

ONTOLOGY-BASED PRODUCTS INFORMATION INTEROPERABILITY IN NETWORKED MANUFACTURING ENTERPRISES

Tursi A.^{1,2}, Panetto H.¹, Morel G.¹, Dassisti M.²

¹ *Centre de Recherche en Automatique de Nancy (CRAN - UMR 7039), Nancy-University, CNRS, France
{angela.tursi, herve.panetto, gerard.morel}@cran.uhp-nancy.fr*

² *Dipartimento di Ingegneria Meccanica e Gestionale, Politecnico di Bari, Italy
m.dassisti@poliba.it*

Abstract: Standardisation initiatives (ISO and IEC) try to answer the problem of managing heterogeneous information, scattered within organizations, by formalising the knowledge related to products technical data. While the product is the centred object from which, along its life cycle, all enterprise systems, either inside a single enterprise or between cooperating networked enterprises, have a specific view, we may consider it as active as far as it participates to the decisions making by providing knowledge about itself. This paper proposes a novel approach, postulating that the product, represented by its technical data, may be considered as interoperable per se with the many applications involved in manufacturing enterprises as far as it embeds knowledge about itself, as it stores all its technical data, provided that these are embedded on a common model. The matter of this approach is to formalise of all technical data and concepts contributing to the definition of a *Product Ontology*, embedded into the product itself and making it interoperable with applications, minimising loss of semantics. *Copyright © 2007 IFAC*

Keywords: Enterprise Integration and Networking, Interoperability, Product Data Management, Ontology

1. INTRODUCTION

Nowadays, information management is considered as a main requirement for products development in a networked environment. The new communication technologies, such as wireless technologies, RFID, etc., allow, from a technological point of view, to consider products as *active mobile objects*, embedding their own information structure, used and updated by the various actors during the product lifecycle.

By this way, these products may be considered as systems, like other classical systems, that collaborate for a common objective (development, traceability, etc.). However, it is possible to demonstrate that the properties of such a system are in line with the so-called *System of Systems* concept (Maier, 1998), in which interoperability is a major concern.

Indeed, a System of Systems (SoS) is a *system arising from collaborative functioning of unit systems, each of them being able to work alone in order to perform its own operational mission (INCOSE¹)*. A SoS is such if it satisfies the following criteria (Maier, 1998):

1. Operational Independence of Elements
2. Managerial Independence of Elements
3. Evolutionary Development
4. Emergent Behaviour
5. Geographical Distribution

Based on these characteristics, studies (Morel, *et al.*, 2007) try to demonstrate that, in some extent, an enterprise may be considered as a SoS, because it is

¹ INCOSE (International Council on Systems Engineering),
<http://www.incose.org>

composed of systems (its elements), and has a particular finality, related to its skill domain, resulting from the execution of enterprise applications.

In this paper, we postulate that a product embedding its own knowledge (information) and equipped with communication facilities can be considered as a component element (an active object) of such a SoS. Indeed a product is able to “act independently” (criteria 1, 2) during its lifecycle if it embeds an information model that gives it the possibility to store data necessary to ensure its existence. The other SoS criteria, related to the enterprise, are not put at fault when one SoS element is an active object.

The main objective of our research activities is then to define the information model necessary to the product to become an active object. With such information, it may be interoperable per se with the many applications involved in manufacturing enterprises and, as far as it embeds knowledge about itself, storing all its technical data, it will be able to act as a common source of understanding between enterprises applications.

The paper puts a first step in this direction by endeavouring existing standards related to product technical data modelling for the definition of products information, allowing a non ambiguous model to represent knowledge and concepts, processable by the many enterprise applications adopted in manufacturing environment. A second step will be to formalise this information model as a *Product Ontology*, thus including domain rules, able to express and share product knowledge among systems. An ontology is *an explicit specification of a shared conceptualization* (Gruber, 1993), which allows the representation of domain’s knowledge. It allows the formalization of the semantics of objects, and then it allows to formalize and to identify the modelling concepts and their dynamic behaviour, in order to express and to share this knowledge.

The shape of the paper is as following: after introducing the concept of active mobile object in section 2, section 3 introduces standards concerned on information interoperability that are used as a base for defining the product ontology; section 4 presents a case study of instantiation of the standards models defining, by a bottom-up approach, the common concepts that will be a base for our product ontology, and finally in section 5 future research activities and conclusions are presented.

2. PRODUCT AS AN ACTIVE MOBILE OBJECT

In order to better optimise its performance, heterogeneous enterprise applications, either at business or at manufacturing levels, either inside a single enterprise or between networked enterprises, need to share information and to cooperate. These information may be stored, processed and communicated in different ways by different applications, according to the scopes for which these have been collected and they will be used. Each enterprise application, in fact, uses an information

repository, which refers to a Reference Information Model (RIM). A RIM specifies the structure and embeds the semantics of the information treated, in relation to the scope of the application to which it is devoted (Dassisti, *et al.*, 2006). However, a problem of misunderstanding when information is exchanged between enterprise applications can occur, due to different view points, for which they have been developed and, consequently, a risk of loss of information semantics may arise when exchanging between heterogeneous systems.

