Adapting HPM to B2M interoperability issues: Towards Interoperability between Business Management Level and Shop Floor Level

Salah Baïna [1], Hervé Panetto [1], Khalid Benali [2], Gérard Morel [1]

[1] *CRAN (UMR 7039), University Henry Poincaré Nancy I*
F 54506 Vandoeuvre les Nancy, France,
{salah.baina, herve.panetto, gerard.morel} @cran.uhp-nancy.fr

[2] *LORIA (UMR 7503), Campus scientifique,*
F 54506 Vandoeuvre les Nancy, France,
benali@loria.fr

Abstract. Enterprise-control system integration between business systems, workflow systems, manufacturing execution systems and shop-floor process-control systems remains a key issue for facilitating the deployment of plant-wide information-control systems. Since years, increasing flexibility and efficiency of their systems has been the new challenge for modern manufacturing enterprises. Business process modelling aims at specifying objects flows and processes inside enterprise levels and among networked enterprises. However, the increased complexity of these models does not help at ensuring coherent relationships between its components. Moreover, when modelling different flows could handle different views of the same objects, thus coherence have to be maintained during the whole manufacturing process. In this paper, we show how Holonic Process Models (HPM) could be adapted for Business to Manufacturing (B2M) interoperability by applying the concept of Holon. We define a Holon oriented approach and design principles for specifying and building HPMs, in the specific domain of manufacturing systems. This approach aims at increasing model abstraction in order to simplify its initial comprehensiveness and ensure coherence between different views of manipulated objects. For reusability's sake, we propose a possible interconnection of the proposed Holon model in existing enterprise models. This integration enables exchanging data between the holon based models and other existing models.

Keywords: manufacturing systems; enterprise modelling; HMS; B2M; models integration; enterprise integration; interoperability; model mapping.

1 Introduction

In the last few years, lot of work has been done in order to ensure interoperability between applications. Enterprise application integration and the opening of information systems towards integrated access have been the main motivation for the interest around systems interoperability. Enterprise Integration which consists in connecting and making interoperable all the functional areas of an organization to improve synergy within the enterprise to achieve its mission and vision in an effective and efficient manner [1]. Enterprise integration occurs when enterprise data is handled and performed in different sites, services or companies. Manipulating all this data as a whole cannot be achieved simply by connecting computers ([2], [3]).Integration aspect and information sharing in the enterprise lead to an organisation of the hierarchy of enterprises applications where interoperability is a key issue (Fig. 1). This hierarchy defines the three main levels in manufacturing enterprise:

- Process control level contains all processes that perform moves and physical transformations on the produced goods and services;

- The Execution level perform the processes that manage decision flows (Workflow) and production flows (MES);
- The management system level is responsible of the management of processes that handle all different informational aspects related to the enterprise (e.g: ERP systems).

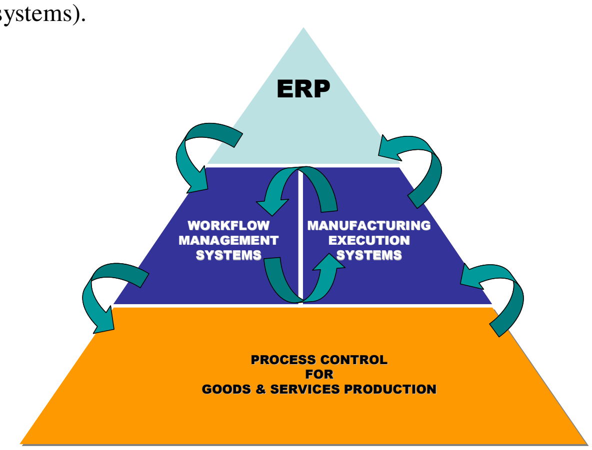

Fig. 1 Manufacturing enterprises structure

The integration of those different levels can be seen from two different points of view (Fig. 2):

- "Horizontal Interoperability" is the interoperability between applications from the same level in the enterprise. This first category of interoperability aims to synchronise models that were created in different enterprises even those managed by different modelling systems (e.g.: enabling organisational interoperability between two systems used in two different organisations).
- "Vertical interoperability" is the interoperability between applications from different enterprise levels. The objective of this category of interoperability is to maintain coherence between information that is handled in two different level of the enterprise (e.g.: ensuring coherence between organisational models of the enterprise and the process models used at shop floor level).

