are not objects. However, there are benefits for subtyping relations, just like the benefits of subtyping entities. Relations should have what computer scientists call "first class" status. That is they should be a type of object, with the ability to receive attributes, by subtyped, and so forth.
- There has been much work arguing for and developing examples of integrating models expressing function, structure and behavior of a product model . Current EXPRESS language facilities do not support modeling of behavior, nor does it support well modeling relations between these three types of design information.
- Both EXPRESS and EXPRESS-G are static languages. They allow definition of entities and relations, but once created, they do not support evolution or change of the models. There is growing recognition that product models need to evolve over time.
- Current reference or interpreted model languages are not able to support the definition of geometry that is derived from user-specified parameters. This makes it difficult to define construction objects that vary in size according to a fixed set of possible dimensions. It also does not allow definition of models where wall, piping or ductwork geometry is derived from a centerline layout. Work is proceeding now on the development of EXPRESS-2, that may extend the current EXPRESS to support parametric modeling.
- Application protocols were developed assuming that data exchange needs could be defined around single application models. However, it is commonly the case that data exchange is also needed between models of different types. For example a structural steel model and a piping model may need to provide data to a common model, to deal with spatial interferences and also load transfers. In response to these types of issue, there has been several efforts to extend EXPRESS with mapping facilities (Spooner and Hartwick,'97), (Bailey,'96). Mapping between aspect models provides another mode for framework integration.

All of these issues together result in EXPRESS models being both incomplete and ambiguous in implementation, when they have to be interpreted by computers. Refinement of these technologies is required in the longer run if we are to make progress in the field. At the same time, the current technologies can advance us from where we are today.

---

[1] Here, we use semantic in a specific concrete way: *semantic* refers to the structure, properties, behavior and relations of an entity in the real world or a computer model in a virtual world. A computer model attempts to represent the significant semantics of a real world object.

## 4.5.2. Adequacy of SDAI

With IGES or other neutral file format, the minimal requirement for implementing a mechanism for data exchange between two applications consists of writing two mapping interfaces for a one-way mapping and four mapping interfaces for two-way exchanges. In each of these mappings, there are two parts: (1) the extraction of data from or the writing of data to the internal format of the application; and (2) writing the data to or reading the data from the neutral file format. The exchanges all were batch oriented. That is, all the data selected for exchange in one application was received by the other.

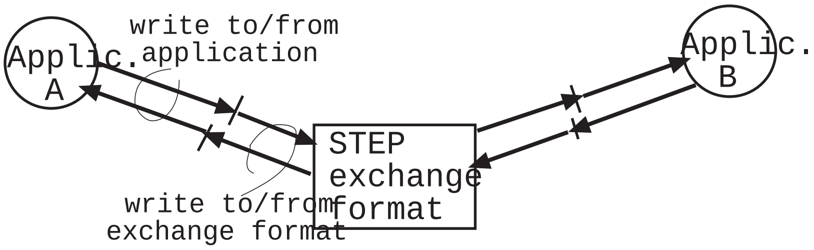


*Figure One: The two part mapping between an application and an exchange model.*

With ISO 10303, support technologies such as SDAI simplify this task. They provide standard facilities in the form of functional calls to read from and write to the neutral format. These are characterized in Figure Two. Thus in each of the exchanges a significant amount of the effort -- from 1/3 to 1/2 -- required to read to or write from the exchange format is resolved. For a given exchange, much of the work has been reduced.

STEP Data Access Interface software libraries provide powerful facilities for implementing interfaces that write out instances of an EXPRESS model or read in the instances. These facilities expedite the software implementation of interfaces between a model and an application. They essentially deal with one half of the problem.

Other approaches to the definition of interfaces are possible. Graphical 4$^{th}$ generation languages allow definition of schemas for databases. They also allow definition of queries for the extraction of data. No CAD system yet supports a simple graphical language for extracting data. There has been some interesting work in the development of mapping languages (Armor,95), (Spooner and Hartwick,'97), (Chen and Eastman,'98). These show that if CAD and engineering applications can be mapped into a common modeling representation, the development of mapping facilities between them can be much simpler than the development of data exchange interfaces are today. Augmented with some level of agent-based querying, possibly using KIF-like capabilities, the potential may exist for relatively simple development of data exchange between heterogeneous applications.