This “Babel tower effect”, induced by the heterogeneity of applications, of users and of domains may cause information understanding problems, leading systems to fail at collecting information from different and heterogeneous sources to effectively ensure their local objective. This problem of managing heterogeneous information coming from different systems, in order to achieve a unique comprehension, falls within the umbrella of interoperability problems. Generally speaking, interoperability can be defined as that intrinsic characteristic of a generic entity (organization, system, process, model, ...) allowing its interaction with other entities - to a different extent of simplicity - to cooperate for achieving a common goal within a definite interval of time, while pursuing its own specific goal. *Interoperability can be defined as the ability of two or more systems or components to exchange information and to use the information that has been exchanged* (IEEE, 1990).

In order to manage heterogeneous information, it is mandatory to develop models able to trace all relevant information related to the product lifecycle (design, manufacturing, sales, use and disposal). This information is, in fact, quite often scattered within organizations: it is a matter of the materials adopted, of the applications used to manage technical data (e.g.: Product Data Management systems (PDM)), of the applications that manage business data (e.g.: Enterprise Resource Planning (ERP)) and, finally, of the applications that manage manufacturing data (e.g.: Manufacturing Execution Systems (MES)).

Standardisation initiatives, for instance in the frame of ISO (ISO 10303) and IEC (IEC 62264), try to answer this problem by formalising the knowledge related to products technical data: nevertheless this approach is rather prescriptive, because it forces users to translate information from generic concepts to more practical and ad-hoc ones.

This paper postulates that, the product, along its life cycle, is the centred object from which all applications have a specific view. It could then be possible to define a common information model, to support information exchange between the product views and the many applications that interact with them.

This model intends to specify an embedded *Product Ontology* that may be formed during the product lifecycle by the force of necessity of using it to communicate with the applications.

An ontology provides formal definitions of basic concepts in a domain and the relationships among them

in a usually logic-based language (Gruninger and Lee, 2002)

There have been, in many different sectors, some efforts examining the use of ontologies in supporting the semantic integration task (e.g. Gehre et al., 2005; Guo et al., 2003; Katranuschkov et al. 2003; Lima et al. 2005, Patil et al., 2005). Aware of the efforts demonstrating the integration of models using ontologies, the matter of the approach, discussed here, is how to formalise such *Product Ontology*, so it is feasible to embed information into the product and bringing and using them without further misunderstanding. In this sense, standards efforts can be taken into account, in term of useful bases for the ontology of the domain. The product may then be considered as active, as far as it participates to the decisions making, providing knowledge about itself. Through the formalization of this model, the product may be considered as interoperable per se as far as it embeds local information (knowledge about itself), as it stores all its technical data, provided that these are embedded on a common model, providing mappings from and to the enterprise applications (either inside a single enterprise or between networked enterprises) with respect to its life cycle.

This paper discusses these ideas and traces a research roadmap on this topic, to explore the possibility to make enterprise interoperable on the basis of product-centred information view.

3. STANDARDS FOR PRODUCT DATA REPRESENTATION AND EXCHANGE

The integrated management of all the information regarding the product and its manufacturing is one of the more complex questions that characterize today's environment, defining a sort of "product centric" or "product-driven" paradigm (Morel, *et al.*, 2003). Information interoperability asks for common shared approaches: in fact, interesting standardisation initiatives already exist, such as the IEC 62264 set of standards (IEC 62264, 2002) and the ISO 10303 technical specifications (ISO/TS 10303, 2004). They try to solve the problem of managing heterogeneous information coming from different systems by formalising the knowledge related to products technical data. Both these standards will be here addressed because they are related to Product Data Management at the business and the manufacturing levels of enterprises (B2M). In the following sub-paragraph a short review will be done of these two standards.

3.1. IEC 62264 set of standards.

The IEC 62264 set of standards specify a set of reference models for information exchange to facilitate the integration of business applications and manufacturing control applications, within an enterprise. The full standard is composed by six different parts designed for defining the interfaces

between enterprise activities and control activities. Among all its parts, the part 1 specifies the relevant functions within an enterprise and within the control domain of an enterprise, stating which objects are normally exchanged between these domains. The key aspects for integrating the business applications and the manufacturing operations and control applications are the information structures and exchanges, related to the products, managed by activities, applications, processes, resources, and functions.

The boundary between the enterprise manufacturing operations and control domains are signed by models: *hierarchy model* that describes the levels of functions and domains of control associated within manufacturing organizations; *data flow model* that describes the functional and data flows within manufacturing organizations; *object model* that describes the information that may cross the enterprise and control system boundary.

