

# Concurrent Engineering

<http://cer.sagepub.com>

---

## Implementation of STEP-Based Product Data Exchange and Sharing

Shen-Chou Yeh and Chun-Fong You

*Concurrent Engineering* 2000; 8; 50

DOI: 10.1177/1063293X0000800106

The online version of this article can be found at:

<http://cer.sagepub.com/cgi/content/abstract/8/1/50>

---

Published by:

 SAGE Publications

<http://www.sagepublications.com>

Additional services and information for *Concurrent Engineering* can be found at:

Email Alerts: <http://cer.sagepub.com/cgi/alerts>

Subscriptions: <http://cer.sagepub.com/subscriptions>

Reprints: <http://www.sagepub.com/journalsReprints.nav>

Permissions: <http://www.sagepub.com/journalsPermissions.nav>



## Implementation of STEP-Based Product Data Exchange and Sharing

Shen-Chou Yeh and Chun-Fong You<sup>1</sup>

*Department of Mechanical Engineering, National Taiwan University, Taipei, Taiwan, ROC*

*Received 13 September 1999; accepted in revised form 14 November 1999*

**Abstract:** Product data exchange and sharing is an important issue for miscellaneous information systems. STEP-based product data exchange and sharing provides a useful mechanism for implementing concurrent engineering and promoting the realization of CALS' environment. This paper implements a pilot system for STEP-based PDE (Product Data Exchange) system based on the requirement of product data exchange between and within enterprises. The system incorporates engineering information in the design and manufacturing stages and guarantees the consistency of the product data exchange and sharing. Complete development processes of systems, including AAM (Application Activity Model), ARM (Application Reference Model), and AIM (Application Interpretation Model), are illustrated herein. AAM utilizing BPR (Business Process Reengineering) approach is employed to develop ARM, which is then mapped to AIM models from part 41, part 42, part 44, AP 203, AP 214 in STEP. The system's database is based on the integrated AIM models in STEP to assure the capability for product data exchange and sharing. The kernel data structure is a combination of engineering information and product-oriented definitions. The data from miscellaneous information systems, such as CAD/CAM, MRP/MRP II, ERP, and PDM, in the design and manufacturing phase of a product's life cycle can be exchanged using this system. This paper also employs STEP to present an integrated product-oriented data structure to manage engineering information from different information systems.

**Key Words:** product data exchange, product data sharing, product data management, concurrent engineering, STEP.

### 1. Introduction

Product data exchange and sharing must assure the interoperability of product data from miscellaneous information systems by either defining standardized data models for representing product data [1–4] or implementing system architecture for reducing the barrier of product data exchange and sharing [5–8]. STEP standard provides a robust data structure for the exchange of product data models [9,10]. However, no information system currently has the capability to flexibly manage STEP-based product data from miscellaneous information systems. The data sources of different product information within STEP-based product data models are defined to integrate the information.

This project aims to define the product data models using EXPRESS language and setup the STEP-based product data repository for product data management and exchange for automobile manufacturing industries and their first-tiered suppliers. The STEP-based product data representation as illustrated in Figure 1 may come from miscellaneous information systems, such as CAD/CAM, MRP/MRP II, ERP, and PDM. Although additional information from one system can be attached and implemented, STEP cannot incorporate pieces of information from miscellaneous systems. The repository accomplished in this paper can amalgamate the data

based on STEP and a real STEP-based data repository to achieve product data integration. Distinct methodologies for product data exchange and sharing, including part 21, SDAI, COM, and CORBA, are discussed in the next section.

### 1.1 Review of the Implementing Conceptual Layer

Object-oriented language with UML (Unified Modeling Language) modeling methodology is proposed to equip the STEP-based repository system with EXPRESS data models, since EXPRESS language lacks the I/O capability for programming [11]. Object-oriented modeling language also has the benefit of being robust, flexible, extensible, direct, and reusable. The conventional means of establishing a STEP-based system is to use EXPRESS language to model the conceptual models, and then construct the application layer by object-oriented and EXPRESS-C languages as depicted in Figure 2. The conceptual data models are translated into an RDBMS (Relational DataBase Management System), merged into an ODBMS (Object-oriented DataBase Management System), or managed by an EXPRESS model management system, to construct the physical layer of the system.

### 1.2 The Physical Layer

The physical layer in Figure 2 contains the kernel database and network communication media structure. The ODBMS

<sup>1</sup>Author to whom correspondence should be addressed.

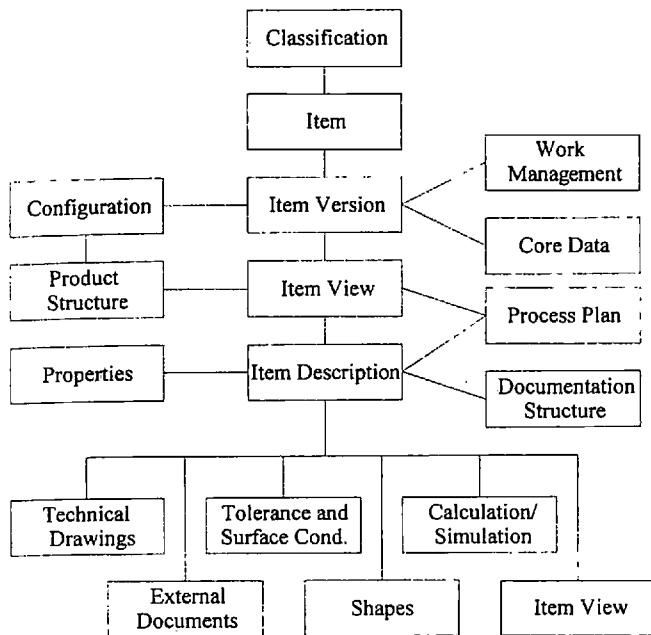


Figure 1. STEP-based product data representation.

provides better storage of information within the STEP-based system than the object-oriented conceptual data models because the latter affects the construction of the database schema. Best results can be obtained by employing the RDBMS after object-oriented modeling because it is compatible with relational databases and it will guarantee the effectiveness of the legacy systems data retrieval. The complexity of the information to be stored is key to choosing between the ODBMS or the RDBMS as the former is suitable for complexly related data, such as the PDM and CAD/CAM system, while the latter is suitable for large quantities of data.