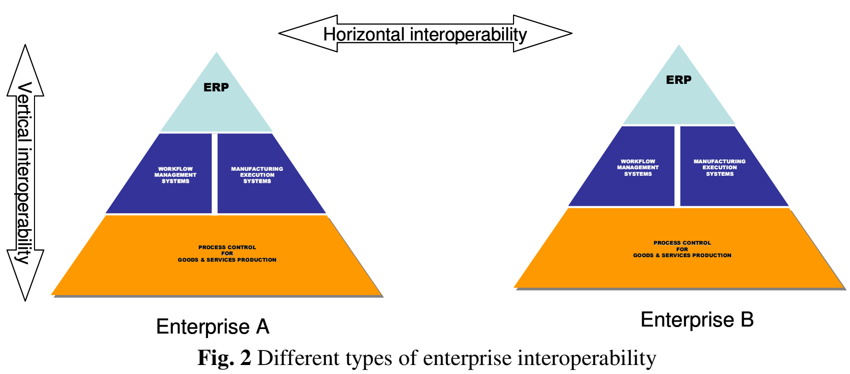

Fig. 2 Different types of enterprise interoperability

In this paper, we define a Holon oriented approach for specifying and building Holonic Process Models, in the specific domain of manufacturing systems. This approach aims at increasing manufacturing model organisation in order to simplify its comprehensiveness. We propose a modelling approach in order to handle enterprise levels interoperability concerns.

2 Applications Interoperability in Manufacturing Systems

The ISO/IEC 2382[1] Information Technology Vocabulary defines interoperability as "the capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units." The IEEE STD 610.12[2] standard defines interoperability as "the ability of two or more systems or components to exchange and use information". In this paper, interoperability definition is adapted from the two previous definitions as the ability to communicate, to cooperate and to exchange models between two or more applications despite differences in the implementation languages, the execution environments, or the models abstraction ([4], [5]). Interoperability has been studied in different domain applications. The following presents an overview of existing methods and technologies enabling interoperability.

2.1 Interoperability between ERP and MES Systems:

In this paragraph, we focus on interoperability between the business level and the shop floor level. The main purpose of interoperability between Enterprise Resource Planning (ERP) systems and Manufacturing Execution System (MES) is to improve the synchronisation between shop floor data and business information. ERP - MES interfacing speeds up the information flow between planning and process control systems. This allows tighter control of process timing and improvement of the quality of the production process. A priori detection and resolution of process disturbances into the schedule can reduce late deliveries. Better visibility of progress information leads to improved sales order control and follow up for the Customer Service department. Defining this interface requires an information analysis to sort out reliable and accurate process data for the ERP system and to define proper recipe information for the MES. Fig. 3 shows some examples of information that could be exchanged between the ERP level and the MES level.

[1] ISO/IEC 2382 (1993). Information technology -- Vocabulary -- Part 1: Fundamental terms

[2] IEEE STD 610.12 (1990). Standard Glossary of Software Engineering Terminology, IEEE, May, ISBN: 155937067X