## 4.5.3. Evolution of Integrated Resources

The third major benefit of STEP is the definition of its Integrate Resources. These are being used widely. However, at the same time, there is background work leading to the extension and upgrading of these resources. Of particular significance is the move to parametric modeling. Eventually this will lead to revisions of Part 042 or other Parts used in geometry. These revisions will be helpful, but much progress is possible without them.

## 4.6. What are the Needed Processes for Doing Product Modeling?

Few people are trained as modeling experts. However, the role of domain experts usually do not overlap with those of modeling experts. Thus communication between the groups must be considered as a serious issue. Careful study of how to extract domain knowledge is needed. User driven specifications are desired, not of what someone would like, but what can be computer implemented in application programs. Such work is a normal part of the field of requirements engineering (Jackson,'95).

There is a lack of testing and assessment of what is successful and not successful in building product modeling, what works and what does not. There is too much re-invention. As a field, we must learn to act as a research community that builds upon and improves upon each other's efforts.

## 5. RECOMMENDATIONS

What answers can we derive out of these thoughts? As a field, how can we proceed toward realizing the strong level of integration and data exchange that all of us would like the future AEC computing environment to provide? We posit a variety of recommendations below:

1. build small size models as 'desired' islands of automation (e.g. CIMsteel) that can both integrate existing applications and encourage domain specific development of new applications. Distinguish this target clearly from sector wide semantic efforts because objectives, stakeholders, and time line are very different. In fact, aspect models for clusters of applications should belong in the industrial development arena, not in the research arena;
2. gain significant involvement of industry to help develop detailed specifications of data exchange tasks, including explicitly the role of human judgment in the process; identify efficiency sinks where the task can be fruitfully augmented by automation without necessarily automating the entire exchange process;
3. build middleware tools embedded in business processes that aid humans to navigate between the islands of automation of financial models and those of product models, between time cost management and the product modeling environment;
4. build rich viewing and interpretation tools, not relying on 100% consistency but on the right level of impressionistic interpretation adapted to the skill set of the observer (in an impressionist painting all brushstrokes don't seem to be related if you are too close to the building);
5. consider more publicly and thoroughly the specification stage of the modeling effort, bringing in industrial interests and end user concerns, to be included with the technical limitations of 'what-can-be-done';
6. develop model test procedures and concepts, that allow assessment of the capabilities and limitation of existing or proposed models. This is especially practical where a suite of applications is identified that the model is to support;
7. while research funding has focused on large prototype demonstrations, there is crucial need to support research addressing the missing gaps in detailed modeling issues, refining what has been proposed in a way that is testable. This includes work at the computer science level addressing the languages used, the access libraries, model viewers and other support technologies;
8. address scalability and reliability issues; consider the size and performance limitations of current SDAI and other methods and develop ways to deal or resolve them.
9. develop a richer set of language constructs, first as macros in the current EXPRESS, then when tested, as primitives in a newer version of EXPRESS.

10. Integrate more fully the process/product integration of models within a broader architecture of modeling technologies;
11. explore new approaches to inter-domain solutions, not based on integrated semantics but a mix of human, process and data centric approaches. This should allow local augmentation of semantic gaps rather than 'integrated' frameworks. A first start is through the use of 'mapping languages' between aspect models;

We personally believe strongly in the potential benefits of product data exchange and process based coordination and collaboration. But we have a tremendous amount of work to do to realize the goals before us.

NOTE:
This work was supported in part by the National Science Foundation, grant No. IRI-9319982.

## REFERENCES

Amor, R., Hosking, J. and Mugridge, W. (1995), "A Declarative Approach To Inter-Shcema Mappings", in Fisher, M.A., Law, K.H. and Luiten, B. (eds), *Modeling of Buildings Through Their Life-cycle*, CIB Proceedings Publication 180, pp.223-232.

Amor, R.W., Clift, M., Scherer, R., Katranuschkov, P., Turk, Z. and Hannus, M. (1997*) A Framework for Concurrent Engineering* - ToCEE, European Conference on Product Data Technology, PDT Days 1997, CICA, Sophia

Antipolis, France, 15-16 April, pp. 15-22.Augenbroe, G. (1995) *COMBINE 2, Final Report*, CEC-JOULE program, Brussels, http://dutcu15.tudelft.nl/~combine/

Bailey, Ian, [1996] *EXPRESS-M Reference Manual*, ISO TC184/sc4/wg5 N243, CIMIO Ltd, Brunel Science Park, Surry, England, 12 August, 1996.

Bijl, A., and G. Shawcross, (1975) "Housing site layout system" *Computer-Aided Design*, 7:1, pp. 2-10.

