ENTERPRISE MODELLING FOR INTEROPERABILITY

Vallespir B., Chen D, and Ducq Y.

LAPS/GRAI University Bordeaux 1 – ENSEIRB – UMR CNRS 5131
351, cours de la liberation   33405 TALENCE CEDEX   FRANCE
tel.: +33 5 4000 2408   fax: +33 5 4000 6644   e-mail: bruno.vallespir@laps.u-bordeaux1.fr

Abstract: This paper aims at presenting an overview on basic concepts and principles relating to interoperability of enterprise applications and software. After a review and tentative clarification on the definitions of interoperability and enterprise application, various domains of interoperability and related modelling languages will be discussed and their associated problems outlined. Existing typologies of interoperability according to different points of view will be tentatively presented. Conclusion will be given at the end of the paper. *Copyright © 2005 IFAC*

Keywords: Enterprise modelling, Interoperability, INTEROP NoE, Manufacturing systems, Enterprise integration

1. INTRODUCTION

Enterprise modelling techniques appeared in the 70's and 80's in the frame of American projects (ICAM) or European ones (CIMOSA for instance) and were developed during all the 80's and 90's through the wish of companies to establish a cartography and to have a coherent diagnosis of their practices. Conversely, the domain of interoperability between systems is quite new and its issue was coming first from the information system domain. Interoperability between systems has been an issue from the very beginning of IT. OSI was a first attempt to get standard communication between systems.

In the frame of INTEROP Network of Excellence, the idea was to study the interest to use enterprise modelling techniques to solve the problem of interoperability of enterprise applications and software.

This paper aims at presenting some tracks of solutions in this domain using enterprise modelling techniques. So, after a part dedicated to definitions of what interoperability is at the level of enterprise applications or at the level of a system, and definitions of enterprise modelling and enterprise application, the difference between the modelling of interoperability and the interoperability of modelling will be presented. Then, the domains of interoperability will be presented. This part aims at showing which domains of interoperability can be solved with using enterprise modelling techniques. These domains are concerning mainly the communication of IT systems and resources, the semantic and knowledge problems associated to interoperability, the compatibility between practices, and the compatibility between the controls of practices. In a last part the different degrees of interoperability are presented in order to synthesise the various problems. In conclusion, the future researches in this domain will be presented as well as the links with other research domains like architecture and platform or ontology.

2. DEFINITIONS

Before going further, some definitions are necessary such as interoperability, enterprise application and enterprise modelling. Some remarks on their status are added as well.

2.1 Interoperability

Research in the interoperability domain in Europe remains badly structured, fragmented, sometimes overlapping. There is no global vision of the consistency and no co-ordination between various European research centres, university laboratories, or other bodies. This situation is not only true for research, but also in the training and education areas. Commonly, interoperability is understood as a measure on the ability of performing interoperation between two different entities (be they software applications, processes, systems, organisations…). According to Oxford Dictionary, interoperable means 'able to operate in conjunction'. The word "inter-operate" also implies that one system performs an operation on behalf of another system. Originally, the concept of 'interoperability' comes from software engineering. From this point of view, interoperability means that two co-operating software packages/ applications/systems can easily work together without a particular interfacing effort. It also means establishing communication and sharing information and services between software applications encompassing communication protocols, hardware software, application, and data compatibility layers. In other words, it describes whether or not two software that were developed with different tools and from different vendor can work together. The ISO/IEC 16100 standard defines the manufacturing software interoperability as 'ability to share and exchange information using common syntax and semantics to meet an application-specific functional relationship through the use of a common interface' (ISO 16100, 2000).

Generally, interoperability is not only concerned with software applications. It may happen between any two entities in a heterogeneous or homogenous networked environment. More generally, at a level of a system, the interoperability can be seen as the ability of systems, units, or forces to provide data, information, material, and services to and accept (receive) the same from other systems, units, or forces and to use the data, information, materiel, and services so exchanged to enable them to operate effectively together.

When the system is a set of software, interoperability is the characteristic demonstrated by two independent software components of exhibiting predefined behaviours (i.e. implemented during design) in a consistent manner, at the levels of user interface, data instance sharing, etc.