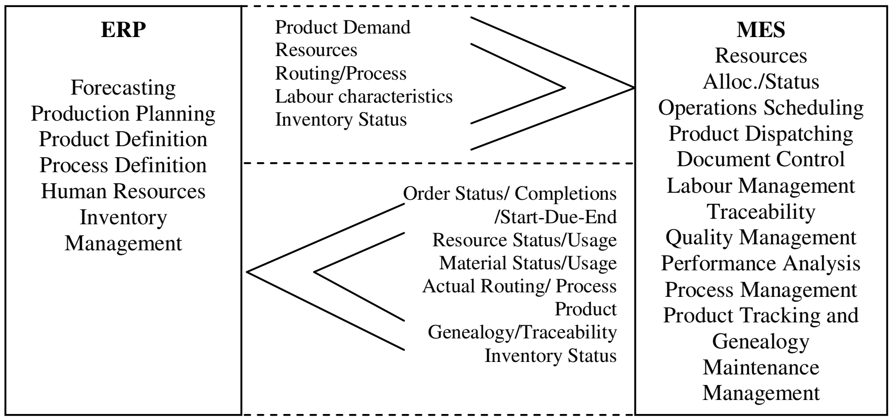

Fig. 3 Example of data that could be exchanged between an ERP and an MES

LinkForSap [6] is an example of interoperability solutions between ERP and MES systems. LinkForSap is a software package that enables information unification between the SAP R/3 ERP system and the Centum production control system. Even if the LinkForSap resolves the problem of interconnection in a computer integrated manufacturing system, this solution is valid for a specific ERP application, and can not be reused for another application. In the special case of the interoperability paradigm, point-to-point solutions that create single links between many applications are known to be expensive in term of maintenance; indeed the interoperability solution may become more expensive to maintain than the connected applications.

2.2 Workflows Interoperability Solutions

In manufacturing systems environment; in addition to ERP and MES, other kinds of applications could be involved. In the enterprise management level, workflows management systems, business process management systems can be used (Fig. 1). In this paragraph, some examples of interoperability solutions for this kind of applications are given.

The workflow management coalition (WfMC[3] [7]) defined an interface for the communication and the interoperation between workflow engines. A key objective of the WfMC is to define standards that will allow workflow management systems, henceforth called workflow product from different vendors to pass work item seamlessly between one another. The interface defined in [8] consists of the specification of all API calls needed to cover all cases and possibilities when two workflows can co-operate (e.g. exchanging status, synchronising execution, sending and answering queries). Wf-XML[9] is an interoperability standard defined by the WfMC working group, that combines the abstract commands defined by the interoperability interface defined by the WfMC, this standard consists in remote control workflow instances exchanging XML messages through http protocol. The

[3]www.wfmc.org

Wf-XML defines a set of request/response messages that are exchanged between an observer (which may or may not be a workflow management system) and a workflow management system (WfMS) that control the execution of a remote workflow instance. Several WfMSs incorporate the Wf-XML formalism to ensure interoperability of their engines (e.g. AFRICA [10]).

Shepherdson et Al. [11] proposes a cross organisational co-ordination of workflow and business process using software agents. In this proposal, a software agents is associated to each activity in a workflow or enterprise activity in a business process; this software agents is responsible of all communication that connect this activity to the others. A workflow provided by the software agent layer is called an agent enhanced workflow. This approach provides a mechanism to handle interoperability between workflows based on the interoperability between agents.

WfMC interoperability standards and most of existing techniques that enable business process or workflow interoperability are based on a message exchange paradigm (e.g. Wf-XML, BizTalk, FIPA ACL etc.). These solutions resolve only the particular case of syntactic interoperability (messages vocabulary, messages format, data types, etc.). To handle semantic concerns in workflow interconnection, Casati et al. ([12], [13]) propose a semantic specification of workflow interoperability.

Table 1 presents the classification of the studied solutions in terms of vertical or horizontal interoperability and also syntactic or semantic classification.

Table 1: recapitulation of the studied solutions for interoperability

Solution	**Domain**	**Generic solution**	**Interoperability V/ H**
LinkForSap[6]	ERP/MES interconnection	No	Vertical
Interoperability abstract specification [8]	Workflow interoperability	Yes (WfMC)	Horizontal
Wf-XML[9]	Workflow interconnection	Yes (WfMC)	-
Software Agents for Interoperability[11]	Workflow interoperability	Yes	Horizontal

In order to aid at taking into account the interoperability requirements during systems modelling phases, we introduce, in the next section, the concept of holon, in the context of manufacturing systems. This concept plays a central role in the specification of Holonic Process Models (HPM) as a specialisation of Business Process Models. We will then show how these models can be mapped into the Unified Enterprise Modelling Language (UEML [14], [15]) and the Business to Manufacturing Markup Language (B2MML) [16], which are modelling languages used to facilitate the interoperability of process models in the context of a Business to Manufacturing (B2M) approach.