The main concern that must be addressed when adopting RDBMS as the kernel of STEP-based system is the integration of legacy data. It is best achieved by ODBC (Open Database Connectivity) because it can incorporate legacy data by merging the schema of legacy data into the STEP-based system [12]. On the other hand, if the ODBC driver of an ODBMS is used, the data residing on ODBMS can also be shared by the RDBMS.

The physical layer of the STEP-based PDMS should be equipped with a data exchange, an external shared database, or a distributed database method to secure product data exchange and sharing. The data exchange method enables product data within a PDMS to be exchanged as physical files expressed in a plain text [13]. This method easily facilitates setting specifications, conducting tests and diverse system applications. The externally shared database method accesses a common database between companies on a contract basis. Data are copied from the PDMS of the companies into the common database. The distributed database method distributes the product data among the PDMS of companies that can concurrently share access [14–16]. The virtually shared database method may be the ultimate in product data sharing intended to create a concurrent working environment because although the data are physically shared on internet network, the database functions as a single unit.

The construction of a STEP-based distributed database management system is illustrated in Figure 3. The left side indicates the generation of the internal representation of EXPRESS codes, such as the migrated data models in Section 2. This can be used to construct the database, generate parser codes, and implement C++ codes. The C++ codes are compiled by an ODBMS to add the management functions into original C++ codes. At the same time, a schema dictionary

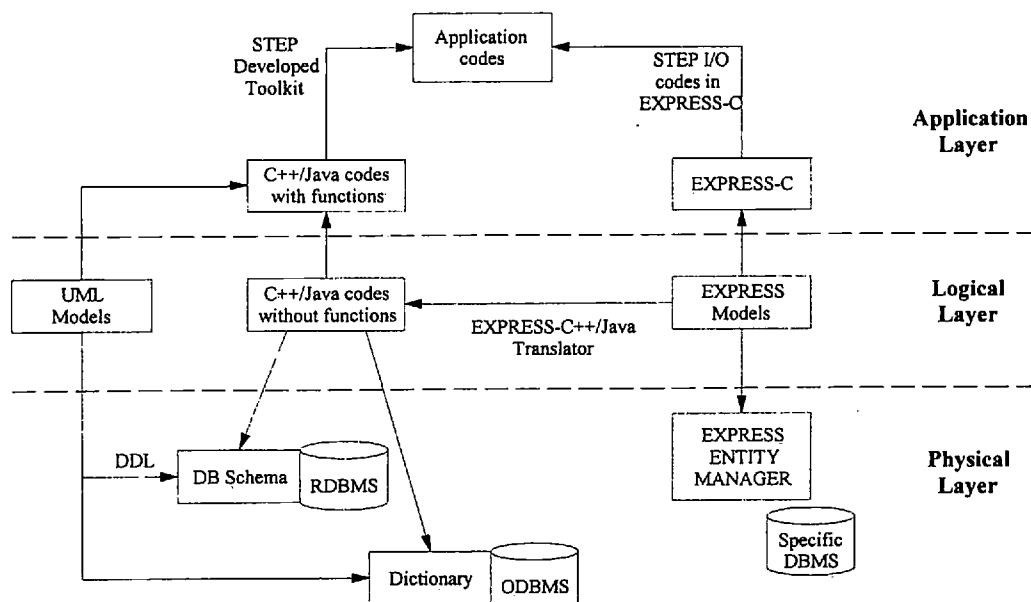


Figure 2. Approach for establishing a STEP-based system.

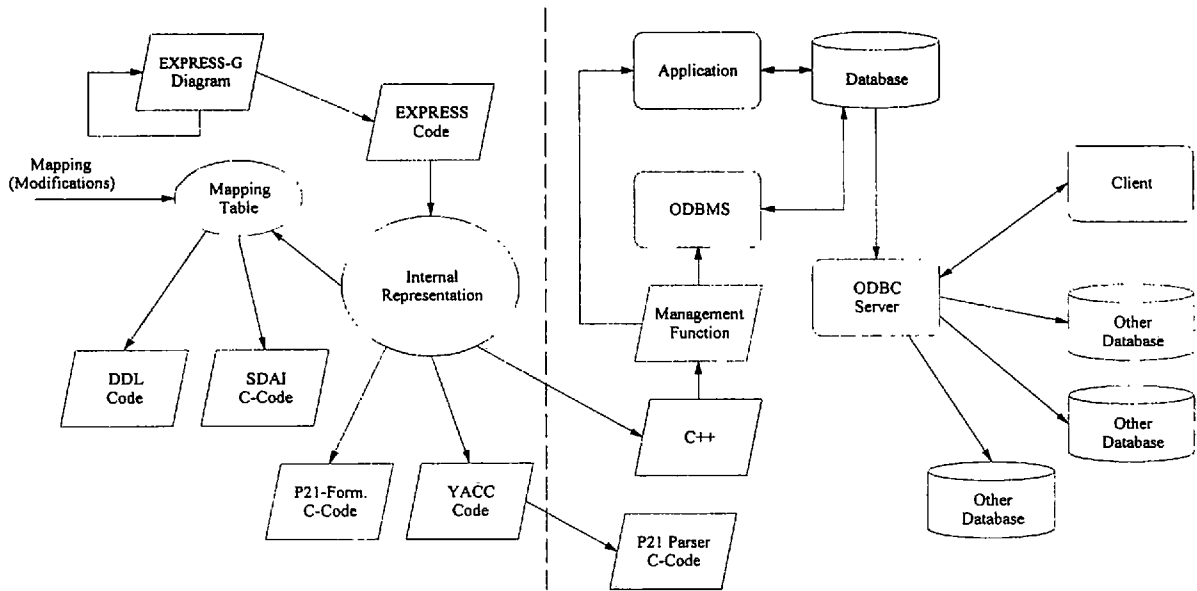


Figure 3. Approach for constructing STEP-based distributed management system.