There is a set of eight object models that specifies all concepts for enterprise-control integration: three are related to the resource hierarchy (Personnel, Equipment, Material), the process hierarchy (Process Segments, Product Definition), and to the production (Production Schedule, Production Performance, Capability Definition).

B2MML² (Business to Manufacturing Markup Language) is an XML implementation of the IEC 62264 part 1. It consists of a set of XML schemas, developed by the World Batch Forum, written using the World Wide Web Consortium's XML Schema language (XSD) that implements the standardised data models. B2MML is meant to be a common data format to link business enterprise applications (such as ERP systems) with manufacturing enterprise applications (such as MES). In particular, MES functions relate to production monitoring including materials (raw and finished) and resources (equipment and personnel) traceability information.

This standard shows a first effort in determining a model that intends facilitate enterprise applications interoperability.

In order to adopt a common understanding for the manipulation of models, we choose the UML (Unified Modeling Language), and more especially the class diagram formalism, as a common conceptual language for representing the concepts specified in the standard. Fig. 3 shows the UML formalisation of the conceptualised Material model, deduced from IEC 62264 and its B2MML implementation. This conceptualisation has been deduced by taking in to account, not only the B2MML relational schemas but also the rules applied when instantiating such model to specific products. This conceptualisation has been done for all models as defined in the part 1 and 2 of the standard.

² B2MML (Business to Manufacturing Markup Language), <http://www.wbf.org>

3.2. ISO 10303 Technical specifications

ISO has been pushing forward the development of standards and models to foster the exchange of information related to goods and services (ISO, 2005). Efforts like ISO 10303 STEP – STandard for the Exchange of Product model data – have tried to deal with the issues of integration and interoperability problem. STEP represents the standard for the computer-interpretable representation of product information and for the exchange of product data. It aims to provide a neutral mechanism capable of describing products throughout their lifecycle. Nowadays, STEP has been recognized as appropriate to help in the integration of manufacturing systems in industries such as automotive, aircraft, shipbuilding, furniture, building and construction, gas and oil.

A significant solution for PDM (Product Data Management) data exchange is the Unified PDM Schema, which is a basic specification for the exchange of administrative product definition data. It has been created by unifying all PDM data between all existing STEP Application Protocols, and allows the exchange of information that is stored in PDM systems. This information typically forms the metadata for any product. In order to deal with the increasing demands on product models exchange, the standard has specified a set of STEP reusable modules related to PDM. These modules are now published as technical specifications (TS) and concern all related information attached or describing products technical data such as product structure, configuration control, persons and organisations, etc. PDM systems maintain a single copy of the product master data in a secure vault; the data are then distributed to those departments requiring them: modified, updated design data are then resaved in the vault. Data integration ensures that the information describing product design, manufacturing and life cycle support is defined only once; STEP data integration eliminates redundancy and the problems caused by redundant information.

STEP uses the EXPRESS language for describing data type, constraints on data type and relationship between data type. However, Application Protocols are required to contain a representation of the information in both EXPRESS and EXPRESS-G. EXPRESS-G is a diagramming technique supporting a subset of EXPRESS language.

In this case, we also use the UML Class Diagram formalism as a common conceptual language for specifying the concepts based on the ARM (Application Reference Model).

4. A CASE STUDY

The STEP PDM Schema is a reference information model for the exchange of a central, common subset of the data being managed within a PDM system. A PDM system is able to integrate and manage all applications, information and processes that define a product during its lifecycle, from design to manufacture, and to end-

user support (Liu and Xu, 2001). This means that typical product-related information includes geometry, engineering drawings, project plans, part files, assembly diagrams, product specifications, numerical control machine-tool programs, analysis results, correspondence, bills of material, engineering change orders, and many more.

However, it is possible to find some of these information scattered also in the IEC 62264 models. Because the space is limited, we will now demonstrate this only by defining some mappings between the PDM schemas and the IEC 62264 material model. Of course, the current work is ongoing taking into account all models.

The IEC 62264 *Material Model* defines the actual materials, material definitions, and information about classes of material. In this model, material means not only raw materials but also consumables, parts and all other products needed for the production. *Material information* includes the inventory of raw, finished, and intermediate materials. *Material classes* are defined to organise materials. A *Material definition* is a means to describe goods with similar characteristics. A *Material Lot* object identifies a specific amount of material, countable or weighable, and it has specific properties (Fig. 3).

In order to verify that the same information is modelled in different way by the two standards, we have denormalised and conceptualised PDM STEP Schema and IEC 62264 models and represented them using the UML class diagram notation. In this way, it is possible to have a common minimum denominator which allows the matching and the mapping between the two standards.

The following step has been the instantiation of the standards models based on a real production system. The proposed case study concerns the production of parts and the assembly of these into different products. This industrial production system is provided by a local technical centre : the AIPL-PRIMECA³ (*Atelier Inter-établissement de Productique Lorrain*).