3. The Holonic Approach and Manufacturing Applications Interoperability

In the manufacturing environment, numerousness of types of flows (data flows, physical flows, communication flows, event flows) introduces chaos in the organisation of flow traffic inside systems. In this context, specific problems appear, synchronising flows that handle related objects is one of those problems. In this section we define the concept of holon an its adaptation in manufacturing context as a first step towards the establishment of an approach for HPM. This approach aims to propose modelling principles for resolving information flows synchronisation in manufacturing systems, taking into account interoperability concerns during the modelling phase.

3.1 The Holon Concept in Manufacturing Process Modelling

The paradigm of "product + information" has been studied and defined as Holonic worldview [17]. The term "Holon" was applied to the manufacturing world, creating the Holonic Manufacturing Systems (HMS) community ([18], [19]). For this community a Holonic Manufacturing "is an autonomous and co-operative building block of a system for transforming, transporting, storing and/or validating information and physical objects. The Holon consists of an information processing part and often a physical processing part. A Holon can be part of another holon." ([20], [21]). A holon is then comprised of (Fig. 4):

- a physical part containing all the physical resources that constitute the holon.
- an informational part composed of all data that describe the holon, its knowledge and its behaviour.

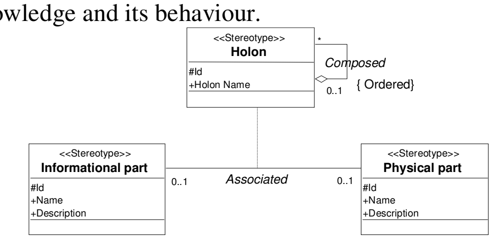

Fig. 4. The model of a Holon [22]

The previous class diagram shows a very high level abstraction of the concept of holon adapted to manufacturing systems. In a manufacturing environment, the concept of holon as defined above can be used to represent products that are equipped with information about their manufacturing, their life cycle and information relevant for its manufacturing scheduling and planning. Every processing states on the physical part of the holon is also stored in the informational part, that way the coherence between physical objects and their informational views is maintained along its life cycle. Moreover, integrating information dedicated to manufacturing decisions in the holon enables distributed decision and manufacturing systems flexibility. In the next section, we give a detailed definition for the holon concept in the manufacturing context.

The Holon concept adapted to manufacturing context. In [23], we have formalised the organisational view of a HPM putting forward holons flows and their interactions with processes. In this contribution, we focus on the structural view of holons.
We then consider that the information about a Holon is distinguished into two categories:

- Information describing the current state of the holon, this state contains three kinds of data handling attributes of space, attributes of form, and attributes of time.
- Information that concerns the holon but that does not fit into the state. This information is defined by a set of properties [19]; each property is defined by its name, description (use and unit of measure), and value.

Holons can be classified into two categories *(i)* simple holons and *(ii)* complex holons;

- Simple holons (elementary holons) are the combination of a single informational part and a single physical part.
- Complex holons (composite holons) are the result of the proceeding and treatment of one or more other holons, this proceeding can be a transformation of one holon to obtain a new one, or integrating a set of holons in order to compose a new one. Each composite holon can be defined as the output of the execution of a manufacturing process on one or more less complex holons.

In the example represented in Fig. 5, the holon *H1* is directly composed of the holons *H2* and *H3*, those holons are also composite holons. In Fig. 5, edges represent processes, nodes represent holons; composite are represented by continuous rings and elementary holons are represented by non-continuous rings.

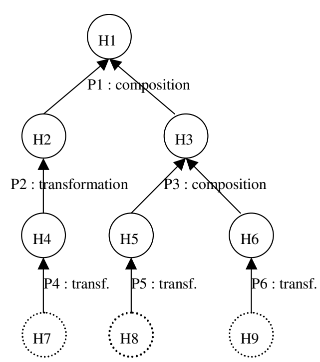

Fig. 5. A composite holon example

The specification and the definition of such holon are formalised by a class diagram showing how the concept of holon, together with its informational parts and physical parts, is linked to constructs coming from manufacturing systems domain, such as Process and Equipement (Fig. 6).