Bjork, B.C. (1995) "Requirements and information structures for building product models", *VTT Technical Research Centre of Finland Research Report*, Publication 245.

Böhms, H.M. and Storer, G. (1993) Architecture, methodology and Tools for computer integrated LArge Scale engineering (ATLAS) - Methodology for Open Systems Integration, *ESPRIT 7280, Technical report*, TNO, Delft, The Netherlands.

Borkin h. J. McIntosh and P. McIntosh, (1981) "ARCH:Model: A Geometric Modeling Relational database System", *ARCH:Model User's Manual*, Architectural research Laboratory, University of Michigan, Ann Arbor, Michigan.

Carrera, G. and Y. Kalay (eds.), (1994*), Knowledge-Based Computer-Aided Architectural Design*, Butterworths-Heinemann Press, N.Y.

CFI (1995) The CAD Framework Initiative, 4030 W. Braker Lane, Suite 550, Austin, Texas 78759, USA, cfi@cfi.org.

Chen, Y and C Eastman (1998) "Mapping facilities supporting model interdependency maintenance in product databases" Design Computing Research Report, Architectural Doctoral Program, Georgia Institute of Technology, Atlanta, April, 1998.

CIMsteel (1995) *Computer Integrated Manufacturing for constructional steelwork*, Eureka project 130, Brussels.

Eastman, C.M. and M. Henrion (1977) "GLIDE: a language for design information systems", *SIGGRAPH'77 Proc. Computer Graphics*, 11:2, pp.24-33.

Eastman, C.M., T.S. Jeng, A. Roy Chowdbury, K. Jacobsen, "Integration of Design Applications with Building Models", *CAADfutures'97 Conf.*, Munich, (August, 1997).

Gielingh, W. "General AEC Reference Model (GARM)" *CIB Conference on The Conceptual Modeling of Buildings*, TC5.2 and 7.8, Lund Sweden, (Dec,) P. Christensson (ed.)

Gielingh, Wim, Robert Los and Bart Luiten, The PISA project, a survey on STEP. Shaker, Aachen, 1996.

Hoskins, E.M. (1973) "Computer aids in system building", *Computer-Aided Design*, J. Vlietstra and R. Weilinga (eds.) North-Holland Press, N.Y. pp.127-140.

International Alliance for Interoperability (1997) IFC Release 1.0.

Jackson, M.A. (1995) *Software Requirements and Specifications*, Addison-Wesley.

Luiten, G. T. Froese, B.C. Bjork, G. Cooper, R. Junge, K. Jarstila and R. Oxman (1993) "An Information Reference Model for Architecture, Engineering and Construction", *Proc. 1st International Conference on Management of Information in Construction*, K. Mathur, M. Betts, K. Tham (eds.), World Scientific Pub. Singapore, pp. 391-406.

Meager, M.A. (1973) "The Application of Computer Aids to Hospital Building", *Computer-Aided Design*, J. Vlietstra and R. Weilinga (eds.) North-Holland Press, N.Y., pp. 424-453.

OMG, CORBA (1995) *Architecture and Specification*. Object Management Group.

Schenk, Doug, [1991], EXPRESS Language Reference Manual: External Representation of Product Definition Data, ISO TC184/SC4/WG5, Document N14 (29 April, 1991)

Spooner, D.L. and Hardwick, M. (1997), Using Views for Product Data Exchange, *IEEE Computer Graphics and Applications*, Sept./Oct. 1997, pp58-65.

van Leuken, R. and van der Hoeven, A. (1996) Framework Services: Design Data and Design Flow Management for Engineering Environments, *The Second World Conference on integrated design and process technology*, Austin, Texas, 1-4 December, (to be published).

Watson, A., A. Crowley, A. Boyle And G. Knowles (1993) "Practical Problems in the Development of Product Models", *Proc. 1st International Conference on Management of Information in Construction*, K. Mathur, M. Betts, K. Tham (eds.), World Scientific Pub. Singapore, pp. 571-582.

West, Matthew (1997) Integration and sharing of industrial data. *European Conference on Product Data Technology*, PDT Days 1997, CICA, Sophia Antipolis, France, 15-16 April, pp. 145-154.

Wix, J. (1995) "Building Construction Core Model: Version STR100", located at http://maillist.civil.ubc.ca/~tfroese/rs…models/directory/step_bc_core/str100.htm

Wix, J. (1997) see http://www.bre.co.uk/bre/research/cagroup/consproc/itra/bccm/