In this environment, 4 types of base part from a products family are manufactured and then assembled in order to compose 6 types of product (Fig. 1).

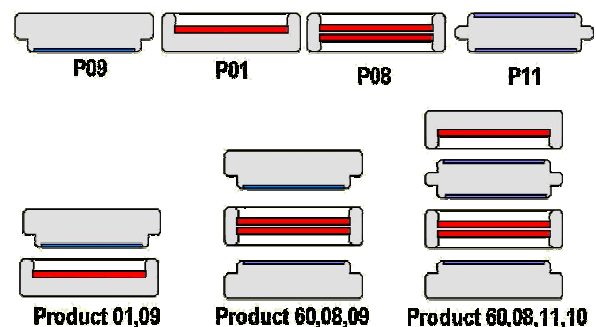


Fig. 1 Parts and some products produced at the AIPL.

In this paper we focus mainly on the proof of concepts regarding the definition of mappings between both

³ www.aip-primeca.net/lorraine/

standards by instantiating the IEC 62264 Material Model and the corresponding STEP PDM modules models on a few parts. (Fig. 4, 5).

The figures show how information related to product properties, such as diameter for example, is defined in both models, even if they are represented in a different way. Indeed, in STEP PDM modules (Fig. 4), the *P09* is a product, which is member of *Pxx* family. It is possible to assign diameter property to *P09* through the class-relationship *assigned_diameter*, which is linked to the numerical value through the representation concept. However, we observe that it is not possible to separate the definition of a concept for an entity from the value that it assumes. In the same way, in the IEC 62264 (Fig. 5), we note that the diameter concept is defined for *Pxx* class: *P09* part belongs to *Pxx* class, consequently it has a diameter, but it may have also other specific characteristics. Finally, the value of *P09* diameter is specified as property of a precise *P09* lot.

In order to demonstrate that the models describe the same information, our approach is based, firstly, on a syntactical analysis whose aim is to compare the instances defined in both models and then based on

semantics analysis, studying properties of the shared objects. The result of the syntactical analysis is presented in the Table 1. Finally, the semantics analysis suggests the possibility to do a mapping between the instantiated concepts. However each relation between different concepts can be studied and it is possible to try semantic correspondences between them (Baïna, 2006) in order to compare the contained information. Different cases can occur (Fig. 2): equivalence, represented by $\equiv$ symbol (same definition for concepts semantics in the two standards), inclusion, represented by $\subset$ symbol (a semantic concept includes the other one), and intersection, represented by $\cap$ symbol, (the concepts intersection defines the common sense of the two concepts). This will be the next step in our approach towards a common model, embedded into the products, which store all technical data along its life cycle: this represents the starting point for the development of our Product Ontology.

Table 1 Syntactical Analysis of concepts

AIPL objects	STEP PDM modules concepts	IEC 62264 concepts
Pxx	Pxx: Product	Pxx: MaterialClassType
P09	P09: Product	P09: MaterialDefinitionType P09_Lot: MaterialLotType
Diameter	Diameter: Independent_property	Diameter: MaterialClassPropertyType
Diameter_value_with_unit	Diameter_value_with_unit: Numerical_item_with_unit	Diameter_value_with_unit: MaterialLotPropertyType

This case study then shows how ISO 10303 and IEC 62264 initiatives formalise the knowledge related to products technical data and because of this they are very useful to define the information model that can allow to consider the product as an active object: other standardisation initiatives can be considered in order to have a complete model which stores all technical data along product lifecycle.

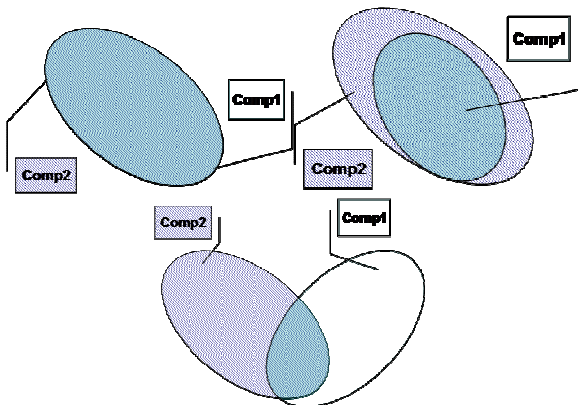


Fig. 2 Correspondences between the concepts semantics: (a) equivalence, (b) inclusion, (c) intersection

For analysing the semantics relationships between both concepts, we choose the First Order Logic (FOL) to express predicates formalising concepts of models in order to define, precisely, the mapping between them. From the syntactical analysis (Table 1), we deduce one or more FOL predicates (1), (2), (3), (4). Each of them formalises mapping between STEP PDM concepts and IEC 62264 ones represented by a DLite *description logic* (Calvanese, et al, 2002).