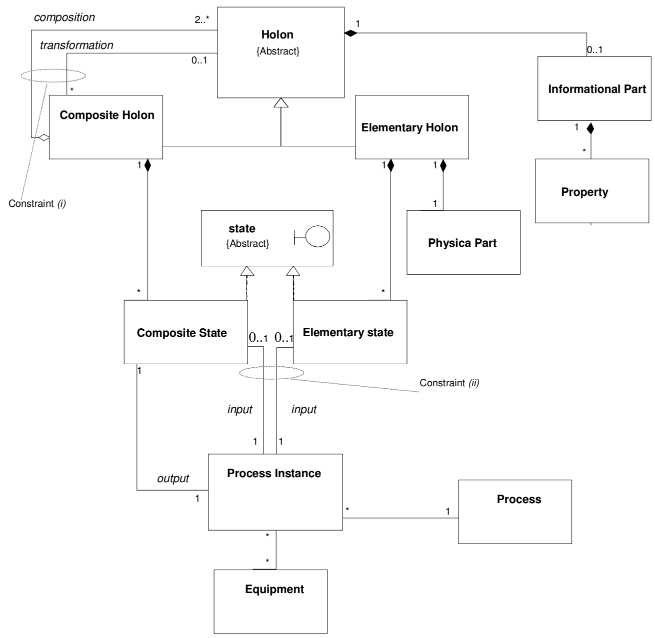

Fig. 6. Class diagram for the Holon model

Here is a brief description of the classes defined in the diagram represented in Fig. 6:
The Class *Holon* defines basic attributes for both composite and elementary holons. An *Elementary Holon* is defined as a holon with no indication about his life cycle. For example a product, produced by external manufacturing systems does not give information about the processes needed for his manufacturing. In general, an elementary state is observed and associated to each elementary holon. A *Composite Holon* is a holon that has been processed through at least a single process during his manufacturing. Only, processes inside the domain of the enterprise are taken into account. A composite Holon can be obtained by two ways:

- Composing existing holons,
- Transforming an existing holon into one or more holons.

The UML interface *state* describes the current state of a Holon (composite or elementary). Every manipulation of a holon through a process (Process Instance) implies a change in the state of the holon processed. For the sake of traceability, it is possible to create a new state instance for each change in the existence of a holon, hence the hole life cycle of the holon can be stored. An *Elementary state* describes specific data that concerns the state of an *elementary holon*. The *Composite state* class describes specific data that concerns the state of a *Composite Holon*. A *Physical Part* is a reference to the physical part encapsulated in an *Elementary holon*. The *Process instance* refers to the execution of a process on a single holon, this class enables the description with high level detail of every processing of the holon(e.g.: elapsed time, start and end of the treatment, used equipment, needed personal). A *Process* describes an internal process, a process that is performed inside the studied domain. The *Equipment* class describes the equipment needed for performing a process instance. A *Property* contains all information concerning a *holon*, that can not be handled only using the *holon state.*

In the diagram represented Fig. 6, two constraints are expressed:

constraint (i) a composite holon cannot be at the same time a composition of other holons and the result of a transformation;

constraint (ii) a process instance input is a holon state, this state can be composite pr elementary but not both, hence a process instance cannot be associated to both a composite state and an elementary one.

Other constraints on class attributes can be expressed (example: *Precedence constraint;* the time of creation a composite holon cannot be anterior to the time of creation of the holons composing this holon.), but for the sake of simplicity those constraints where omitted from the diagram.

The Holon integration in a manufacturing process modelling tool. To apply the Holonic model defined above in real use cases, an implementation has been proposed. This implementation consists on integrating this model in an enterprise process modelling tool named MEGA Suite[4], which is a business process analysis, modelling and development environment.

MEGA Suite is based on its own meta-model which can be specialised for specific needs. We used that facility to embed our own Holon model into that environment in order to test the usability of our proposal on specific application test cases. This implementation enables using Holon flows exchange when modelling processes, specifying then a Holonic Process Model.

In the next section, we show how the holon concept and the model defined can match with other reference models and meta-models of languages used in the area of enterprise integration and enterprise modelling. This matching between the different concepts enables mappings at model level, thus models using holons could be translated and integrated into models based on other standards.