In the IDEAS project, interoperability is considered achieved if the interaction can, at least, take place at the three levels: data, application and business process with the semantics defined in the business context (IDEAS, 2003). The Open Group has considered interoperability as: '(1) the ability of two or more systems or components to exchange and use shared information and (2) the ability of systems to provide and receive services from other systems and to use the services so interchanged to enable them to operate effectively together' (Open Group, 2000). In (Vernadat, 1996), interoperability is defined as the ability to communicate with another system and use the functionality of the other system (interoperability = communication + interaction). However, integration is more large and broad (integration = communication + co-operation + co-ordination).

2.2 Enterprise application

The term 'application' is often misunderstood as a synonym of software. According to ISO 15745, an (manufacturing) application can be modelled as a combination of a set of processes, a set of resources, and a set of information structures that are shared and exchanged among the resources. In ENV 12204 (a European pre-standard), three types of resource have been considered: machining, computing and human types. An UML model of enterprise application, combining the definition of ISO 15745 and ENV 12204, is shown in figure 1 (ENV12204, 1995).
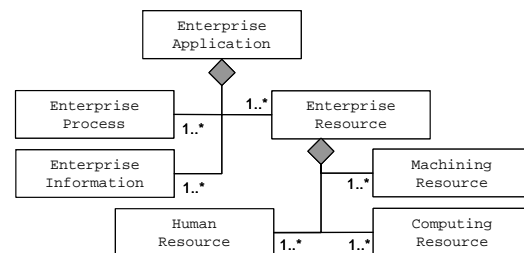

Fig. 1. An UML model of enterprise application

2.3 Enterprise modelling

A company is a complex system which must be modelled for several reasons. The first one is to understand the enterprise. In this way, the prime goal of enterprise modelling is to support analysis of an enterprise (Vernadat, 1996). Secondly, the purpose of enterprise modelling is to support any actor having a project of design or evolution concerning the enterprise.

An enterprise model is always related to a finality, i.e. the goal of the modeller. Moreover, it must be able to represent several points of view related to the modeller's needs (such as structural, functional or behavioural) and express the reality at different abstraction levels (conceptual, organizational, and technical). Finally, a specific approach is often linked to each modelling language, describing the various steps necessary for the building and use of the model (life cycle).

This statement explains why the research works in the domain of enterprise modelling have led to many languages and modelling tools. Then, it is very difficult to give an exhaustive list of them. Enterprise modelling uses modelling languages which may be informal (as natural languages), semi-formal (graphical languages) or formal (mathematical languages) (Chapurlat et al., 1999). Among semi-formal languages, first results came up in United States during the seventies and led to SADT, SSAD, IDEF0, Data Flow Diagram. In Europe, programmes of the European commission enabled the development and diffusion of these tools (Esprit programme, 80's). Since then, numerous modelling

languages were developed both in North America and Europe such as MERISE, NIAM, M*, CIM-OSA, OMT, IEM, IDEFx, METIS or ARIS Toolset (Vallespir *et al.*, 2003).

Finally, an enterprise model aims at formalizing a part of the enterprise or the enterprise as a whole in order to understand and explain a current situation or to design and validate a project (Braesch *et al.*, 1995). Then, enterprise modelling may be understood as a way to express a part of the knowledge of the enterprise.

2.4 Modelling of interoperability vs. interoperability of models

When speaking about the relations between enterprise modelling and interoperability, confusion often comes up between two notions, i.e. modelling of interoperability and interoperability of models.

Whatever the definition of interoperability, the concept of modelling interoperability aims at answering the question: *"how to model the capability of a system to be interoperable?"* or *"how to model that several systems are interoperating?"* Because there may exist different classes of interoperability or modes of interoperation, the answer cannot be simple and different solutions have to be provided. For each domain of interoperability that may be defined, the question can be asked and the answer will be different. Section 3 of this paper presents several domains of interoperability. Finally, answer to these questions is precisely the purpose of this paper.

Interoperability of models deals with another notion. Let us assume that a system is modelled with two different languages (because for instance this system needs to be modelled from several points of view). Let us assume now that these models need to be compared, merged, or proceed. This is possible only if the semantics of the models are the same or, at least compatible for their part involved. This is this compatibility that may be called interoperability of models. This concept is the basis of research works about Unified Enterprise Modelling Language (UEML) as presented in chapter 3.3.

