HAL
open science

The Missing Link: Culture And Language Barriers To Interoperability

Larry Whitman, Hervé Panetto

To cite this version:

Larry Whitman, Hervé Panetto. The Missing Link: Culture And Language Barriers To Interoperability. IFAC Annual Reviews in Control, 2006, 30 (2), pp.233-241. 10.1016/j.arcontrol.2006.09.008 . hal-00115878

HAL Id: hal-00115878

https://hal.science/hal-00115878

Submitted on 23 Nov 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Whitman L. E., Panetto H. (2006). The missing link: culture and language barriers to interoperability. *IFAC Annual Reviews in Control*. **30/2**, 233-241, extended version from *9th IFAC Symposium on Automated Systems Based on Human Skill And Knowledge*, May 22-24, 2006, Nancy, France, Elsevier, December 2006, ISSN: 1367-5788,
http://www.elsevier.com/wps/find/journaldescription.cws_home/429/description

THE MISSING LINK: CULTURE AND LANGUAGE BARRIERS TO INTEROPERABILITY

Lawrence E. Whitman[1] and Hervé Panetto[2]

[1]*Industrial & Manufacturing Engineering Department, Wichita State University, USA*
larry.whitman@wichita.edu)
[2]*Research Centre for Automatic Control, CRAN-UMR 7039, Nancy-University, CNRS, France*
Herve.Panetto@cran.uhp-nancy.fr

Abstract: Interoperability is key to ensuring that a global supply chain operates as seamlessly as a vertically integrated organization. Much research has been accomplished and is on-going related to the technical, organisational and scientific issues concerning interoperating dissimilar enterprise systems and languages. However, there are significant issues concerning interoperating information across the barriers of cultures and national languages. This paper presents the key drawbacks regarding the cultural and language barriers to true exchange of knowledge. The paper then presents an example enterprise model in light of these identified issues. *Copyright © 2006 IFAC*

Keywords: Enterprise integration, Enterprise modelling, Socio-technical issues.

1. INTRODUCTION

The activities within an enterprise are complex as companies manufacture a variety of products using different production methods to satisfy different customers. In order to face global competition and fluctuating market conditions, companies require the management of change (Vernadat 1996). To simplify managing the change, Enterprise Integration (EI) plays an important role. Enterprise Integration consists of connecting and making interoperable all functional areas of an organization. Interoperability will improve organization's synergy in achieving its mission and vision in effective and efficient manner (Molina *et al.*, 2004). The process of integrating an enterprise is simplified by using a model. Enterprise

models are often used to depict these various activities within an enterprise. An enterprise model is defined as "the art of externalising enterprise knowledge, which adds value to the enterprise or needs to be shared" (Vernadat 2000). Developing an enterprise model provides a common understanding of the process and the associated activities.

There are various factors that drive changes to the internal processes of the organization. There are changes in market conditions due to competitor or customer related issues. The emergence of new technologies and changes due to change in product features exacerbates this speed of change (Harding & Popplewell 1999). Basically, as the organization or business objective changes so must the enterprise model change to maintain the accuracy of information. Nowadays, the enterprise model is not reused to accommodate changes in the process. One reason is due to the inability of the enterprise to be aware of existing models.

An enterprise modelling language is required to create an enterprise model. There are many enterprise modelling languages and tools available and each modelling language has different characteristics (Petit *et al.*, 2002). "This intensive production of tools has led to a Tower of Babel situation in which the many tools, while offering powerful and distinct functionalities, are unable to interoperate and can hardly or not at all communicate and exchange models" (Panetto *et al.*, 2004b). Each enterprise modelling language and tool has its own characteristics, features and suitability for each type of industry and kind of modelling objective. It is impossible to force industries to only utilize one type of enterprise modelling language and one enterprise modelling tool. Diversity of enterprise models may create obstacles in achieving the goal of enterprise integration. To respond to these obstacles, interoperability of enterprise modelling languages is needed.

IEEE (1990) defines interoperability as "the ability of two or more systems or components to exchange information and to use the information that has been exchanged.". Even though there are many different enterprise modelling languages, it would be productive if they can interchange with each other. In addition, interoperability leads to a standard for different languages. One example of interoperability language currently in development is called the Unified Enterprise Modelling Language (UEML) (Berio *et al.*, 2002; Panetto et al., 2004a). UEML is not another enterprise modelling language, but it is a meta-model that supports interoperability of common enterprise modelling languages. The issue of

interoperability has been dealt with extensively from a technical perspective, but only to a limited extent from a cultural and language perspective. This paper continues by discussing interoperability, continues with knowledge management, and then presents some issues with the exchange of knowledge.

