Product data exchange using STEP

C. Demartini, S. Rivoira, A. Valenzano
CENS and Dip. di Automatica e Informatica, Politecnico
c.so Duca degli Abruzzi, 24 – 10129 Torino (Italy)
tel. +39-11-5647016, fax. +39-11-5647099

Abstract

Product data technology and the related international standards have obtained increasing interest both in the academic and industrial environments in the last decade. Today several software tools are available to exchange and share product data information according to the ISO 10303 specification. This paper presents a case study for transforming the information model and data currently used in an industrial organisation (legacy system) into a subset of the Configuration Controlled Design (AP203) data scheme defined in STEP. This mapping is important because it allows the user to follow a graceful migration path to the STEP technology and gives her/him the ability to exchange product information in a standard form, while preserving data values already present in different formats inside the legacy system.

Keywords

Product data technology (PDT), STEP, Express language, product data modelling

1 INTRODUCTION

The exchange and sharing of product data between different enterprises or inside the same organisation is becoming a primary need in a number of production environments all over the world. In particular, the evolution of the markets into new global scenarios is rapidly changing the relationships in the supplier-producer-customer chain, and new standard technologies are demanded to exchange product information between partners at different stages of the production cycle, beginning with the conception and design of a product right up to the moment when it has to be disposed of.

The ISO 10303 standard collection (ISO 10303-1, 1995), better known as STEP, is the most important standardisation effort in the last decade aimed at defining rules and methods to manage product model data. Besides defining structures and relationships in the application data models, ISO 10303 also includes the specification of the formal methods used to describe the models themselves such as, in particular, the EXPRESS data description language (ISO 10303-11, 1995) that is extensively used to describe several parts of the standard documents.

STEP maintains a clear distinction between the conceptual model of data, its physical implementations and the different views that applications have of the model itself. This logical separation is enforced by the standard documents organised in several autonomous classes identified by a suitable numbering scheme. In particular the conceptual model is modular in nature and is built up of an (expandable) set of constructs called *integrated resources* that are used to build the application views of the data.

Application views are called *application protocols* (APs) in ISO 10303 terminology and consist of formal models that specify how application requirements, for the activities within the scope of the protocol, are satisfied by means of the integrated resources. Application protocols are designed according to the needs of specific fields: so, even though a relative small number of APs was included in the initial ISO 10303 release, several new APs are currently being developed so they can be added to the standard and involve, for instance, different application areas in the automotive, ship building, avionics, electronic and electrotechnical industrial scenarios.

The physical views of the conceptual model are obtained by means of the so-called *implementation forms* (IFs). An IF specifies the mapping between a formal description of the data model (that is a specification in EXPRESS language) and a particular physical format for exchanging and/or sharing data items. In this paper we are particularly interested in the "part 21" (ISO 10303-21, 1995) implementation form, since it represents the most common way to exchange STEP clear text data files between heterogeneous systems.

Two different STEP implementations are able to exchange information if and only if they adopt the same data model(s), or, in other words, if they comply with the same application protocol(s). This requirement is more critical than reaching an agreement on the data transfer syntax. In fact, very early on the existing legacy data have to be converted and organised according to ISO 10303 standard models, that specify uniquely the meaning of the various data entities and their relationships.

Even though there are already good standard specifications for product data modelling, the user is faced with the problem of how he/she can migrate to the product data technology (PDT) architecture without wasting the significant investments available in proprietary hardware and software and the existing (valuable!) data already present in his/her own legacy systems.

This paper presents a case study on how this graceful migration can be achieved. By modelling the legacy information system with the formal techniques of the ISO 10303 standard, it is possible to map the entities that are present in the proprietary data structure onto a coherent subset of an ISO 10303 application protocol. Once

the mapping has been identified, proprietary data can be converted according to the standard model rules so that the user can benefit from all the advantages offered by the STEP architecture.

Figure 1 Mapping of product data.

Transformation of product data is conveniently supported by mapping tools that operate on EXPRESS representations of two different data models as shown in Figure 1. The mapping specification describes how the entities in the source model have to be transformed in order to obtain the entities in the destination model.