In conclusion, interoperability of models is not equivalent to modelling of interoperability but constitutes a domain of it.

3. DOMAINS OF INTEROPERABILITY AND RELATED MODELLING LANGUAGES

This section presents a tentative classification of interoperability. For each class, it is shown how enterprise modelling may express interoperability. In each case, only few modelling languages are presented, this does not mean at all that other suitable modelling languages do not exist.

3.1 Communication (data interoperability)

Interoperability is first of all a capacity related to communication between various entities involved in interoperation. The communication problem can be (1) human to human, (2) machine to machine, (3) human to machine and (4) machine to human. From a modelling point of view, this level of interoperability may be easily represented by any language representing data flows. However, communication problems may appear, at (i) syntactic, or (ii) semantic (see § 3.2) levels.

Concerning the syntactic mismatch, according to (Missikoff, 2004), there are two cases:

- data syntax used by the two parties is different,
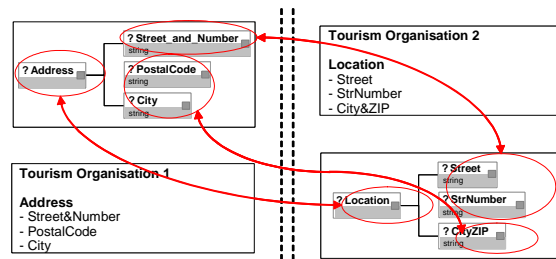- data structure used by the two parties is different (ex. figure 2).

Fig. 2. Same meaning, different structure (Missikoff, 2004)

3.2 Semantic interoperability

To achieve interoperability, different parties involved in interoperation must share the same meaning of the vocabulary used and exchanged. In other words, the semantics of information must be interpreted in the same way by all agents involved in the communication process so that they can understand each other. This is known as the difficult problem of semantic unification. This unification is today partially resolved by conceptual data modelling with the support of languages such as entity/relation or UML class diagram.

This issue is today addressed by the definition of ontology (Vernadat, 1996). Ontology is a set of concepts, corresponding to the relevant entities of the domain. The concepts must be agreed on by the domain community and meaningful for doing business. Concepts are represented by a terminology that is used to annotate the enterprise entities, such as: object, actors, processes, events, messages, rules, and documents (ATHENA, 2004). Ontology must not be information technology dependent. A suitable mechanism for reconciling semantic mismatch, between software applications that need to cooperate, has to be developed.

3.3 Interoperability of knowledge expressed by the mean of modelling languages

We shall consider here the part of knowledge of the enterprise expressed through the use of enterprise modelling languages (as introduced in section 2.3.). In this way, knowledge interoperability is practically related to modelling languages interoperability. Because the large amount of languages existing, this interoperability is not so easy to get. This issue is tackled by analyzing modelling languages, i.e. decomposing them in elemental modelling constructs. Then, interoperability between two modelling languages may be stated by expressing the

part of semantics common to the two languages. So, the knowledge interoperability may be understood as a case of semantics interoperability in the domain of modelling.

Practically, the approach is, first to define a common language by gathering modelling constructs coming from existing languages and, second, to define mapping mechanisms between each language and the common language. The development of this common language is the topic of several initiatives around the world under the acronym of UEML (Chen, 2002; Panetto, 2001). The definition of modelling constructs is proceeding by meta-modelling languages generally by using class diagrams. Figure 3 gives as example the part of the meta-model of IEM related to one specific model and expressed with UML classes diagram. This meta-model was built up during the UEML project (UEML, 2003).
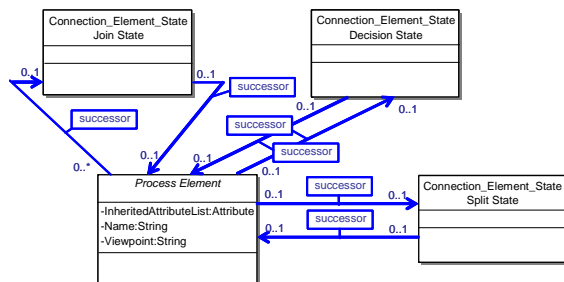

Fig. 3. Partial meta-model of IEM

3.4 Practices interoperability

Enterprise modelling aims to describe the enterprise practices from several perspectives: functional, physical, business process, decisions, information.