2. INTEROPERABILITY

The word interoperability has many wide uses. The term interoperability is increasingly used in enterprise engineering and its related standardization activities (Chen & Vernadat 2002). To achieve interoperability, the systems need to interoperate their data, resources and business processes with semantics defined in a business context regardless of different languages, data formats, interfaces, executions platforms, communication protocols or message formats (Tsagkani 2005). Interoperability is not only about transferring information but also performing an operation on behalf of another system.

One of the main obstacles to interoperability arises from the fact that the systems that support the functions in many enterprises were created independently, and do not share the same semantics for the terminology of their process models. Interoperability requires data stored in software systems on one machine to be sent and interpreted by another software system on another machine and for different purposes. To make these happen, standards on message format and transfer are needed.

1.1 Levels of interoperability.

To achieve interoperability of an enterprise, four levels are needed to achieve. The first level is technical interoperability, which then continues with syntactic, semantic and pragmatic interoperability.

Technical interoperability means transporting messages from one application to another. With today's technologies, technical interoperability is not the issue anymore. Current applications have full support in technical interoperability. Many applications have been able to understand each other. For example, a file from one application can be translated and read with a different application.

Syntactic interoperability is the second level. Before performing data exchange, the systems must agree on the format for data exchange. The eXtensible Markup Language (XML) (W3C, 2004a) has solved this issue. XML has addressed the syntactic interoperability issue (Lilleng 2005).

Semantic interoperability means understanding the content of messages and models in the same way by senders and receivers. Semantic interoperability refers to a system's ability for exchanging information inside organizations with heterogeneous information and-or between organizations without having to do tailoring to make this possible (Lilleng 2005). Semantic interoperability is the focus of this paper.

Finally, pragmatic interoperability captures the willingness of partners for the actions necessary for the collaboration. This willingness to participate involves both capability of performing a requested action, and policies dictating whether the potential action is preferable for the enterprise to be involved in collaboration (Tsagkani 2005).

The syntactic level is the most effective solution, but not totally sufficient, to achieve practical interoperability between computerized systems using existing technologies such as XML (eXtensible Mark-up Language) and its related applications (SOAP (Simple Object Access Protocol)) (W3C 2003), WSDL (Web Services Description Language) (W3C 2004b), ebXML (Electronic Business XML Initiative) (OASIS 2002), to name a few. In fact, current research work (Vernadat 1996; UEML 2003; INTEROP 2003; Panetto et al. 2004b) deals with trying to enable a seamless data and model exchange at the semantic level. In that sense, standardisation initiatives (ISO 14528 1999; ISO 16100, 2002; IEC 62264 2002; ISO 19440 2004) then try to cope with this issue by defining generic constructs focusing on the domain concept definitions. For more details on standardisation initiatives around integration and interoperability, the reader may refer to (Chen and Vernadat 2002).

Interoperability is a means to achieve integration (Chen and Vernadat 2002). The difference between integration and interoperability has been further clarified in ISO 14528 (1999) (Concepts and rules for enterprise models). This standard considers that models could be related in three ways: (1) integration when there exists a standard or pivotal format to represent these models; (2) unification: when there exists a common meta-level structure establishing semantic equivalence between these models; and (3) federation when each model exists per se, but mapping between concepts could be done at an ontology level to formalise the interoperability semantics.

Integration is generally considered to go beyond mere interoperability to involve some degree of functional dependence. While interoperable systems can function independently, an integrated system

loses significant functionality if the flow of services is interrupted. An integrated family of systems must, of necessity, be interoperable, but interoperable systems need not be integrated. Integration also deals with organisational issues, in possibly a less formalised manner due to dealing with people, but integration is much more difficult to solve, while interoperability is more of a technical issue.

Compatibility is something less than interoperability. It means that systems/units do not interfere with each other's functioning. But it does not imply the ability to exchange services. Interoperable systems are by necessity compatible, but the converse is not necessarily true. To realize the power of networking through robust information exchange, one must go beyond compatibility.

Additionally, from a cultural point of view, organisational interoperability is a significant issue. If you compare occidental engineers and oriental engineers, you may think that the organisation of the work is influencing the interoperability.