Data models are represented in standard EXPRESS language, while the mapping specification is usually written using such languages as EXPRESS-M (Bailey, 1993) or EXPRESS-C (ISO PISA, 1994). These languages extend the data definition language EXPRESS with constructs to model processes, events, activities, transactions, build up execution models and so on. Then, the mapping program resulting from the compilation of the mapping specification makes it possible to read a STEP physical file of data which conforms to the source model and transform it into a STEP physical file of data which conforms to the destination model.

The paper is structured as follows: section 2 describes the model of the legacy system developed using the same formal methods adopted in STEP. Section 3 deals with the subset of the application protocol AP203, which is relevant in order to perform the mapping between the proprietary and the standard data models. For the sake of simplicity we will present a simplified version of the model which is sufficient to understand our work without boring the reader with a lot of unnecessary and, to some extent, confusing details.

2 LEGACY SYSTEM MODEL

A preliminary step in our project was the development of a model of the legacy system according to the formalism and conventions adopted in the ISO 10303 standard. This aspect should not be underestimated since usually the structure of data and procedures used in an industrial organisation depends heavily on the evolution of the different departments and plants of the organisation itself.

It should not be surprising, therefore, that information concerning a single product be kept in a number of different places (archives), managed by different (incompatible) systems and that often there is not even a complete and precise description of the meaning of the data items, their location and their consumers/producers !

A detailed and punctual analysis of the activities within the organisation, carried out with the fundamental contribution of experts from different departments (i.e. design, engineering, production, management departments) is then necessary to develop a data model such as the one depicted in Figure 1. The reader is reminded that, for the sake of simplicity, we will introduce in this paper only that subset of the legacy system model, which is useful to illustrate the results of our work.

The (part of the) model in Figure 1 is specified in EXPRESS-G and, according to that formalism, it consists of five entity objects and of their attributes and relationships.

In particular, the key point here is that the central role in the whole product data organisation is played by drawings and dash-number (part) elements, while the standard models developed for STEP, such as the application protocols AP203 (ISO 10303-203, 1995) and AP214 (ISO 10303-214, 1997), are based on parts and assemblies.

The mapping between the systems has then to take into account these different points of view, while trying to align the entities in the two models.

The abstract *component* entity in Figure 1 is used to describe product components independently of their origin. A component, in fact, is either a *standard_part* or a *dash_number* object. Standard parts are used to take into account those elements, which are needed to assemble a product, but are not produced inside the organisation. A standard part can be simply a screw, a bolt or a more complex object (such as a pre-assembled mechanical subsystem) but in the legacy system it can be considered as a basic element described by a textual specification.

Figure 2 EXPRESS-G specification of the legacy system model.

On the other hand, a *dash_number* is a product part designed and produced inside the organisation, it is given a suitable identification number and is associated with a drawing.

Both standard parts and dash numbers have common attributes such as a description and a part type that are taken into account in the definition of the abstract component entity. Instead, a particular situation occurs when the *item_number* attribute is considered; this kind of information, in fact, is not present in this form in the legacy system and must be obtained by concatenating the *drawing_number* and the dash number elements, so as to build up a unique component identifier.

A complex product is obtained by assembling several parts together. The connection between an assembled part and a component is modelled by the *structure* entity, which represents an assembly relationship. A structure has two main attributes to link an assembly to a component: while the former can never be a standard part (standard parts are considered “elementary” components independently of their complexity), the latter is necessarily a dash number element.

Note that the inverse is not necessarily true: a dash number can be either an assembly or an elementary part.

In this way a product corresponds to a structured organisation of structures: if N composite components are needed to build up a product, $N-1$ structure entities must be used to specify the inter-component connection relationships. Structures can be used “in parallel”, that is, to specify assembly relationships among components at the same level, or hierarchically to connect an assembled subsystem to a higher level element.

A structure also contains an *effectivity* attribute used inside the target organisation to define a kind of limit to the validity of an assembly or of a change to the usual structure of a product.