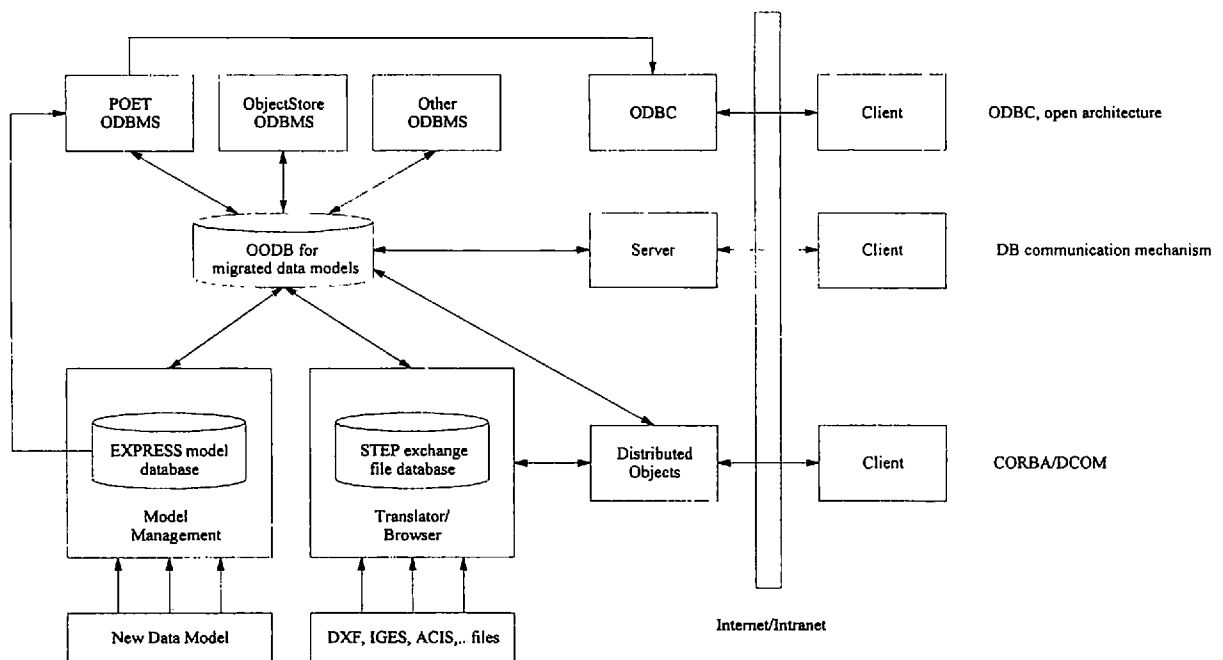
and its related codes are generated and managed by the ODBMS. The shared database environment is achieved with the ODBC driver provided by ODBMS [17,18].

Figure 4 presents three mechanisms of network communication, including an ODBC mechanism, a database communication mechanism, and a CORBA/COM mechanism, with their operation methods, that can all work within a STEP-based PDMS. The ODBC mechanism combines all information from the different systems into the unified data schema. Database communication relies on the ODBMS, while the most popular approach is the use of TCP/IP protocol to address the PDMS server side. The CORBA/COM mechanism

allows the distributed objects to communicate with each distribute the information of PDMS.

### 1.3 Review of Implementation of Application Layer

The application layer must connect the operations of the physical layer to the conceptual layer. The effectiveness of data management and compatibility with STEP are always opposite, because the implementation of STEP-based system using ARM or AIM indicates two different approaches [19]. Popular database preprocessor tools gradually solve



this phenomenon, so that prototyping applications can be rapidly developed and easily connected to the target database. The application or presentation layer is formed by referencing models in the conceptual layer to meet diverse requirements. A flexible modeling and better communication interface are the major issues for a STEP-based PDMS. They should be considered to adopt different requirements from distinct ventures. It emphasizes how to flexibly synchronize the conceptual models and the data structure and how to establish a network communication interface. Communication interfacing is important for a distributed PDMS to meet the need of managing and sharing information. Since the STEP-based system is an object-oriented system, most communication interface relies on Java, CORBA, or DCOM to achieve the distributed environment.

STEP utilizes SDAI to integrate the system by sharing the AIM product data of information systems such as CAD system, PDMS, etc. [20,21]. The SDAI can be regarded as an API (Application Programming Interface) to store and share the product data independent of the system's physical layer. The SDAI should not be used during the practical utilization stage since it is merely an interface to operate the database of EXPRESS language. The STEP-based data based on the application object level is managed by using HLDIAI (High Level Data Access Interface) that is a user-friendly interface with an application view. A STEP engineer can access the STEP database without an SDAI or AIM by using the HLDIAI (High Level Data Access Interface). A STEP engineer shall construct the STEP data exchange system according to its kernel data schema that is a data structure definition of information systems.

## 2. System Architecture

The bolder lines in Figure 2 indicate the methodology applied to develop the STEP-based repository system in this paper. The integrated product data models in STEP are created by a UML modeling tool that generates programmable C++ codes and is translated into an object-oriented database, *POET*. The generated C++ codes are used to develop the applications in the presentation layer, while the object-oriented database is ready for storing the instances of the data models. Any change in the conceptual models will immediately alter the data models in the physical layer. An OO management mechanism, such as an OODBMS or serialization mechanism of object-oriented language, is used to develop STEP-based PDMS to accomplish the management of STEP-based information, rather than using SDAI or HLDIAI.

A STEP-based product data repository is developed to fulfill the requirements of product data management, exchange, and sharing. The product information models follow ISO-10303, STEP standard, while data exchange, sharing, and management methodology follow the methodologies defined in STEP. The repository can exchange product data and manage the CAD models and technical information as illustrated in Figure 5. The functional components include an

Internet-based exchange subsystem, applications, utilities, and a repository database with an access apparatus.