These aspects of interoperability are coherent with the definitions proposed by the European Interoperability Framework (EIF 2004), which considers three aspects of interoperability:

Organisational Interoperability: This aspect of interoperability is concerned with defining business goals, modelling business processes and bringing about the collaboration of administration that wish to exchange information and may have different internal structures and processes. Moreover, organisational interoperability aims at addressing the requirements of the user community by making services available, easily identifiable, accessible and user-oriented.

Semantic Interoperability: This aspect of interoperability is concerned with ensuring that the precise meaning of exchanged information is understandable by any other application that was not initially developed for this purpose. Semantic interoperability enables systems to combine received information with other information resources and to process it in a meaningful manner. Semantic interoperability is therefore a prerequisite for the front-end multilingual delivery of services to the user.

Technical Interoperability: This aspect of interoperability covers the technical issues of linking computer systems and services. It includes key aspects such as open interfaces, interconnection services, data integration and middleware, data presentation and exchange, accessibility and security services.

In sum, interoperability lies in the middle of an “Integration Continuum” between compatibility and full integration, taking into account cultural requirements. It is important to distinguish between these fundamentally different concepts of compatibility, interoperability, and integration, since failure to do so sometimes confuses the debate over how to achieve them. While compatibility is clearly a minimum requirement, the degree of interoperability/integration desired in a joint family of systems or units is driven by the underlying operational level of those systems crossing the cultural/organisational matter of the involved enterprises.

3. MANAGING KNOWLEDGE IN ORGANIZATIONS

This section presents the basic concepts involved in knowledge management. There are various indispensable components required for strategically managing knowledge. As an initial step, the company must first identify its knowledge assets and then it should share that knowledge across the knowledge network and learn from experience. There are three main aspects of knowledge management: storage, transfer and transformation of knowledge, which refer to the three main concepts of the General System Theory (Bertalanffy 1969).

In this context, it is also important to discuss the various areas in which knowledge is dispersed in an organization:

Individual: Knowledge can be found in the hands of an individual worker who serves as a fundamental unit in the process of knowledge creation, storage, and use within the enterprise. Many times this knowledge is tacit and therefore not well documented.

Group: Group knowledge is more powerful than the sum of the knowledge acquired by an individual. This knowledge can be both formal and informal and is frequently intangible but is one of the most important knowledge assets within a company.

Organizational: The organization, in turn, serves as a storehouse of knowledge with its own peculiar structure and divisions of functions, with multiple processes and activities to aid in the search for knowledge.

Knowledge links: Any organization has certain links with suppliers and customers and they exchange knowledge during their course of operation. This is believed to be more effective as inter-organizational links can provide more information than an isolated organization.

Finally, the common essence of models is to recognize the different levels of knowledge assets that reside within an organization. This will give an idea of where to look for information when you start the process of storing the knowledge with the help of models. Knowledge is inherently dynamic and may guarantee long term competitiveness. Therefore, it is important for every organization to be able to learn and update the knowledge base periodically. This will ensure model is constantly reused to maintain relevancy (Dutta 1997).

4. CULTURAL INTEROPERABILITY?

As previously mentioned, semantic interoperability and knowledge interchange is the ultimate goal of an interoperability effort. The common understanding of the models leading to a better understanding of the processes leads to the final goal of pragmatic interoperability where the senders and receivers have the same actions in the same process. For this common action and understanding, cultural issues must be identified and addressed. This section lists some of those issues and provides examples.

When engineers design there is obviously much skill involved. However, there is also much tacit knowledge involved as well. An experienced engineer uses some tacit rules involved in their design process. There are many assumptions involved that are not explicitly stated. Design rules do not apply in every circumstance therefore they are not listed as design laws, but rather as design heuristics. In the US this is frequently referred to as “rules of thumb.” Interestingly, there are similar terms in other languages. Koen (2003) presents several of these idioms:

- In France *le pif* (the nose)
- In Germany, *Faustregel* (the fist)
- In Japan, (measuring with the eye)
- In Russia, (by the fingers)”

So, it is important to communicate this tacit knowledge as well. Unfortunately, tacit knowledge in one culture may be explicit and that which is explicit in one culture may be tacit in another. Frequently,

enterprise models are considered to be ‘documenting the obvious’. But, again what may be ‘obvious’ in one culture (and frequently between departments in a single enterprise) is not ‘obvious’ to others. Some research work is currently on-going in the domain of semantics annotation of models, based on common ontology, in order to deal with this issue (Boudjlida, *et al.* 2006) however real applicable solutions are fare away.