$$\begin{aligned} & \text{Product(Pxx)} \wedge \text{Product(P09)} \wedge \\ & \text{Product_relationship(member_of)} \wedge \\ & \text{member_of(Pxx, P09)} \wedge \\ & \text{member_of.relation_type="membership"} \end{aligned} \quad (1)$$

$$\begin{aligned} & \Rightarrow \text{MaterialClassType(Pxx)} \wedge \\ & \text{MaterialDefinitionType(P09)} \wedge \\ & \text{defines_a_grouping(Pxx, P09)} \end{aligned}$$

Product $\supset$ MaterialClassType Product $\equiv$ MaterialDefinitionType
--

Remark: MaterialLotType is a concept, present in IEC 62264 Material Model but without semantics equivalence in PDM STEP modules

Independent_property $\equiv$ MaterialClassPropertyType Numerical_item_with_unit $\equiv$ MaterialLotPropertyType
--

Remarks: Representation concept (Property_representation, Representation, Representation_context) is present in STEP PDM modules and not in IEC 62264.

$$\begin{aligned}
& \text{Assigned_property}(\text{Assigned_diameter}) \wedge \\
& \text{Representation}(\text{Representation_diameter}) \wedge \\
& \text{Property_representation}(\text{Diameter_property_representa} \\
& \quad \text{tion}) \wedge \\
& \text{Diameter_property_representation}(\text{Representation_dia} \\
& \quad \text{meter, Assigned_diameter}) \wedge \\
& \text{Numerical_item_with_unit}(\text{Diameter_value_with_unit}) \quad (3) \\
& \quad \wedge R(\text{Diameter_value_with_unit, Representation}) \\
& \Rightarrow \text{MaterialClassPropertyType}(\text{Diameter}) \wedge \\
& \text{MaterialLotPropertyType}(\text{Diameter_value_with_unit}) \\
& \quad \wedge \text{value_of}(\text{Diameter_value_with_unit, Diameter})
\end{aligned}$$

Assigned_property $\subset$ MaterialClassPropertyType

$$\begin{aligned}
& \text{Numerical_item_with_unit}(\text{Diameter_value_with_unit}) \\
& \quad \wedge \text{length_measure}(\text{Diameter_measure}) \wedge \\
& \quad \text{Unit}(\text{Millimeter}) \wedge F(\text{Diameter_value_with_unit,} \\
& \quad \quad \text{Length_value}) \wedge \\
& \quad G(\text{Diameter_value_with_unit, Millimeter}) \wedge \\
& \quad \quad \text{Length_value.length_measure} = "50" \wedge \\
& \quad \quad \text{Millimeter.name} = "millimeter" \quad (4) \\
& \Rightarrow \text{ValueType}(\text{Diameter_Value}) \wedge \\
& \text{MaterialLotPropertyType}(\text{Diameter_value_with_unit}) \\
& \quad \wedge \text{Value}(\text{Diameter_with_unit, Diameter_Value}) \wedge \\
& \quad \text{Diameter_Value.UnitOfMeasure} = "Millimeter" \wedge \\
& \quad \text{Diameter_Value.ValueString} = "50"
\end{aligned}$$

Unit $\subset$ ValueType Length_measure $\subset$ ValueType
--

Finally, we have started the formal verification of these mapping rules through *Protégé*⁴, a suite of tools to construct domain ontology, by applying skill-based axioms through an inference engine and using it with knowledge-based applications.

5. CONCLUSIONS AND FUTURE RESEARCH

Starting from the consideration that the product is the common added value, for which each part of the organization works, and it is the common element perceived in the same way by all manufacturing operators, we have postulate that it is possible to consider the product as an interoperable system per se, as far as it embeds knowledge about itself as it stores all its technical data. This information can be structured in a common formal model, including domain rules, which is able to provide mappings from and to the enterprise applications information, either inside a

$$\begin{aligned}
& \text{Independent_property}(\text{Diameter}) \wedge \text{Product}(\text{P09}) \wedge \\
& \text{Assigned_property}(\text{Assigned_diameter}) \wedge \\
& \text{Assigned_diameter}(\text{Diameter, P09})
\end{aligned}$$

$$\begin{aligned}
& \Rightarrow \text{MaterialClassType}(\text{Pxx}) \wedge \\
& \text{MaterialDefinitionType}(\text{P09}) \wedge \\
& \text{defines_a_grouping}(\text{Pxx, P09}) \wedge \\
& \text{MaterialClassPropertyType}(\text{Diameter}) \wedge \\
& \text{MaterialClassProperty}(\text{Pxx, Diameter}) \quad (2)
\end{aligned}$$

single enterprise or between networked enterprises, with respect to its life cycle.