Defining the product data models using STEP standardizes the information representation to assure the implementation of information sharing. The STEP files export/import functions in the system can be used to generate the exchange files. The accompanying utilities and applications can be used to demonstrate the integration and management capability of STEP-based repository.

The best possible model of the STEP system must be created herein since the quality of the models affects the implementation of the system. A subset data models based on AP214 is used as the kernel information models of the information within the development and manufacturing stages in the target industry [22]. This section describes the development of each task, including modeling the information models, developing kernel database, developing the applications, and developing the utilities. The functionality and implementation methodologies used in the pilot system are discussed in detail.

### 2.1 Overall System Architecture

The repository system should have the following components to manage product information effectively, database management system with database interface, repository system with access interface, utilities, and applications, as illustrated in Figure 6.

The database interface provides the registration function, the manipulating functions of application interfaces and meta-data, and the managing functions of user information and database engine. The repository interface provides the vaulting, formatting, and browsing functions of all executing threads. The utilities, including system administration and functions management, provide and maintain information infrastructure and operations. The user registration, management, and verification are also supported by the utilities. The applications provide the end-user with a working interface to fulfill their requirements. The imported and exported information follows the subset data models of AP214 in STEP [23]. The information retrieval and exchange of the applications in the system, including product information and CAD models, follows ISO-10303 Part 20 serials like Parts 21 and 22.

### 2.2 Modeling the Information Models

The conceptual data models of the repository system must be modeled to define the data schema. The models include AAM, ARM, AIM, and other models with specific programming language. AAM is used to model the activity models within design and manufacturing stages while the data managed within the activity is modeled in ARM. ARM is then mapped onto the standardized data models in STEP. AIM is then translated into the models in C++ for implementation by the object-oriented language.

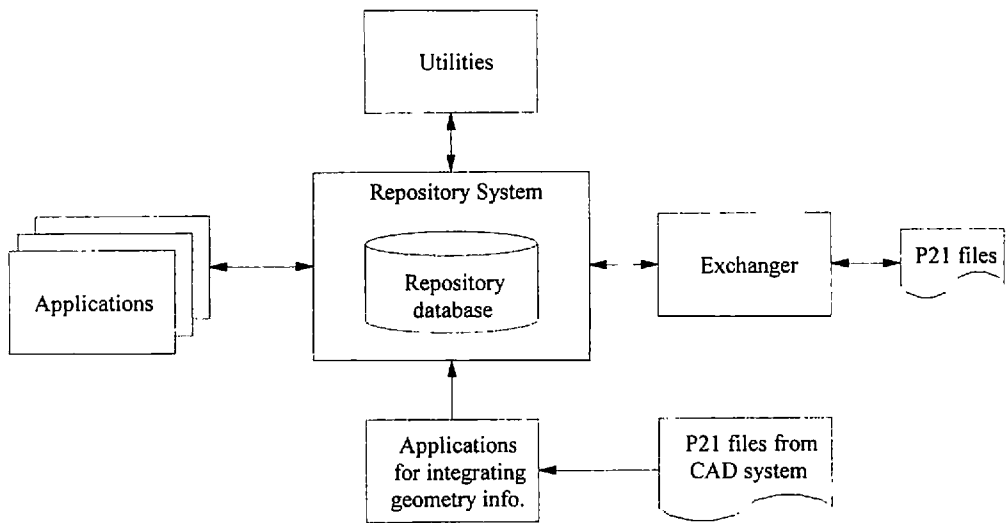


Figure 5. Overall architecture of the repository system.

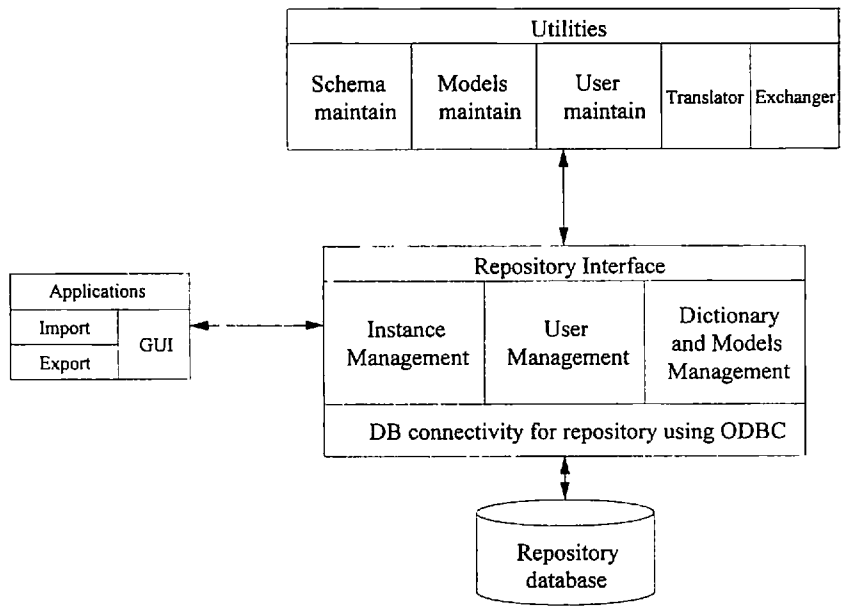


Figure 6. Major components of the repository system.



### 2.2.1 MODELING CONCEPTUAL DATA MODELS

The conceptual data models of the repository system include AAM, ARM, and AIM. All conceptual data models of the tables and CAD models in the pilot system are represented using EXPRESS language and illustrated by EXPRESS-G. The application objects of CAD models and the ten sets of technical information, including engineering related documents, material, specification, expressed with EXPRESS models are defined and represented by EXPRESS-G notation. The context of the conceptual data models can be classified as the information models of CAD models and technical documents. The data models of CAD models in AIM follow AP203 in STEP, while the ones in the technical documents follow AP203, 214, and Part 40 serials parts. The AAM captures the process information models in the design and manufacturing stages. AAM utilizes IDEF0 modeling methodology. The ARM employs the data models in the venture, and EXPRESS language to model the information. The AIM, derived by mapping ARM with STEP standard, represents STEP-based standardized industry data models in this paper.

