

# Integrated Product and Process Data Management

## INTRODUCTION

One of the most significant impacts on product development in recent years has been made by the application of data management techniques, which are known as product data management, engineering data management, engineering documentation management, and integrated product data management. The initial objective of applying these techniques was to provide methods for the storage and retrieval of electronic product data (e.g. product geometry, product structure, and design documents) that were created in the product development process. Lately, the capabilities for process and work flow management and product structure management are also provided to better manage product data and development process.

In the 1990s, concurrent engineering has been employed to improve product manufacturability and quality, and to reduce development cycle time and cost by resolving product, process, and organizational issues at the early stage of design. Concurrent engineering involves all activities in the product life cycle, including product design, process planning and management, and manufacturing. To accomplish these activities, product developers must manage diverse, heterogeneous data that support the needs of various applications. Therefore, data management plays an increasingly important role in tracking products from conception to retirement and supporting effective communication and information sharing among functional groups and individuals of a product development team.

Along with *product data*, *process* and *organization data* are also integral parts of product development. Therefore, to offer an integrated approach that fully supports the activities of concurrent engineering, we enlarge the scope of product data management in order to address integrated product and process data management for this special issue.

## ISSUES OF INTEGRATED PRODUCT AND PROCESS DATA MANAGEMENT

Integrated product and process data management deals with the modeling and management of product, process, and organizational data that can be shared throughout the entire product life cycle. Product data that define a product include not only the product geometry and product structure, but also information about product functions, product constraints and design rationales, and development history. Process data that describe how a development process can be accomplished include work flow, development policies and rules, process capabilities, facilities, and available resources. Organizational data is related to the organization of individuals and functional teams, use of resource, and the responsibilities of people and their roles within the development cycle. We classify the issues of integrated product and process data management into six categories.

### Product Modeling and Representation

Product modeling deals with the creation and manipulation of product models that are capable of representing all interesting aspects of the product at various levels of abstraction. Computer-aided design (CAD) technologies have made a significant impact on product modeling. With CAD tools, designers are able to create product geometry quickly, and engineers can evaluate the performance and producibility of products by accessing and analyzing the relative images of alternative designs. Along with the geometry, a product model should also include a product's functions, specification (e.g., geometric and engineering constraints, tolerances, surface finishes, and engineering relationships) and design history.

In a concurrent product development process, everyone involved in product development shares a common product model; however, each has different views of the model. Therefore, the model must meet the modeling requirements in all phases of a product's life cycle. A product model that is able to

represent complex product geometry and define all required information, yet support multiple views for various applications, will be the basis of future modeling systems.

### Product Data Management

Product data includes the product model, product structure, engineering documents, and data files (e.g., IGES files). Product models, which contain geometric and nongeometric information about products, are created to describe the product and to provide support for product development activities. The product structure defines the relationships among the product components and the bill-of-materials. Engineering documents describe how the product is produced. Product data management provides version and access control for the product data as well as maintains the relationships between the product and the related files and documents.

Another requirement for product data management is the ability to support product model refinements and to maintain the association between the refined models. This requirement is needed for the support of different application perspectives and involves the generation of application-specific geometry, propagation of product requirements and specifications, and the establishment and maintenance of the association between application models.

Design is an iterative process that takes into account the influences of a number of applications. For instance, the results of molding analysis might cause a change of the molding design, which, in turn, affects the mold model and the results of manufacturing planning. Therefore, authorized design changes must be allowed to be automatically reflected in all other applications, regardless of where they are used. For instance, analysts can perform simulations of a molding that is still being refined by the molding designer. Once design changes are approved, the results of molding simulation will be automatically updated to reflect these changes.

### Process Modeling and Management

Process modeling refers to a systematic way of expressing product development activities and the influence of these activities on products so as to optimize the product development process, tools, and their supporting organization. A process model describes the following: (1) processes to accomplish product development, (2) interprocess relationship and process coordination, (3) the sequence of tasks, operations, and steps to perform in a process, and

(4) the tools used to accomplish each process, task, and operation. Items (1) and (2) express the macro view of the development process, while items (3) and (4) define the micro view and the engineering details of each process activity. Process management provides the abilities to track process activities, the interaction and coordination among these activities, the utilization of resources and time, and the assignment of roles and responsibilities defined in the process model.

One of the challenges of process modeling and management for concurrent engineering is to decouple the otherwise linear development activities so that they can be performed in parallel. Since product data and models are the results of process activities, process management and product data management must be applied in an integrated fashion. The management of the interaction and coordination between processes and product data is also a difficult problem, especially in a concurrent development environment. It requires mechanisms to describe and manage (1) how process tasks and operations influence the product data and models, (2) how modifications on a product model affect the related process activities, (3) who should be notified when product modification occurs, and (4) how to propagate changes to the affected product models.

### Product and Process Information Sharing

The essence of concurrent engineering is an integrated and collaborative process, where people in different disciplines cooperate to specify, design, and manufacture products through coordination, communication, and negotiation. The key to the collaborative process is a complete understanding of the product and process information that can be shared by all members of the product development team. Information shared in a concurrent engineering environment includes design and process knowledge, product data, and process status, which are distributed through various information repositories such as knowledge bases and databases, mail and message libraries, and file and document libraries in the form of texts, graphics, images, formulas, CAD data, and multimedia data.

An environment that allows information sharing should provide facilities for team members to (1) archive and exchange ideas, (2) access and send development status, and (3) relate development results to the work of other team members. Besides the above facilities, mechanisms for version control, access management, product model refinement, and

association maintenance (as discussed before) are also required for information sharing.

Product data standard and exchange also plays an important role in information sharing in a heterogeneous environment. The management of information, along with the transformation and extraction of application-specific information, fall under the control of product data management.