3.2 Integration and Interoperability with other Models

The main objective of this section is to show that models based on the holon concept defined in section 3.1 could be expressed and transformed into models based on existing data exchange standards and other unified languages. This transformation will enable model and data exchange between applications that are based on holon models and different enterprise applications from different levels in the enterprise. To experiment the holon integration and interoperability with other enterprise modelling frameworks, two standards have been chosen:

---

[4] MEGA Suite, MEGA International, www.mega.com

- The UEML a Unified Enterprise Modelling Language used at the organisational level of the enterprise;
- The B2MML language is an implementation of the IEC/ISO 62264 standard [24] used for interfacing the manufacturing control and execution systems with higher level systems.

Mapping Holon with the Unified Enterprise Modelling Language. The Unified Enterprise Modelling Language (UEML) is the result of the UEML project [25] which aimed to solve the problem of numerousness of enterprise modelling (EM) languages [26]. The project defined a Unified Enterprise Modelling Language as an Interlingua between EM tools. The meta-model of UEML1.0 ([14], [15]) defines the set of most relevant concepts and notions for Enterprise modelling. The UEML meta-model defines a class *Object*, with two sub-types:
- *Information Object*: describes object that contain only data. Information objects can be organised in "control flows" (constraint flows or trigger flows);
- *Resource*: can be divided into two sub-types *Material resource* and *Human resource*. Resources are organised in *Resource flows*.

A trivial mapping consists in representing *Informational part* (resp. *Physical part*) using the concept *Information Object* (resp. *Material resource*), to represent holons we need a specific sub-type of the class *Object*, this specific concept must be the link between the *Material resource* (*Physical part*) and the *Information Object* (*Informational Part*).

In the UEML meta-model, an activity represents a generic description of a part of enterprise behaviour that produces outputs from a set of inputs. It represents a set of similar activities executions. The concept of "Activity" defined by the UEML constructs can be used to express the notion of "Process" which is defined in the holon model. Table 2 summarise the mapping between concepts defined in the holon model into UEML constructs.

Table 2: Correspondence between Holons related concepts and UEML concepts.

Holonic Process Model Meta-model	**UEML Meta-Model**
Holon	Object
Informational Part	Information Object
Physical Object	Material resource
Process	Activity

Mapping with the B2MML language and the IEC/ISO62264 standard: Business To Manufacturing Markup Language (B2MML) is an XML implementation of the IEC/ISO62264, it consists in a set of XML schemas written using the World Wide Web Consortium's XML Schema language (XSD) that implement the data models defined by the IEC/ISO62264 standard. The IEC/ISO 62264 is the standard specifying the exchange of data and models interfacing the shop floor level into the enterprise. It is composed by six different parts designed for defining the models and interfaces between enterprise activities and control activities. Each model concerns a particular view of the integration problem. Those models show increasing detail level in the manufacturing system. They can be classified in to two categories; operational models or resource models. To map the holon concept with the IEC/ISO 62264 models, two views are possible:

- Genealogical view: this first view concerns the genealogy of the holon, which describes relationships between holons (composition and transformation). Information handled in this view corresponds to the "Product Definition Information", as defined in the standard;
- Process view: allows representation of the lifecycle of the holon and the different processes involved during each step in the manufacturing cycle. This view fits into the "*Product production rules*" concept of the IEC/ISO 62264 standard, especially the "*Product segment*" concept, which can manage information about assembly steps and assembly actions for discrete manufacturing.

Both the genealogical and the process view are relevant for managing holons traceability along its lifecycle. In the IEC/ISO 62264, the material model is a resource model that defines the actual materials, material definitions and information about classes of material. Material information includes the inventory of raw, finished, intermediate material and material decomposition into lots and sublots. The IEC/ISO 62264 material model is the most adequate to express and handle the information about genealogy of holons (products), thus it should be used to represent the genealogical view of holon. In the IEC/ISO 62264, the notion of product segment defines the values needed to quantify a segment for a specific product, such as the number with specific qualifications. In the holon model, the class "*Process instance*" groups all information that concern the manufacturing of a specific holon. Since we consider that a holon represents a product, we can then assume that the "*Process instance*" of a specific holon describes the product segment of that holon (product). This means that all concepts associated to the *Process instance* class, can match the concepts associated to Product segment.