### 2.2.2 MIGRATE DATA MODELS FOR TECHNICAL INFORMATION AND CAD MODELS

The CAD geometry information model represents the product-oriented data structure as presented in Figure 1. The technical information, or product-oriented data structure, is represented by miscellaneous industrial table sheets. An item view is used to combine the technical information with the CAD models so one can track the information through the entire entity. With this mechanism, we can reduce data exchange, reformatting, data management, design history, document barriers [24] while implementing concurrent engineering.

### 2.2.3 IMPLEMENTATION MODELS

All implementation models used to define the data models of the repository and the applications follows UML for OOA and OOD of the applications. The Data Models (I.E. Static Models) and Sequence Models (I.E. Dynamic Models) of the applications are used to manage the data structure and operating procedures. Since the conceptual data models of product information are specified by EXPRESS language and EXPRESS-G graphical notation, the EXPRESS language is translated into C++ format for use as the reference models for the data schema of the repository system.

## 2.3 Developing Database

The database interface of the repository system manages the database according to conceptual models. The information technology will not be restricted by specified methods since the applications are not directly connected to a database management system. The relationship of all objects with a list of functions and attributes of classes can be obtained using UML modeling language to specify the API. The API of

the repository comes from the database API, which is specified by UML. EXPRESS-C models and IDL (Interface Definition Language) can also be used to handle objects within the CORBA architecture. The applications of the repository system use conceptual models to access the information stored in the database through a set of interfaces. The application interfaces can be from the lowest level API, which is the kernel application interface, to the highest level API, which is the application-oriented API. The application-oriented API can use the information models to map the corresponding instance to find the necessary information. The API can be classified into four kinds of API: Kernel API, Dictionary API, Classification API, and Object API.

The relational database management system is used by the repository system as the kernel database. A DDL (Data Description Language) is generated and used to setup the table schema and data schema of the RDBMS. The applications can access information of the tables or CAD models through the RDBMS. With the EXPRESS-to-DDL translator, the EXPRESS language can be translated into the corresponding DDL. The definition and relationship between objects can be mapped into the corresponding data schema. Therefore, the generated DDL can be used to get the definition and relationship of the object-oriented models in the RDBMS.

The database access operations of the applications use the ODBC Bridge of the RDBMS to communicate with the database. These access operations encapsulate corresponding SQL statements. For example, the "Add New" operation will encapsulate the INSERT statement of SQL, while the "Edit" operations will encapsulate the UPDATE statement of SQL. All of these operations will be packaged as a set of API.

ODBC is used as the standard access interface of database to amalgamate with other RDBMS. The applications utilize the ODBC Bridge of the RDBMS to send SQL statement to easily access the object information of the database through ODBC interface. The applications can use the same API to access data from distinct database to amalgamate different RDBMS, even ODBMS with an ODBC bridge.

## 2.4 Developing Applications

The configuration management, engineering data management, geometry, and ordering applications are all developed according to the AAM models advanced herein. The configuration management application can be used to manage the product configuration data of the repository. The content of the configuration data consists of the product definition as well as engineering change request, suggestion, and engineering change issue. The engineering data applications can be used to manage the repository's engineering data, such as material management and test report. The geometry application is used to display and render the 3D CAD model of the product, while the ordering application can be used to manage the data from the purchase department.

The applications use kernel API encapsulating the SQL of the database to access the kernel database. The API of the re-

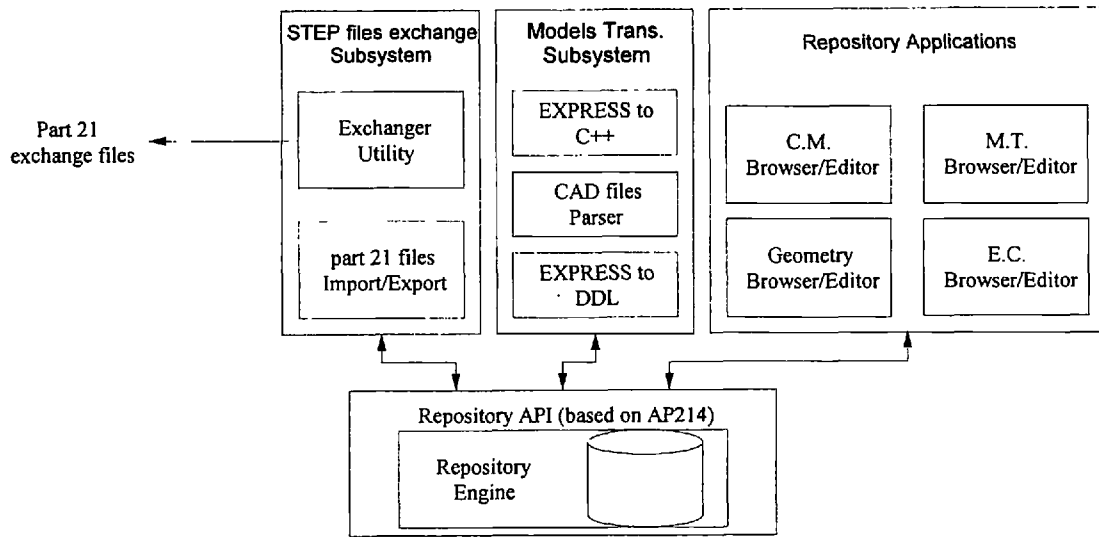


Figure 7. Functionality of the repository system.

pository accesses the kernel database because the API is packaged as the API of the repository. The export/import subsystem translates information from/to the repository with a STEP-based approach. The subsystem can be a stand-alone application or a module of an application. It can be classified as an export/import subsystem or a CAD model export/import subsystem depending on its function. The major functionality of the subsystem is translating the information of the tables or CAD models to and from part 21 formats.