Language is not the only cultural issue in semantic interoperation. There is also the concern of different cultures having different design philosophies. Cultures have different constraints and different objectives. If engineering is, “design under constraint.” Then, as those constraints (and objectives) change, so does the design process. Again, another example from Keon (2003, pg 76) concerning nuclear engineers in America and Russia. The American engineers tested many different designs before selecting the final design. Whereas, Russian engineers decided quickly on the design and then made it work. If the initial design choice is near optimum, the Russian method is better compared to the initial objective. If the initial design choice is not good, the American method is better as it allows alternate designs. So, in the exchange of a process model, each method carries with it an inherent design philosophy. If engineers from the two countries were collaborating on a joint design, a process model would help point out the difference in design (cultural) philosophies. However, some of these concepts are tacit in their nature and might not be explicit.

Culture impacts business. Also linguistic issues impact business in the context of culture. Successfully interchanging business and engineering information requires Intercultural Communication Power (ICP) (deSilva and deSilva 1995). Whereas, engineers do not need to be completely culturally literate, the exchange of knowledge across dissimilar cultures in different native languages is imperative (Clark & Jones 1999).

The history of engineering progression is rich. The 1970s saw the incorporation of the factory with automation to allow the communication between different departments within the factory (CIM). The 1980s say the integration between engineering and manufacturing (CAD/CAM). The 1990s saw the integration of multiple sites and the beginning of supply chain integration. The 21st century will saw the integration of multiple companies in what is called extended and virtual enterprises. With globalization all this integration must now take place across multiple countries with different languages and cultures.

5. APPLICATION

With the context of the previous sections, an example application of these concepts is presented. The example chosen is that of the engineering change process. The engineering change process is common to all designs. As engineers want to make the world a better place, they frequently do this by changing and improving existing designs. Designs are usually currently in the manufacturing process in different stages of completion. So, the engineering change process must communicate those changes to be incorporated into the manufacturing process. The process of how these changes are communicated and incorporated is the examples application for this paper.

When a need for a change is identified, an example of a method for the technical details can be found in (Molina and Wright 2005). In this paper, the authors address the need for addressing more than the technical details, "Finally it is important to form, train and cultivate 'communities of e-engineering practice'. These communities must be formed with the mission to exchange experience, support collaborative learning, foster professional development and create the awareness to engage in trans-national and trans-cultural engineering partnerships" (Molina and Wright 2005).

As engineers incorporate changes between companies, the incorporation of changes and the communication of knowledge is now required across the globe. The desire is not simply to exchange models, but rather to interoperate knowledge. In the context of the example, the process of incorporating design change intent must be communicated between all partners.

An example of a change order process is shown in figure 1. The intent is to provide the basic complexity of the process. This process model was developed with an electronics manufacturing facility in the Midwest with facilities also located in the Ireland. A detail section of the process model is shown in figure 2. Figure 3 shows the same detail of the process model in another modelling language and in another language (French). The remainder of the paper will focus on the details of the process model shown in figure 2. Figure 2 shows that at a certain step in the process two asynchronous processes begin. The ECO and the associated BOM are analysed and this information is entered into the ORACLE database system. Two of these process steps include acronyms. There are three items that may appear as acronyms; ECO, BOMS, and ORACLE. Each of these terms is shown in both languages in table 1. The

term for bill of material is an acronym in English, but not in French. The term for the engineering change order is an acronym in both languages. The term Oracle may appear to be a strange word to anyone not familiar with the largest enterprise software vendor in the world may think that Oracle is an acronym or a strange term for engineers. To students just starting out, or even to a new engineer on the job, even this term may be confusing. To ensure commonality of terms, an ontology may be developed.

Ontology, as defined in philosophy, studies existence or being. In engineering and computer science, an ontology is defined as the conceptualization of a domain (Gruber 1993). Therefore, an ontology can aid in defining the terms and understanding the context of the terms. However, the ontology must move beyond just the required technical terms, but should include the cultural and linguistic issues that can confuse the transfer of knowledge.

Indeed, an ontology is used to express the semantics of models annotations when models exchange or transformations have to occur (Boudjlida, et al. 2006).

Model exchange relates to the issue of mapping concepts between models through ontology. These mappings may be either peer-to-peer between the exchanged models, or on a commonly agreed ontology of the domain where the models are used. In all cases, these mappings are usually incomplete because one model may be more expressive than the other or one model is not complete with regards to the modelled applications. Of course, in the second case, model completeness checking may apply and thus, annotations may be useful for that process but this is not always the case as completeness checking is a difficult task which is typically automated.