The matter of the proposed bottom-up approach is then the formalisation of knowledge and skill around products and semantics of standard modelling concepts, for making interoperable enterprise applications related to products views. Our current work is, based on these concepts mapping coming from standards, the definition of a *Product Ontology* to contribute to an interoperability solution between active mobile objects and enterprise applications that will manage them.

REFERENCES

- Baïna, S. (2006). *Interoperabilite Dirigee par les Modeles : Une Approche Orientée Produit pour l'interopérabilité des systèmes d'entreprise*. Thèse de doctorat de l'Université Henri Poincaré, Nancy I, France. Décembre 2006.
- Calvanese D., De Giacomo G., and Lenzerini M. (2002). Description logics for information integration. In Computational Logic: From Logic Programming into the Future (In honour of Bob Kowalski), *Lecture Notes in Computer Science*. SpringerVerlag, 2002.
- Dassisti M., Panetto H., Tursi A. (2006). Product-driven Enterprise Interoperability for Manufacturing Systems Integration. Proceedings of the BPM2006 Business Process Management Workshops. 2nd ENEL Workshop, Vienna, Austria, September 4. Springer Verlag, Lecture Notes in Computer Science, September, LNCS 4103, 249-260.
- IEC 62264 *Enterprise-control system integration, Part 1. Models and terminology, Part 2: Model object attributes*. ISO/IEC, 2002, Geneva.
- IEEE (1990), *Standard Computer Dictionary- A Compilation of IEEE Standard Computer Glossaries*. NY. 610-1990. ISBN: 1559370793
- ISO/CD TS 10303-25 *Industrial automation systems and integration - Product data representation and exchange - Implementation methods: EXPRESS to XMI binding*, 2003, Geneva.
- ISO/TS 10303 *STEP modules related to Product Data Management. Industrial automation systems and integration — Product data representation and exchange*, 2004, Geneva.

⁴ protege.stanford.edu

- Gehre A., Katranuschkov P., Stankovski V., Scherer R.J. (2005), Towards semantic interoperability in virtual organisations, In: *Proc. 22nd Conference of Information Technology in Construction, cibW78, Dresden, Germany* (July 18–21 2005).
- Gruber T.R. (1993). Toward principles for the design of ontologies used for knowledge sharing. In *Formal Ontology in Conceptual Analysis and Knowledge Representation* (N. Guarino and R. Poli, eds.), Kluwer Academic Publishers.
- Gruninger M., Lee J. (2002). Ontology — applications and design, In: *Communications of the ACM* **45/2**, 39–41
- Guo M., Li S., Dong J., Fu X., Hu Y., Yin Q. (2003). Ontology-based product data integration, In: *Proc. 17th International Conference on Advanced Information Networking and Applications*; ISBN 0-7695-1906-7, 530–533.
- Katranuschkov P., Gehre A., Scherer R. (2003), An ontology framework to access IFC model data, In: *ITcon*; **8**, 413–437
- Lima C., Silva C.F., Sousa P., Pimentao J.P. (2005), Interoperability among semantic resources in construction: is it feasible?, In: *Proc. 22nd Conference of Information Technology in Construction, cibW78, Dresden, Germany* (July 18–21 2005).
- Liu D.T., Xu X.W. (2001). A review of web-based product data management systems. In: *Computer In Industry*; **44**, 251–262
- Maier M.W. (1998). Architecting Principles for Systems-of-Systems, In: *Systems Engineering*, **1/4**, 267–284.
- Morel G., Panetto H., Zaremba M.B., Mayer F. (2003). Manufacturing Enterprise Control and Management System Engineering: paradigms and open issues. In: *IFAC Annual Reviews in Control*. **27/2**, 199–209, Elsevier
- Morel G., Auzelle J.P., Mayer F., Panetto H. (2007). System of enterprise-Systems integration issues: an engineering perspective. *Invited survey paper. in this issue*.
- Patil L., Dutta D., Sriram R. (2005), Ontology-based exchange of product data semantics, In: *IEEE Transactions of Automation Science and Engineering*; **2**, 213–225.
- Terzi S. (2005). *Elements of Product Lifecycle Management: Definitions, Open Issues and Reference Models*, PhD Dissertation, UHP Nancy I, Politecnico di Milano.
- Terzi S., Panetto H., Morel G., Garetti M. (2007). A holonic metamodel for product lifecycle management. *International Journal of Product Lifecycle Management*, **2/2**, ISBN 1743-5110