The browser and editor are the fundamental functionalities for each application of the repository system, as illustrated in Figure 7. The application instances of the physical objects and the relationship between these objects are displayed with different views. The structural browser and editor divide the graphical user interface into two partitions: a rapid display of the information items on the left, and the content of the item selected on the right. Different form views with proper controls should be used to display its information for distinct applications of the tables. A browser for CAD models is also an important issue for application. A browser for AP203 CC6 is developed to solve the inconsistency for STEP files of distinct CAD systems. An integrating system is also developed to move the information from the CAD systems and the repository system.

## 2.5 Utilities

The system administration, application management, and auxiliary tool utilities can be used to help develop and maintain all information in the repository system. The major functions include:

1. To generate and manage the product data models of the repository system
2. To generate and manage the data schema in the repository system

3. To record the user's information for managing the user's privilege
4. To translate information from and to proper format, such as EXPRESS to C++ codes, and translation between CAD systems

The exchange utility is developed in Java to secure usage across distinct platforms between industries to transfer part 21 files. The exchanger can monitor the progress of files and examine the completeness of the exchange files. It can be used in either server or client modes for switching the roles for application requirement. The transaction information could also be saved to trace the user's login. The translators used in this paper adopt one of the UG and CATIA systems to avoid the information loss.

## 3. Implementation

The repository system is tested in two manufacturing industries, industry A and industry B, to verify the system architecture. Industry A has the entire repository system, while industry B, the first-tiered supplier of industry A, has just the applications and exchanger. The procedures are discussed separately because the processing of tables and CAD models are not the same.

The CAD translators used for the test are UG and CATIA translators, which are the major CAD systems used in the two industries. The operating system used for the UG translator is Microsoft NT 4.0, while the one used for the CATIA translator is the IBM RS6000 UNIX. Both of these CAD translators can translate the native CAD models into standard formats based on AP203 and AP214. The translators for AP203 and IGES 5.3 of the two CAD systems are used to translate solid and surface CAD models to verify the CAD models translation. The exchanger utility runs on both



PC Windows NT 4.0 and IBM RS6000 UNIX environments because of the system environments of the industries. The application scenario is the co-design situation between these two industries. Industry A employs a conceptual design or engineering change case, and then transfer related information, including CAD models and related technical information, to industry B. Industry B evaluates the conceptual design or engineering change and then transfers their designs or suggestions for that case. The same process is repeated as the final design is approved and the manufacturing and ordering information is readied for production. The screen dumps for the developed repository systems and exchanger are illustrated in Figure 8.

### 3.1 Scenario 1

Industry A translates the engineering data and CAD models of a product and then sends them to industry B. Application A includes configuration management, engineering

data, geometry, and ordering applications as illustrated in Figure 9. Each application has its corresponding conceptual data models using EXPRESS, formed by a serial of process from AAM to implementable models in Section 2.2. The final models are translated and built into the Oracle databases of the repository system. The databases utilize the data structure, schema, and definition of the migrated data models to assure data consistency. The applications of the repository can then be developed based on these databases.

Geometry information is included in the product information, since the product information is linked and managed by PDMS. The part 21 files of the CAD models are generated from the translator of UG and CATIA. The part 21 files in industry A are generated by the translator of the UG System. A STEP CAD model viewer for AP203 is developed to render the 3D models to browse the CAD models in part 21 formats. The part 21 files are parsed and stored in the corresponding tables of the databases in repository to store the geometry information in part 21 files. Both of these operations are sup-

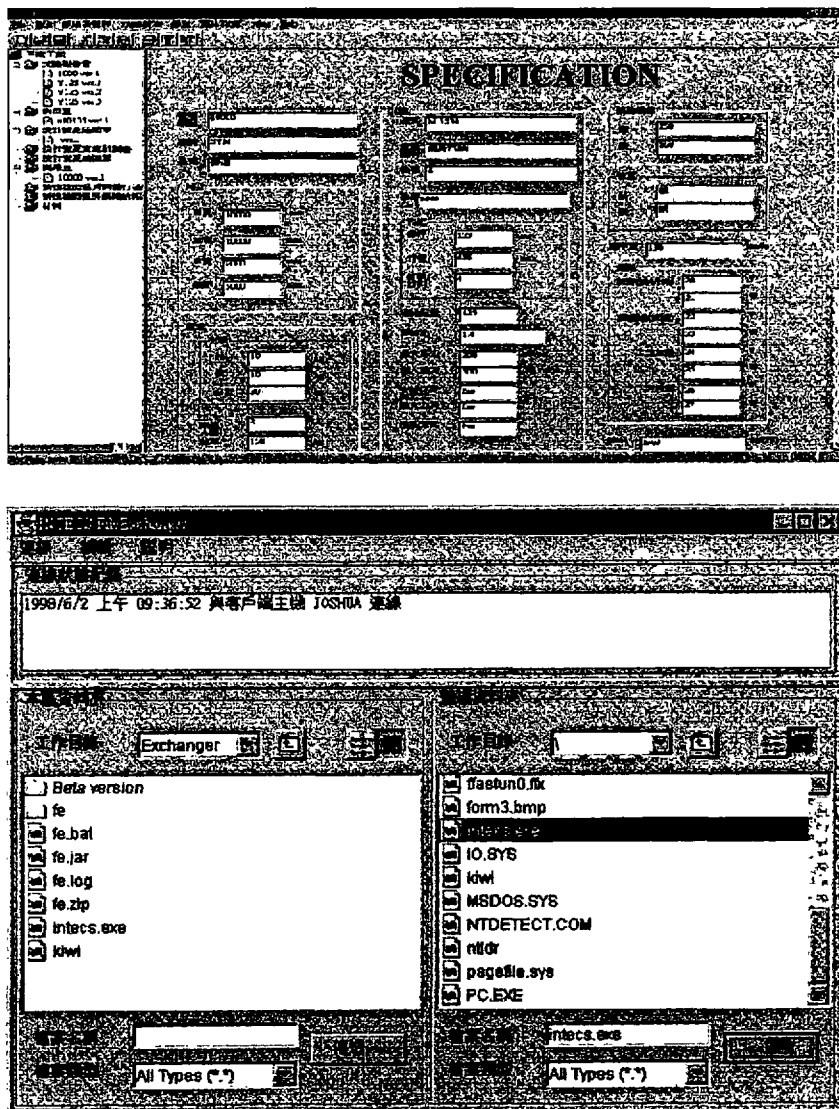


Figure 8. Screen dumps for the developed repository system.