The interoperability has to be considered at all of these levels in order to improve the running between two enterprises or systems. In particular, the interoperability between business processes is an important issue to synchronise the flow of products or services as well as the flow of information. The synchronisation allows ensuring that a product delivered by one enterprise can be received in this status by the following one. Two solutions can serve the interoperability between practices: standardisation and mutual adjustment of practices.

Standardisation of practices. One of the solutions, with using enterprise modelling, is to allow the standardisation of practices of both systems as shown in figure 4.

More than the synchronisation of flows, the second interest of this standardisation is to avoid redundancy of activities like quality control or order confirmation between two partners in particular in the case of the supply chain. The third benefit of this standardisation is to allow defining close specifications of ERP systems for both systems with will lead to interoperability in enterprise applications. Finally, this standardisation will allow facilitating the definition of common objectives and interoperability of decisions through coordination, as defined in the following part.

Nevertheless, the standardisation of practices, even if this can be considered as the most robust solution,

can lead to implementation of drastic changes in both enterprises that can disturb strongly the rest of enterprise running. Then, another solution can consist of synchronisation of practices.
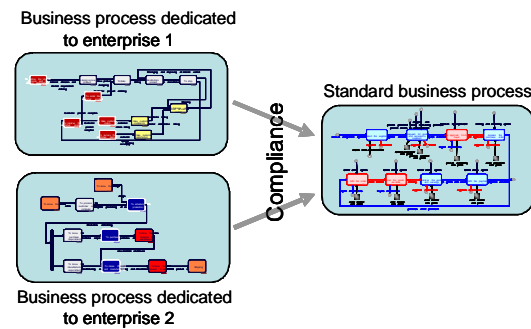

Fig. 4. The standardisation of practices

Mutual adjustment of practices. This solution consists in the use of enterprise modelling to design few supplementary activities (supplementary business process) that allow to link both business processes of the two enterprises or systems, as shown in figure 5. The advantage of this configuration is to not disturb existing organisations in case they are optimised but it obliges to invest in a new business process that it is necessary to control using decisional model. This control of this supplementary business process must be coordinated with the ones of both other systems.
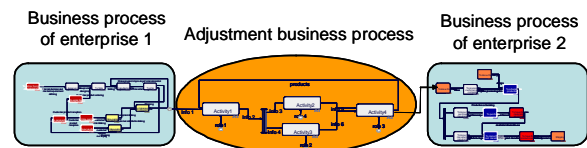

Fig. 5. The mutual adjustment of practices through a specific business process

Business oriented modelling languages can be useful for the standardisation and synchronisation issues.

Time aspects. Developing interoperability or more generally speaking, developing cooperation and collaboration between different parties (both inter and intra organisations) also concerns the time aspect. It mainly refers to synchronisation of activities / processes, i.e. some activities and processes must be done before a certain time point or some activities / processes must be carried out conjunctly or simultaneously. Standardisation and mutual adjustment are both solutions for time aspect. Thus, business oriented modelling languages are useful for the time aspects as well.

The specific situation of design and engineering. Practices interoperability is also a problem during design and engineering phase. Within this framework, practices we are speaking about are related to all activities proceed to design a system (analysis, design, evaluation, simulation, etc.) and supported by analysis and design methods.

Here, the interoperability issue can be addressed by decomposing methods into elementary elements. This analysis allows to know what part of each

method is specific and what can be put in common. In fact, interoperability of methods is related to this commonalty. The principle is the same that UEML with modelling languages and this approach is very close to method engineering (Rolland, 2004).

In this domain, enterprise modelling would be useful if it would be used to model and analyse methods. It is not the case today. However, some research works are related to this topic such as GERAM (Williams, 1994; GERAM, 1999) (figure 6).
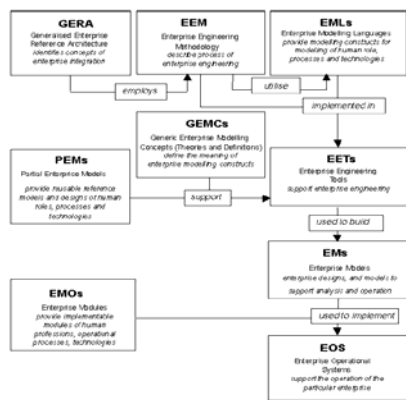