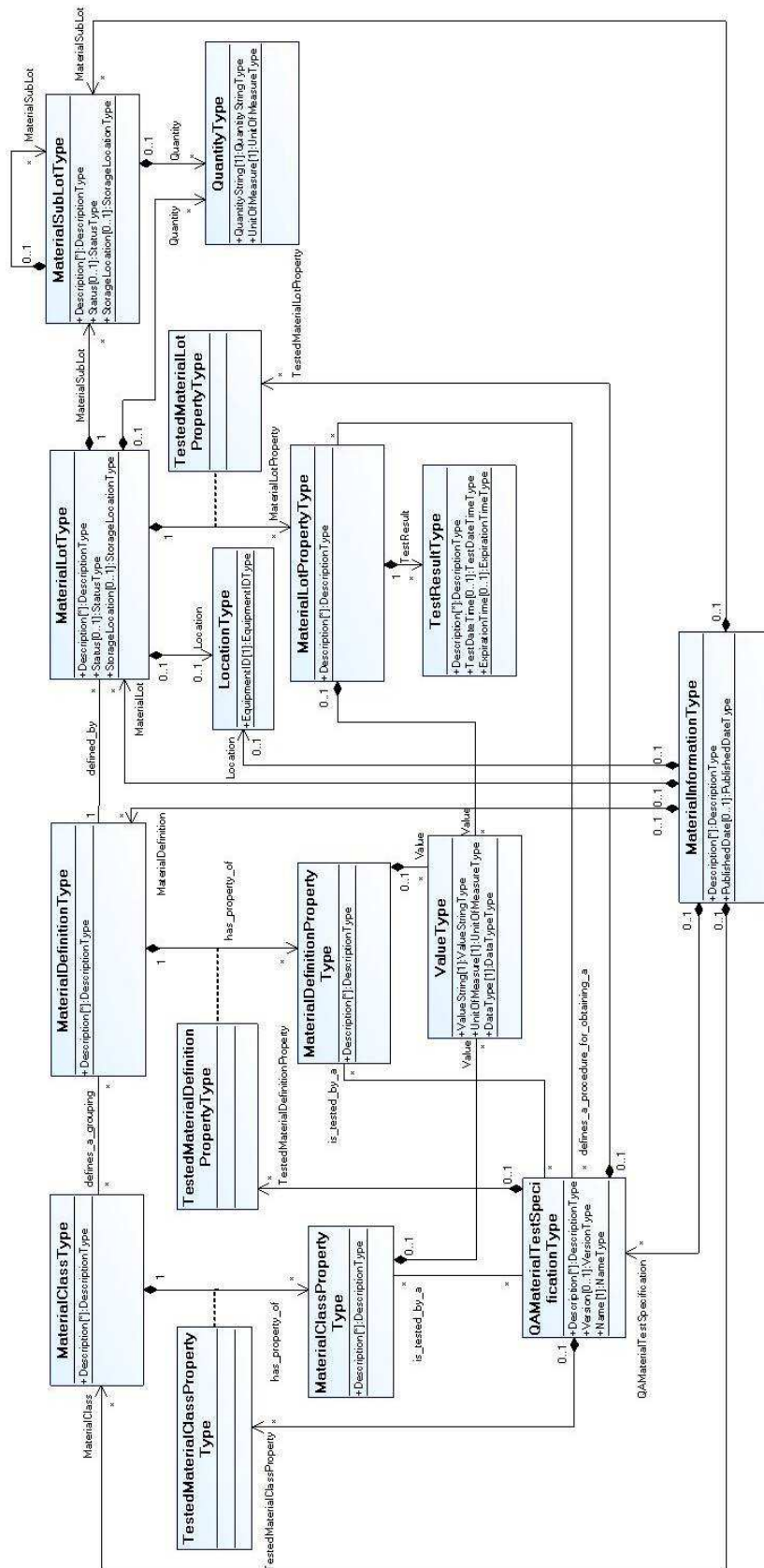


Fig. 3 The conceptualised IEC 62264 Material Model.