As mentioned above, a component is associated to a drawing. In the legacy system model each drawing can in turn represent several dash-numbers: this is outlined in Figure 1 by the inverse attribute *represents* that consists of a set of one or more dash-numbers. Obviously each dash-number is associated to one and only one drawing.

In addition each drawing has other important attributes: besides a unique number and a textual description, in fact, this entity is assigned a security classification field and a project code.

The legacy system model depicted in Figure 1 corresponds to a textual EXPRESS specification automatically processed by the software tools used to check the correctness of the model itself, or to verify that a physical file encoded according to the ISO 10303 standard rules (that is a “part 21” file in our case) be conformant to the STEP specifications.

Figure 2 shows a visual representation of some data items described according to our Express legacy system model and obtained by means of the Karlsruhe University ECCO toolkit (Staub, 1997).

Figure 3 Graphic representation of legacy data entities.

Each box in the picture represents a different entity: in particular the entities numbered 10, 15, 17 and 14 are abstract components represented by dash number objects (entities 10, 15, 14) or by standard parts (entity 17).

The structure entities marked 12, 18 and 16 specify the assembly relationships for components, thus, for instance, component 15 is the assembly for both structures 12 and 18, but it also represents a constituent part for the assembly described by structure 16. Finally drawing entities can be associated to a single component (entity 11) or to multiple parts (entity 13).

The physical text file containing the same data instances of Figure 2 has obviously a more concise form, suitable for data exchanges. The file structure, in fact, conforms to the ISO 10303-21 specifications that are used to get a generic data model written in EXPRESS to correspond with a clear text encoding of the data item values. Instances are numbered progressively to distinguish them uniquely, while their description contains the name of the entity type and the ordered list of the attribute values.

3 CONFIGURATION CONTROLLED DESIGN MODEL SUBSET

The AP203 application reference model (ARM) describes the structure of a product in terms of *parts*. A part entity is an element produced or employed in a production process and the application protocol defines attributes to describe its type (detail, assembly, customer supplied material etc.), name, version, whether the part is a “standard” item or not and so on. Then, for our purpose, each part instance can be made to correspond with a single drawing-dash number pair.

However, this mapping is to some extent complicated by the fact that each ARM entity is built up by a network of integrated resources constrained to satisfy the

information requirements of the target application (in our case the “configuration controlled design” application requirements). Figures 4 and 5 show the EXPRESS-G representation of some of the integrated generic resources (ISO 10303-41, 1995) used in the AP203 application interpreted model (AIM) to describe the ARM part entity. In practice, each part is mapped onto a *product* resource, while its attributes correspond to attributes of either a *product* or *product_related_product_category* element. The latter is a subtype of the *product_category* entity which is used, for instance, to describe classes of products such as mechanical parts, electrical parts and so on and takes into account the part classification.

Each part (product) can be associated to several *product_definitions* used to identify the part itself in different application contexts (i.e. physical design, functional design, engineering etc.) that are grouped by means of a *product_definition_formation* entity.

Assemblies are obtained by means of *product_definition:relationship* entities that are used for specifying associations between couples of components (parts). A product definition can have one or more associated documents and this is enabled by the *product_definition_with_associated_documents* subtype shown in Figure 5.

Documents are used in our mapping to transform the drawing entities in the legacy system: in fact the drawing attributes and their meanings correspond quite well with the document attributes and can be translated by means of simple string operations.

The effect of the transformation of the data instances is evident when Figure 3 and 6 are compared. As it is easy to verify, each assembly (structure) in Figure 3 corresponds to a *product_definition_relationship* entity in Figure 6, while each component (*dash_number*) is mapped onto a different product element by means of a *product_definition_formation* entity.

In the legacy model each structure is assigned its own effectivity (which is, roughly speaking, a sort of range of validity for the associated assembly).

This information is maintained in the AP203 model where the effectivity entity is linked to the product definition relationship. It should be noted, however, that in general the structure of the effectivity is more articulated and complex in the AP203 schema.