Table 3 shows the mapping between the genealogical aspect and the process view of the Holon model into the material model of IEC/ISO 62264 standard.

Table 3: Correspondence between Holons related concepts and IEC/ISO 62264 concepts

	Holonic Process Model	**IEC/ISO 62264**
Material Model mapping	Holon	Material sublot
	Holon Flow	Material lot
	Informational Part	Material definition
	Properties	Material lot property and Material lot property definition
Product Definition Model mapping	Process Instance	Product segment
	Equipment	Equipment specification

Mappings in action for interoperability. The previous paragraph defines mappings between the holonic model with UEML and IEC/ISO62264 models. In order to validate these concepts, we are currently developing an automatic translator based on proposed mappings from our holonic model into the other models. To describe objects and models manipulated by each meta-model the XMI [27] (XML Metadata Interchange) language has been chosen. XMI is a widely used interchange format for sharing objects using XML [28]. The XMI formalism proposes a solution enabling meta-models description in XML syntax. Moreover XMI enables also description of objects and their instances based on those models.

The XMI interface enables generating different XML files with a specific structure for each one using adequate translators, these structures are defined by the adequate standards, as shown in Fig. 7. The B2MML and the UEML XML schemas are used to generate valid and reusable documents from holonic process diagrams created in the managed in the MEGA Case tool, those documents can the be imported in applications compatible with UEML or B2MML standards.

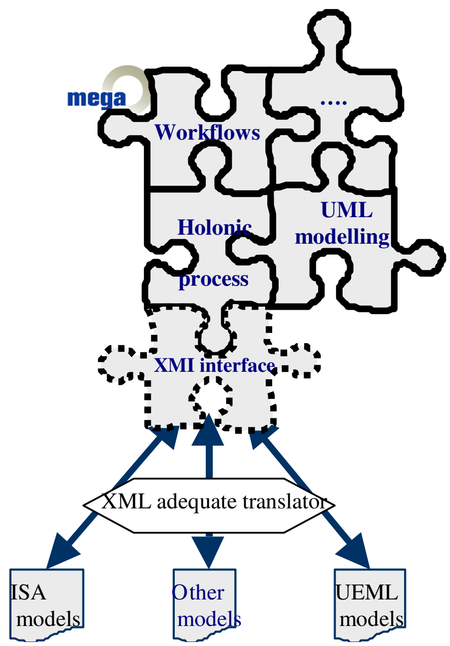

Fig. 7. Transformation of models

4 Conclusion

In this paper we defined an approach for specifying Holonic Process Models. In these models, the holon represents the key notion. This concept enables maintaining coherence between the physical objects and their informational views, and thus between informational flows and physical flows. This approach aims also at increasing model abstraction in order to simplify its initial comprehensiveness and ensure coherence between different views of manipulated objects. We gave a proposal to map and integrate the holon concept in existing enterprise modelling frameworks for interoperability concerns. In this proposal, we focus on ERP interconnection with an MES in order to exchange manufacturing and production related information. In order to validate our approach, further work will be to develop translators from our XMI interface to the respective B2MML and UEML XML files. Perspectives for this work concern the interconnection with a WfMS, in order to exchange business process modelling information and then enabling a process oriented execution of enterprise models.

References

[1] Molina A., Chen D., Panetto H., Vernadat F. and Whitman L. (2004). Enterprise Integration and Networking: Issues, trends and vision. IFIP International Conference on Enterprise Integration and Modeling Technology (ICEIMT'04), Kluwer Academics Publisher, October 9th-11th, Toronto, Canada.

[2] Petrie C. *Entreprise Modeling and Integration*. C. Petrie (ed). McGraw-Hill. 1992.

[3] AMICE, 1993 AMICE (eds.). CIMOSA: Open System Architecture for CIM, 2nd revised and extended edition. Springer-Verlag, Berlin, 1993.