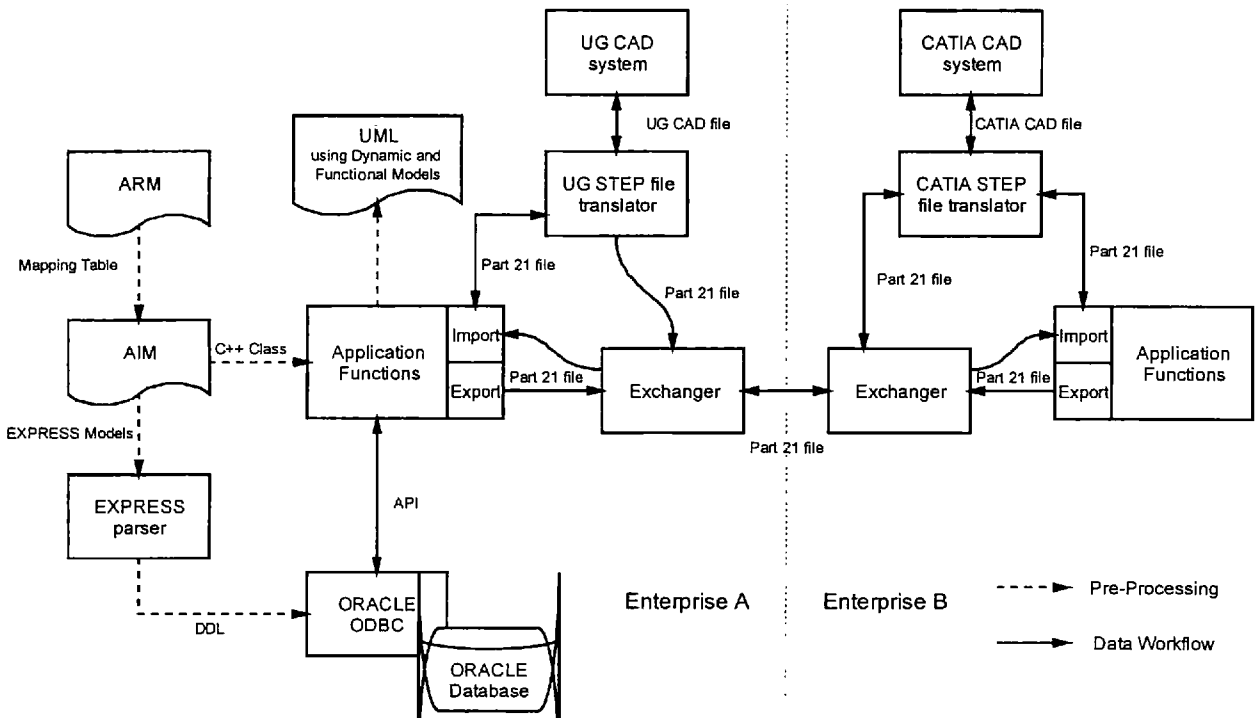


Figure 9. Working process of the applications of the repository system.

ported in application A since the source of the geometry information can be part 21 files in the directory of the disk or databases in the repository.

### 3.2 Scenario 2

Industry B receives the engineering data and CAD models of a product, and then sends back the design information after design or redesign. Application B consists of the configuration management, engineering data, geometry, and ordering applications of industry B, as illustrated in Figure 9. Although industry B has no data repository, the information exchange and management can also be accomplished by using part 21 exchange files. Application B imports and exports the part 21 files of the engineering data and CAD models from industry A. Since the conceptual data models of each table are pre-known to these ten sets of applications, the

data from application A can be received and interpreted by application B directly by mapping the same migrated models. This procedure is also applied to the data exchange of CAD models from application A.

New part 21 files of the CAD models from the CATIA system are generated and then passed to industry A through the exchanger after the redesign or engineering change of the CAD files. The STEP viewer is used to browse the part 21 files of the CAD models in application A as in scenario 1. The part 21 files of the CAD models are generated by the translator of CATIA system.

## 4. Conclusion

Product data exchange and sharing mechanism can be accomplished based on the repository system. Figure 10 illus-

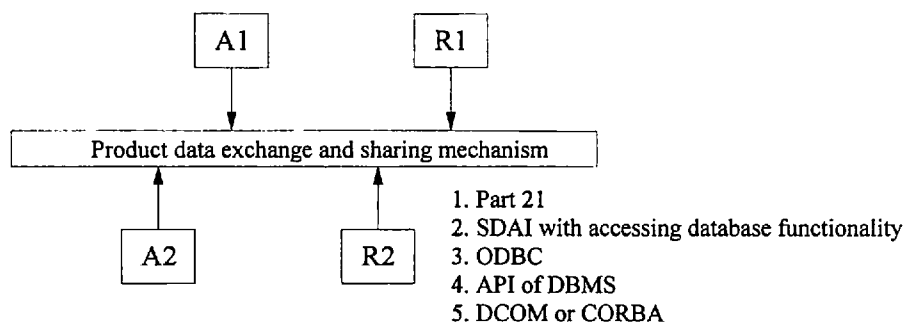


Figure 10. Product data exchange and sharing mechanisms for STEP-based repository system.

trates the summary of applicable methodology of the STEP-based repository system. R1 and R2 are the full set of the repository system, while A1 and A2 represent the application of the repository system without a repository database. The data exchange and sharing scenarios can be R-R, R-A, or A-A. The data exchange and sharing mechanism are listed in Figure 10: part 21, SDAI with accessing database functionality, ODBC, accessing API of DBMS, COM or CORBA. The mechanism using part 21 relies on the quality and flexibility of import/export functional module to assure the quality of the data exchange and sharing. The mechanism using SDAI relies on the flexibility of information models management and translation to assure the quality of the data exchange and sharing. The mechanism that employs ODBC relies on the translation and consistency of information models from the object-oriented data structure and relational data structure to assure the quality of the data exchange and sharing. The mechanism that manipulates the API of DBMS relies on the flexibility of information models management and translation. The mechanism using COM or CORBA relies on the development of services on distributed environment to assure the quality of the data exchange and sharing. These mechanisms co-exist to accomplish the data exchange and sharing between and within industries.