Fig. 6. GERAM

3.5 Coordination and objectives interoperability

One of the highest levels of interoperability is to consider that more than communicate and act together, two systems are interoperable if only they actively participate to a same objective i.e. to be co-ordinated. That means that two systems acts to match objectives that have been defined by another system considered being at a higher level (the co-ordinator). The purpose of the GRAI grid (Doumeingts, 1995, 1998) is to globally model a management system and to express co-ordination links. Figure 7 gives an example of GRAI grid. In this figure co-ordination links are represented by double arrows.
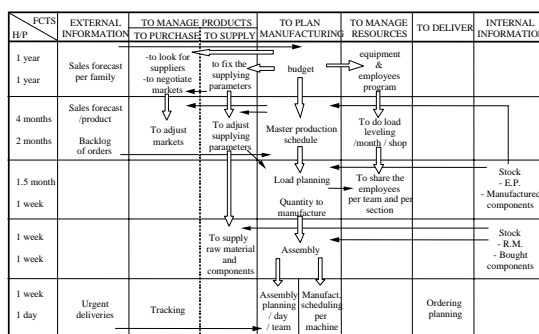

Fig. 7. GRAI grid

4. TYPOLOGY OF INTEROPERABILITY

Section 2.1 showed that several definitions of interoperability exist. Furthermore, many domains concerned with interoperability exist as well, as presented in section 3. This is why a typology is necessary.

Interoperability can be established at different levels with various focuses and degrees. This section tentatively reviews existing typologies relating to interoperability.

4.1 Typology according to the degree of compatibility

(IEC TC65/290/DC, 2002) has characterised the concept of interoperability as a certain degree of compatibility (figure 8). The full interoperability means that "The application data, their semantic and application related functionality of each device is so defined that, should any device be replaced with a similar one of different manufacturer, all distributed applications involving the replaced device will continue to operate as before the replacement, but with possible different dynamic responses". For instance, two organisations which share only the three last system features are considered as inter-connectable.

Compatibility level / System feature	Incompatible	Coexistent	Interconnectable	Interworkable	Interoperable	Interchangeable
Dynamic Behaviour						x
Application Functionality					x	x
Parameter Semantics					x	x
Data Types				x	x	x
Data Access			x	x	x	x
Communication Interface			x	x	x	x
Communication Protocol		x	x	x	x	x

Fig. 8. Compatibility levels (IEC TC65/290/DC)

4.2 Typology according to the degree of integration

Interoperability can be established at different levels with various focuses and degrees. This section reviews existing typologies relating to interoperability. ISO 14258 considers that interoperation occurs between two (or more) entities, whatever the nature of these entities (industrial systems, software applications …). There are three ways to relate them to one another (ISO, 1999):

- Integrated with a standard format for all constituent systems. Diverse models are interpreted in the standard format. This format must be as rich as the constituent system models. This is so called 'total integration or full integration'.

- Unified with a common meta-level structure across constituent models, providing a means for establishing semantic equivalence. The metamodel is not an executable entity but a neutral model allowing mapping between diverse models.

- Federated where models must be dynamically accommodated rather than having a predetermined meta-model. This assumes that concept mapping is done at an ontology level, i.e. semantic level. This is considered as 'full interoperability'.

This view clarifies the difference between full integration and interoperability. Two integrated systems are interoperable; but two interoperable ones are not necessarily integrated.

4.3 Other dimensions of the typology

The typology coming up from the domains presented chapter 3 covers many operational points of view about interoperability such as processes, software applications, data or communication interoperability. Moreover, these interoperability classes may be considered by several approaches as for example:
- horizontal interoperability (interoperation between resources at the same organisation level) vs. vertical interoperability (interoperation between resources at the different organisation levels);
- inter-enterprise interoperability (interoperation between resources in the same enterprise) vs. intra-enterprise interoperability (interoperability between different enterprises).