4 CONCLUSIONS

This paper has presented a case study for transforming product data instances stored inside a (proprietary) legacy system into equivalent information structured according the STEP AP203 data model. This mapping is particularly important when product data technologies based on the ISO 10303 international standard have to be introduced in an organisation and a smooth migration path must be ensured for such innovating approaches. The adoption of STEP, in fact, has a strong influence on the architecture of the system software used for storing, exchanging and sharing product data but also involves radical changes in the procedures adopted to generate, organise and manage the data themselves.

In several situations the preservation of the valuable information already present in a production environment is a mandatory issue and this can be achieved only by carefully analysing the legacy data model and by getting it to correspond with one or more application protocol data schemas.

Figure 4 EXPRESS-G specification of the AP203 subset model (part 1).

Figure 5 EXPRESS-G specification of the AP203 subset model (part 2).

Figure 6 Graphic representation of data entities in the AP203 subset.

5 ACKNOWLEDGEMENTS

The authors are grateful to B. Chiavola, L. Garda, P. Verzino from Alenia for the useful discussions and comments about the enterprise data models.

This work was partially supported by UNINFO and Camera di Commercio di Torino.

6 REFERENCES

- Bailey I. D., Mead M. (1993) EXPRESS-M – A Schema Mapping Language, Proc. 3rd EXPRESS User Group Conference, Berlin, 1993.
- ISO 10303-1 (1995) Industrial Automation Systems and Integration - Product Data Representation and Exchange – Part. 1: Overview and Fundamental Principles, International Standard 10303-1, 1995.
- ISO 10303-11 (1995) Industrial Automation Systems and Integration - Product Data Representation and Exchange – Part 11: Description Methods: The EXPRESS Language Reference Manual, International Standard 10303-11, 1995.
- ISO 10303-21 (1995) Industrial Automation Systems and Integration - Product Data Representation and Exchange – Part 21: Implementation Forms: Clear Text Encoding of the Exchange Structure, International Standard 10303-21, 1995.
- ISO 10303-41 (1995) Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part. 41: Integrated Generic Resources: Fundamentals of Product Description and Support, International Standard 10303-41, 1995.
- ISO 10303-203 (1995) Industrial Automation Systems and Integration - Product Data Representation and Exchange – Part. 203: Application Protocols: Configuration Controlled Design, International Standard 10303-203, 1995.
- ISO 10303-214 (1997) Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part. 214: Application Protocols: Core Data for Automotive Mechanical Design Processes, Draft International Standard 10303-214, 1997.
- ISO PISA (1994) Industrial Automation Systems and Integration – Product Data representation and Exchange – PISA Information Modelling Language: EXPRESS-C”, TC184/SC4/WG5/WD N202, 1994.
- Staub G., Maier M., Grabowsky H. (1997) The ECCO Toolkit User Reference Manual, Karlsruhe University, Germany, 1997.

7 BIOGRAPHIES

Silvano Rivoira graduated in Electronic Engineering at the Politecnico di Torino, he was associate professor of Computer Science at the University of Torino and full professor at the University of Perugia. Since 1991 he has been full professor of Computer Systems at the Politecnico di Torino, where he is director of the Dipartimento di Automatica e Informatica.

His research activity has concerned speech recognition, operating systems and multiprocessor architectures. Currently his main interests are in product data technology.

Claudio Demartini graduated in Electronic Engineering at the Politecnico di Torino, he is currently full professor in the same institution, where he teaches Computer Engineering and Database fundamentals. His interests include communication protocols, industrial communication networks and database principles. He is currently involved in international and national activities in the area of industrial communications and product data technology.

Adriano Valenzano graduated in Electronic Engineering at the Polytechnic of Turin. He is currently Director of Research at Centro Elaborazione Numerale dei Segnali (CENS), Polytechnic of Turin where he is responsible for researches about distributed systems, local area networks and communication protocols.

His current research interests are in the fields of computer networks and communication protocols (in particular, industrial communications), formal methods for protocol engineering and product data technologies.