Moreover, model exchange is, at some extent, also related to model transformation. Indeed, once a model has been successfully exchanged through a communication medium, the model has to be transformed by the destination application in order to cope with its own internal model (Figure 4). Annotations are then used, together with the global mappings, to assist the transformation process. Similarly with research works in the domain of databases regarding data integration and schema integration (Halevy, 2001), global mappings may be associated to the GAV (Global-As-View) approach where local annotations deal with LAV (Local-As-View) approach.

In this situation, the issue is mainly syntactic transformation but also cultural transformation of partial models between two languages. Semantic annotations may help where semantic addition in the source model may influence the syntax of the destination model. As it is shown on Figure 5, a ticket booking model represented using the BPMN[1] is not as expressive as the same model represented using the MEGA Process language[2] (Figure 6). Indeed, BPMN is a graphical language for representing workflows where the main constructs are activities and information flows while MEGA Process language may be used to represent also other types of flows (material, financial, quality) and all elements regarding the technical architecture. The main difference in these two models is the expressiveness of the constructs. While the MEGA Process language provides 16 constructs (a few of them, used in our example) have been defined in Table 2), BPMN allows 12 constructs (see Table 3 for the definition of a few of them). The constructs regarding organisational modelling (Sites, Applications, organisation-units such as actors, person ...) are missing in BPMN. Some annotations (here in textual form but they may be more formal) are then added in order to add some details that will be represented, in the destination model, using specific notation (instances of the constructs “Org-Unit” and “Application”).

8. CONCLUSIONS

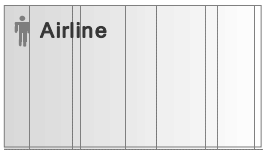
The main focus of this paper was to describe the non-technical issues in interoperating enterprise models. This paper reviewed interoperability, the types of interoperability, and managing knowledge in organizations. Then issues concerning cultural interoperability were addressed. Finally, these concepts were presented in the context of an example in the engineering change process. The syntax, language, behaviour, plus cultural issues all inhibit the interoperation of knowledge leading to true semantic interoperability. The paper then presented a practical application of how these concepts may realistically be implemented. Research and development is still necessary to put these concepts into practice in enterprise modelling tools to become more common place in the practice of all engineers.

[1] (Business Process Modelling Notation) - OMG – http://www.omg.org/bpmn
[2] MEGA Process – MEGA International – http://www.mega.com

Table 1. Process Model Terms

English	**French**
BOM – Bill Of Material	Nomenclature
ECO – Engineering Change Order	DMT - Demande de modification technique
Oracle	Oracle

Table 2. Subset of MEGA Process constructs glossary

	Org-Unit	An org-unit represents a person or group of persons participating in the business processes or information system of the enterprise. An org-unit can be internal or external to the enterprise: An org-unit is an organizational element of enterprise structure such as a management, service, or workstation. It is defined at a level depending on the degree of detail to be provided on the organization (as org-unit type). Example: financial management, sales management, marketing department, account manager. An external org-unit is an organization that exchanges flows with the enterprise, Example: Customer, Supplier, Government Office.
TicketOrder	Message	A message is information flowing within an enterprise or exchanged between the enterprise and its business environment. Messages can be information flows such as orders or invoices. For convenience, financial and material flows such as payments or product deliveries are also represented by messages
	Flow	A flow links two concepts and carries messages
	Role	A role is a participant in a collaboration, workflow, procedure or business process. It can be the initiator, that is the requester of a service, or it can represent a sub-contractor carrying out processing outside the service. A role is an integral part of the object that it describes, and is not reusable. It can subsequently be assigned to an org-unit internal or external to the organization or to an IT component. Examples: Client, Traveler.
	Application	An application is a coherent set of software tools from the software development viewpoint.
	Operation	An operation is a step in a procedure, executed by an org-unit within the context of an activity. An operation can be industrial (manufacturing a component), logistical (receiving a delivery), or can involve information processing (entering an order). An operation can be subdivided into elementary tasks.

Proposal confirmed	Condition	A condition indicates the state in which an object should be in order to perform an action.