The data repository developed herein is used to setup the management system for technical document and CAD models. The information exchange and sharing is easy to process because the information representation and management is based on standardized product data models. The international standard, STEP, secures the implementation and extensibility of the repository system. The system integration of miscellaneous systems, such as MRP, MRPII, PDMS, ERP, can be ensured with the standardized data models. The same application will be extended to other areas of the manufacturing industries to organize their virtual enterprise environment and promote their productivity.

## References

1. K. Hodota, "Product and information life-cycle with product configuration database system on Internet," in *21st Century Commerce & CALS EXPO International Conference '97*, 1997.
2. A. McKay, F. Erens, and M. S. Bloor, "Relating product definition and product variety," *Research in Engineering Design*, vol. 2, pp. 63–80, 1996.
3. O. A. Suarez, J. L. A. Foronda, and F. M. Abreu, "Standard based framework for the development of manufacturing control system," *International Journal of Computer Integrated Manufacturing System*, vol. 11, pp. 401–415, 1998.
4. J. K. Wu, T. H. Liu, and G. W. Fischer, "An integrated PDES/STEP based information model for CAE and CAM applications," in *The Second International Conference on Automation Technology*, 1992, pp. 179–187.
5. Y. M. Chen and Y. T. Hsiao, "A collaborative data management framework for concurrent product and process development," *International Journal of Computer Integrated Manufacturing System*, vol. 10, pp. 446–469, 1997.
6. R. Jordim-Goncalves et al., "Implementation of computer integrated manufacturing systems using SIP: CIM case studies using a STEP approach," *International Journal of Computer Integrated Manufacturing System*, vol. 10, pp. 172–180, 1997.
7. Y. V. R. Reddy et al., "Computer support for concurrent engineering," *Computer*, vol. 26, pp. 12–16, 1993.
8. G. L. Smith and J. C. Muller, "PreAmp—a pre-competitive project in intelligent manufacturing technology: an architecture to demonstrate concurrent engineering and information sharing," *Concurrent Engineering: Research and Applications*, vol. 2, pp. 107–115, 1994.
9. F. Chaxel, E. Bajic and J. Richard, "Mobile databases nodes for manufacturing information management: a STEP based approach," *The International Journal of Advanced Manufacturing Technology*, pp. 125–133, 1997.
10. I. Bailey, P. Spiby, and J. Vuoskoski, "Pilot implementation of process plant lifecycle functional data exchange conforming to STEP AP-221, Part 2: Logical & physical data models," *ES-PRIT-IV, PIPPIN*, 1997.
11. P. Spiby and D. Sanderson, "Introduction to EXPRESS 2," *ISO SC4/WG10 Document N58*, June 1998.
12. L. Bernosky, "Turning legacy data into enterprise information," in *21st Century Commerce & CALS EXPO International Conference '97*, 1997.
13. D. An et al., "A product data exchange integrated structure using PDES/STEP for automated manufacturing application," *Computers Industry Engineering*, vol. 29, pp. 711–715, 1995.
14. Y. Tanaka et al., "Product data sharing among distributed heterogeneous PDMs," in *CALS EXPO '97 International*, 1997.
15. S. D. Urban et al., "A heterogeneous, active database architecture for engineering data management," *International Journal of Computer Integrated Manufacturing System*, vol. 7, pp. 276–293, 1994.
16. S. D. Urban, J. J. Shan, and M. T. Rogers, "Engineering data management achieving integration through database technology," *Computing & Control Engineering Journal*, pp. 119–126, 1993.
17. D. L. Spooner, "An object-oriented product database using ROSE," *Journal of Intelligent Manufacturing*, vol. 5, pp. 13–21, 1994.
18. S. K. Ke and S. C. Yeh, "The pilot system for STEP-based product data exchange," in *CALS/EC EXPO '98 Japan*, pp. 197–206, 1998.
19. S. Rahimifard and S. T. Newman, "A methodology to develop EXPRESS data models," *International Journal of Computer Integrated Manufacturing System*, vol. 9, pp. 61–72, 1996.
20. R. N. Botting and A. N. Godwin, "Analysis of the STEP standard data access interface using formal methods," *Computer Standards & Interfaces*, vol. 17, pp. 437–455, 1995.
21. A. Goh et al., "A study of SDAI implementation on object-oriented databases," *Computer Standards & Interfaces*, vol. 16, pp. 33–43, 1994.
22. H. Scheder, "Requirements of car manufacturers for product data management in an extended enterprise," *STEP-Forum 97*, 1997.
23. ISO, 1997, *ISO 10303-214 CDII—Core Data for Automotive Mechanical Design Processes*, 1997.
24. B. Prasad, *Concurrent engineering fundamentals*, Vol. 2, Integrated product development, Upper Saddle River, N.J., Prentice Hall, 1996–1997, pp. 367–375.

**Chun-Fong You**

Chun-Fong You, Associate Professor in the Department of Mechanical Engineering, NTU, got his Master's and Ph.D degrees from Cranfield Institute of Technology, UK. His professional specialties include Solid Model Systems, Computational Geometry and the Integration of CAD/CAM.

**Shen-Chou Yeh**

Shen-Chou Yeh received his Master's degree in Mechanical Engineering from NSYSU. He is currently a Ph.D student at the NTU working in the area of product data exchange, sharing, and management using STEP. His research interest is in the enabling technologies for STEP-based information system and developing and implementing STEP-based manufacturing industry standard in Taiwan.