### Product Data Standard and Exchange

Concurrent engineering deals with highly heterogeneous environments. The differences in representations, structures, and contents of data in different applications often result in "islands of automation" and prevent product development from reaching the goals of concurrent engineering.

IGES (Initial Graphics Exchange Standard) was one of the early attempts in supporting product data sharing and exchange among different applications and organizations. It was conceived to provide a common file format for transferring data between CAD systems and application programs. However, because of geometric inaccuracy and numerical instability, it sometimes loses product characteristics during the data exchange process. Further, IGES does not support some of the product definition data that are required in product and process development.

PDES/STEP (Product Data Exchange Specification/STandard for Exchange of Product data) was developed as a comprehensive product model capable of describing product life cycle data independent of any particular systems. Besides product geometry, STEP supports a wide range of product data, such as configuration management information, manufacturing features, tolerance specifications, material properties, and testing specifications using high-level, feature-based, and object-oriented representations.

One of the major achievements of PDES/STEP is the use of high level information instead of raw data in information exchange. However, a set of data may represent different types of information that reflect various sources of knowledge. Hence, there is a need to convey the knowledge behind the information in the product model, encompassing the rules, principles, and rationales used in the decision making process during product development. One of the most promising enhancements of PDES/STEP will be the incorporation of knowledge into the product model to facilitate data interpretation and information sharing. This allows STEP appli-

cations to interpret product data consistently and thus enhances interoperability.

Both IGES and PDES focus on the standardization of product model data. However, product development involves more than product models. Therefore, the concept of product data standard and exchange should be broadened to meet the increasing needs in a concurrent development process.

### Database Management and Storage

Classical data models (e.g., relational, network, and hierarchical models) have been employed in modeling engineering data, which mainly involves the representation of entities and the relationships between the entities in an engineering environment. However, they are too primitive for modeling complex product and process data, and their levels of abstraction are too limited for capturing various types of information needed in engineering applications. Semantic data models were developed to alleviate these problems. Semantic models allow the relationships among high-level entities to be defined explicitly and semantically, although they may still rely on the classical models for internal data storage. Vendors of products in this area have typically developed proprietary methods for data storage in order to avoid the efficiency problem with the classical models in physical data access.

Recently, object-oriented data models have made a great impact on modeling objects and their behaviors. They are widely used in applications which demand features such as data encapsulation and abstraction, modularization and inheritance capabilities. Efficiency was a problem in object-oriented database systems. However, efficient access techniques and persistent object stores developed recently are mature enough to be used in primary object repositories. Industrial efforts in standardizing object-oriented concepts and languages and the evolution of SQL to support more and more object-oriented features reflect the increasing influences of object-orientation in the data modeling and database system areas. To support data management in concurrent engineering activities, further attention must be focused on transaction modeling, remote data access, distributed object environment (DOE), object linking and embedding (OLE), heterogeneity, and temporal management.

### OVERVIEW OF SPECIAL ISSUE

We received 18 submissions for this special issue. Each submission was reviewed by at least three re-

viewers. Based on their technical quality and relevance to the special issue, we selected 5 papers to be included in this special issue. Although these papers, even collectively, don't cover all of the aspects that we mentioned earlier, we hope this special issue represents a snapshot of the current research in this area and sparkles further research in this area.

The paper by Munch et al. presents a software system and architecture for integrated product and process management. The system, called EPOS, supports versioning, cooperating transactions, tasking, behavior modeling, activity planning and modeling. An example on software development is given to illustrate the concepts. The Sukits environment, by Westfechtel, describes a CIM manager to support product and process modeling. The data model supports versioning and configuration of engineering data and the process model uses a process net model to model the engineering processes, including the state changes and dependencies of the processes. Belkhatir and Melo describe the experiences with the Adele/Tempo project in developing a software engineering environment, where software objects and software development activities are modeled. The concept of role and temporal rules are introduced into the object-oriented model to handle, respectively, temporal constraints and multiple perspectives of objects in different stages of the software life cycle. The paper by Venkatesan and Davis proposes a meta-model approach to the problem of semantic heterogeneity in information models, which is inevitable in systems where data are generated and shared by different applications. A method for specifying the semantic mapping between information models is given. The authors illustrated the techniques with detailed examples in electrical CAD modeling. Patankar and Segev describe a method based on mobile objects to support CIM. The dis-

tributed nature of the architecture is particularly suitable for CIM.

We sincerely appreciate the efforts of the reviewers in reviewing the manuscripts. Their careful reviews are essential for the quality of this special issue.

*Dik L. Lee*  
*Hong Kong University of Science and Technology*  
*Hong Kong*

*Yuh-Min Chen*  
*National Cheng-Kung University*  
*Taiwan*

*Guest Editors*

Reviewers:

Donald J. Bagert, Texas Tech University  
 J. Cleetus, West Virginia University  
 Prasun Dewan, Purdue University  
 Pamela Drew, University of Science and Technology  
 Gary W. Fischer, University of Iowa  
 Farshad Fotouhi, Wayne State University  
 V. Jagannathan, West Virginia University  
 Gail E. Kaiser, Columbia University  
 Mark Klein, Boeing Computing Services  
 Alexander Kott, Carnegie Group  
 Andrew Kusiak, University of Iowa  
 Andrew R. LeBlanc, Clemson University  
 Qing Li, University of Science and Technology  
 Thu-Hua Liu, Chang Gung University  
 Toshimi Minoura, Oregon State University  
 Joe Mize, University of Science and Technology  
 Roger Morenc, SDRC  
 Ravi Rangan, SDRC  
 Chia-Hui Shih, National Cheng Kung University  
 Rao Vemuri, AMCT  
 Land S. Wimberley, Lockheed