5. CONCLUSION

This paper shows the different issues of interoperability that can be solved using enterprise modelling techniques. However, even if many classical domains of interoperability are covered by the typology presented in this paper, it is sure that the exhaustively is not achieved. This paper is a tentative and contributes to the explicit participation of enterprise modelling to the solution of interoperability issues. The INTEROP project addresses some of these issues and in particular the interoperability of enterprise applications and software. Nevertheless, this domain is emergent and the approach to consider the impact of ontologies and architecture and platform research beside the one on enterprise modelling is quite new. In the future, the main researches in this domain will concern mainly the standardisation of practices, the interoperability of enterprise modelling techniques through UEML and the automatic parameterisation of enterprise applications based on enterprise models.

REFERENCES

ATHENA (2004). Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Applications, FP6-2002-IST-1, Integrated Project Description of Work.

Braesch C., Haurat, A. and Beving J.-M. (1995). L'entreprise-système. – in *La modélisation systémique en entreprise*, Paris: Hermès.

Chapurlat, V., Larnac, M., Lamine, E. and Magnier, J. (1999). Definition of a formal analysis framework for existing enterprise modelling approaches. – in proceedings of ASI, Louvain, Belgium, 22-24 September.

Chen, D., Vallespir, B. and Doumeingts, G. (2002). Developing an unified enterprise modelling language (UEML) – Roadmap and requirements. – in proceedings of PROVE, Sesimbra, Portugal, 1er-3 May, Kluwer Academic Publishers.

Doumeingts, G., Vallespir, B. and Marcotte, F. (1995). A proposal for an integrated model of manufacturing system: application to the re-engineering of an assembly shop. – in *Control Engineering Practice*, vol. 3, n° 1, Oxford: Pergamon Press Ltd.

Doumeingts, G., Vallespir, B. and Chen, D. (1998). Decisional modelling GRAI grid. – in *International handbook on information systems*, Bernus P., Mertins K. and Schmidt G. ed., Berlin: Springer.

ENV 12204 (1995). Advanced Manufacturing Technology - Systems Architecture - Constructs for Enterprise Modelling, CEN Brussels.

GERAM (1999). GERAM: Generalised Enterprise Reference Architecture and Methodology. – Version 1.6.1, IFIP-IFAC Task Force on Architectures for Enterprise Integration, March.

IDEAS (2003). IDEAS Project Deliverables (WP1-WP7), Public Reports, *www.ideas-road map.net*, 2003.

IEC TC65/290/DC (2002). Device Profile Guideline, TC65: Industrial Process Measurement and Control, June 28.

ISO (1999). ISO 14258 - Industrial Automation Systems - Concepts and Rules for Enterprise Models, ISO TC184/SC5/WG1, April 14.

ISO DIS 16100 (2000). Manufacturing Software Capability Profiling, Part 1 - Framework for interoperability, ISO TC/184/SC5, ICS 25.040.01.

Missikoff, M., Taglino, F. and Schiappelli, F. (2004). ATHENA Kick-off meeting, Session 12/A/3 Tutorial: Knowledge Management and Ontologies for Interoperability, 13 February.

Open Group (2000). TOGAF: The Open Group Architecture Framework, Document No. 1910, Version 6, December.

Panetto, H., Mayer, F. and Lhoste, P. (2001). Unified Modelling Language for meta-modelling: towards Constructs definition. – in proceedings of INCOM'01, Vienna, Austria, 20-22 September.

Rolland, C. (2004). Evolution des méthodes d'ingéniérie des systèmes d'information. Deuxième école de modélisation d'entreprise, Nîmes, France, 10-12 March.

UEML (2003). Examples of application. – Deliverable 3.2, WP3, project IST - 2001 - 34229 UEML, Berio G. ed., Turin, Italy.

Vallespir, B., Braesch, C., Chapurlat, V., Crestani, D. (2003). L'intégration en modélisation d'entreprise : les chemins d'UEML. – in proceedings of MOSIM, Toulouse, France, 23-25 april.

Vernadat, F.B. (1996). Enterprise Modelling and Integration: Principles and Applications, Chapman & Hall, London.

Williams, T.J., Bernus, P., Brosvic, J., Chen, D., Doumeingts, G., Nemes, L., Nevins, J.L., Vallespir, B., Vlietstra, J. and Zoetekouw, D. (1994). Architectures for integrating manufacturing activities and enterprises. – in *Computers in Industry*, vol. 24, n° 2-3, Amsterdam: Elsevier