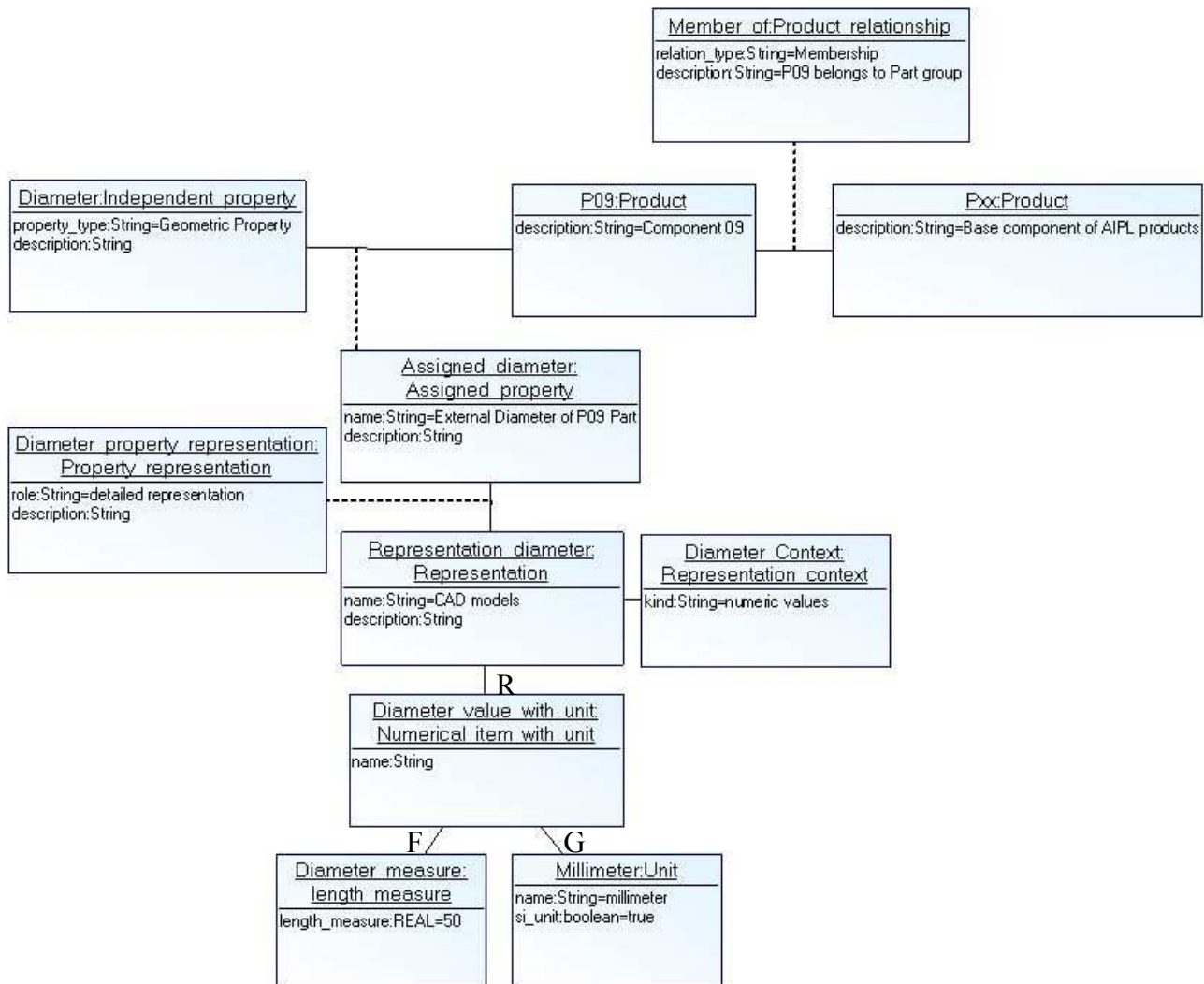


Fig. 4 An extract of the instantiated STEP PDM modules.

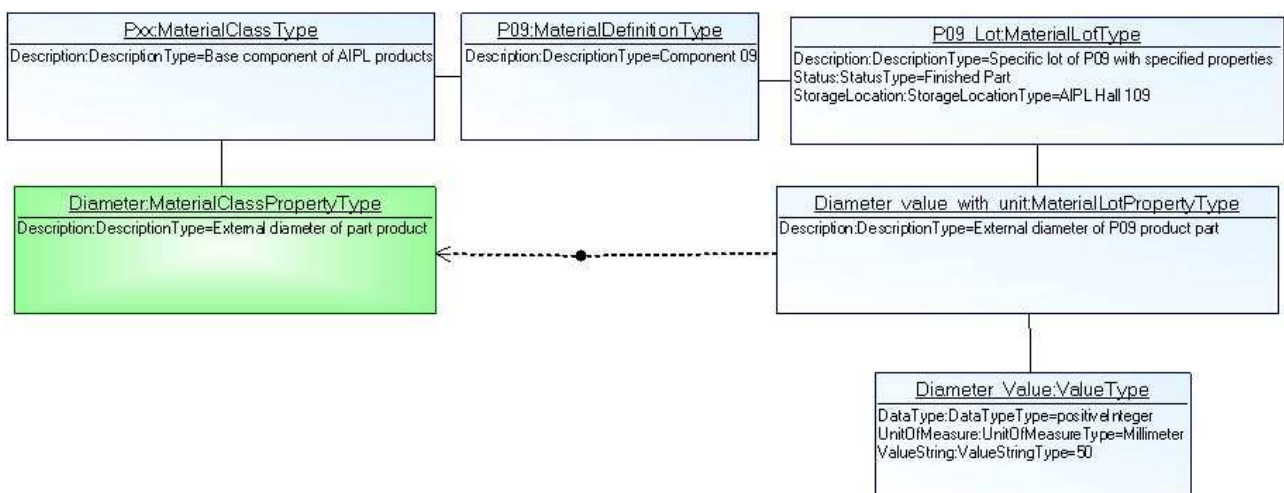


Fig. 5 An extract of the instantiated IEC 62264 Material Model.