[4] Formal Support for Representing and Automating Semantic Interoperability Yannis Kalfoglou and Marco Schorlemmer.

[5] Vallecillo, A., Hernàndez, J., and Troya, J. M., Component Interoperability.

[6] Yutaka S. Information unification between enterprise resource planning system and production control system, Yokogawa Technical Report, English Edition, N° 25 (1998)

[7] Workflow Management Coalition: The Workflow Reference Model. Document Number TC-1003, January - 1995

[8] Workflow Management Coalition: Workflow standard – Interoperability abstract specification. Document Number WFMC-TC-1012, 26-October-1995

[9] Workflow Management Coalition: Workflow standard – Interoperability Wf-XML Binding. Document Number WFMC-TC-1023, 1-May-2000

[10] Zur Muehlen, Michael; Klein, Florian: AFRICA: Workflow Interoperability based on XML-messages. CAiSE 2000 Workshop on Infrastructures for Dynamic Business-to-Business Service Outsourcing (ISDO ’00). Stockholm, Sweden, June 5-6, 2000

[11] Shepherdson J.W., Thompson S.G. and Odgers B. (1999). Cross Organisational Workflow Co-ordinated by Software Agents Work Activity Co-ordination and Collaboration (WACC '99) Workshop Paper, February 1999

[12] Casati, F.; Ceri, S.; Pernici, B.; Pozzi, G. Semantic workflow interoperability. In Proc. of the 5th Conf. on Extending Database Technology (EDBT), pages 443--462, 1996.

[13] Casati, F.; Ceri, S.; Pernici, B.; Pozzi, G.: Deriving active rules for workflow enactment. In: 7th Intl. Conf. on Database and Expert Systems Application. 1996, pp. 94-115

[14] Berio G, et al. D3.2: Core constructs, architecture and development strategy, UEML TN IST – 2001 – 34229, March 2003

[15] Panetto H., Berio G., Benali K., Boudjlida N. and Petit M. (2004). A Unified Enterprise Modelling Language for enhanced interoperability of Enterprise Models. Proceedings of IFAC INCOM Symposium, April 7th-9th, Salvador de Bahia, Brazil

[16] The World Batch Forum. Business To Manufacturing Markup Language (B2MML), version 02, 2003.

[17] Koestler A. (1967) The Ghost in the Machine Arkana, London.

[18] Marik, V and Pechoucek, M (Eds.) (2001). Special issue on industrial applications of Holonic and Multi-Agent Systems. *Journal of Applied System Sciences*. **2/1**.

[19] Morel G., Panetto H., Zaremba M.B. and Mayer F. (2003). Manufacturing Enterprise Control and Management System Engineering: paradigms and open issues. IFAC Annual Reviews in Control. 27/2, 199-209, December

[20] Wyns, 1999, Reference architecture for holonic manufacturing system, Ph.D. Thesis, Katholieke Universiteit Leuven

[21] Seidel D. and Mey M. (1994). IMS – Holonic Manufacturing Systems: systems components of autonomous models and their distributed control.
[22] Gouyon D., Simão J. M., Alkassem K. and Morel G., (2004). Work in progress for product driven manufacturing automation, Proceedings of IFAC INCOM Symposium, April 7th-9th, Salvador de Bahia, Brazil
[23] Baïna.S, Panetto H. and Morel G. A holonic approach for application interopearbility in manufacturing systems environment, *In Proc of the 16th IFAC World Congress, Prague, July 4-8, 2005.*
[24] ISO/IEC 62264 (2002). ISO/IEC 62264-1:2002. *Enterprise-control system integration, Part 1. Models and terminology*, ISO
[25] UEML. (2003). Unified Enterprise Modelling Language (UEML) Thematic Network. IST-2001-34229, www.ueml.org.
[26] Vernadat F.B. (1996). *Enterprise modelling and integration: principles and applications.* Chapman & Hall.
[27] Object Management Group: XML Metadata Interchange Specification, version 2.0, May 2002.
[28] W3C. World Wide Web Consortium, Extensible Markup Language (XML) 1.0, 1998