Table 3. Subset of BPMN constructs glossary

Get Order Request	Task	A task is an atomic activity that is included within a process. A task is used when the work in the process is not broken down to a finer level of process model detail.
Proposed Itiniraries	Message	A message flow is used to show the flow of messages between two entities that are prepared to send and receive them.
Proposal confirmed	XOR Gateway	XOR -- exclusive decision and merging.
Airline	Role	A role is a participant in a collaboration, workflow, and procedure or business process. It can be the initiator, that is the requester of a service, or it can represent a sub-contractor carrying out processing outside the service. A role is an integral part of the object that it describes, and is not reusable. It can subsequently be assigned to an org-unit internal or external to the organization or to an IT component. Examples: Client, Traveller.

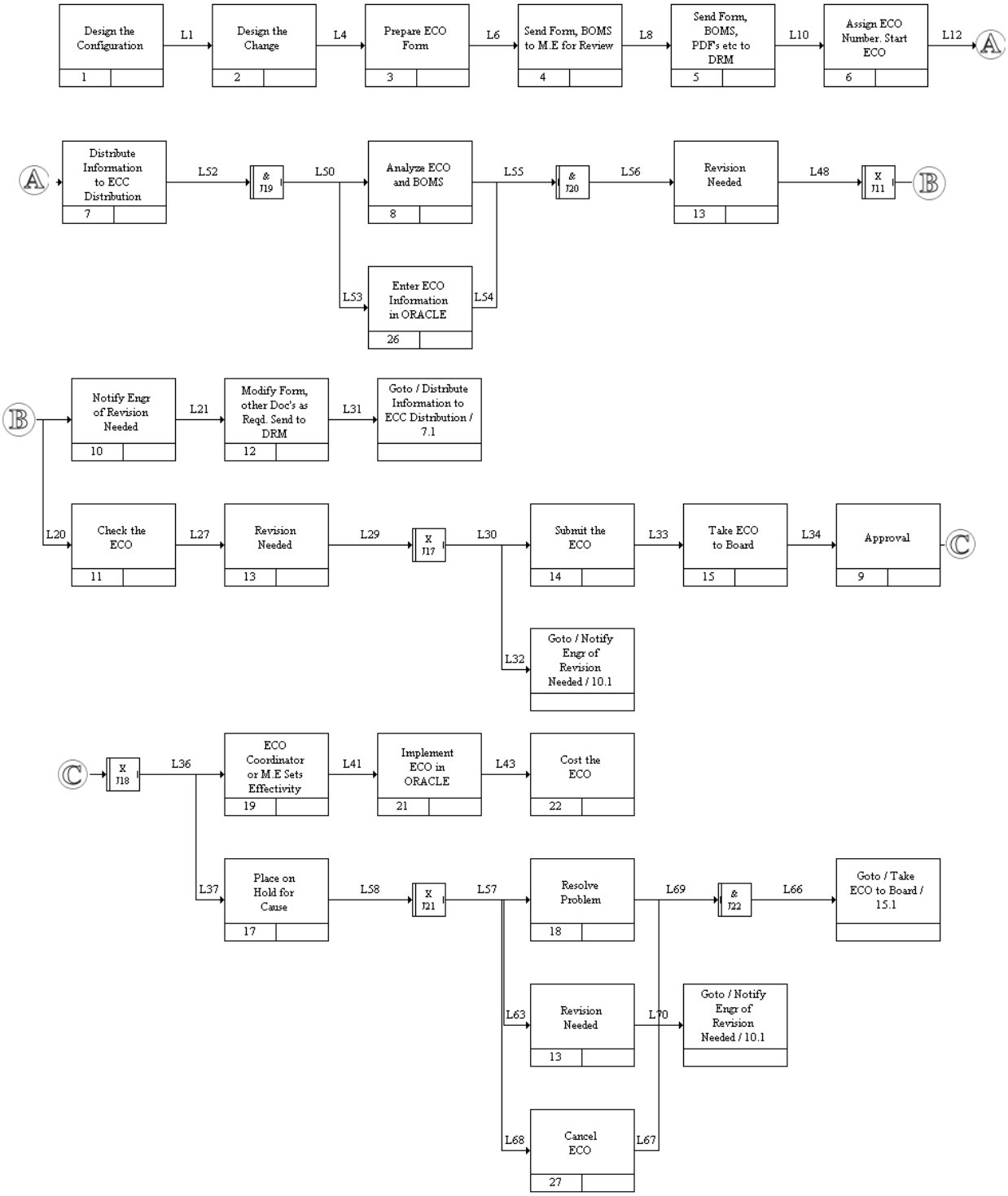


Figure 1. Process Model of Engineering Change Process

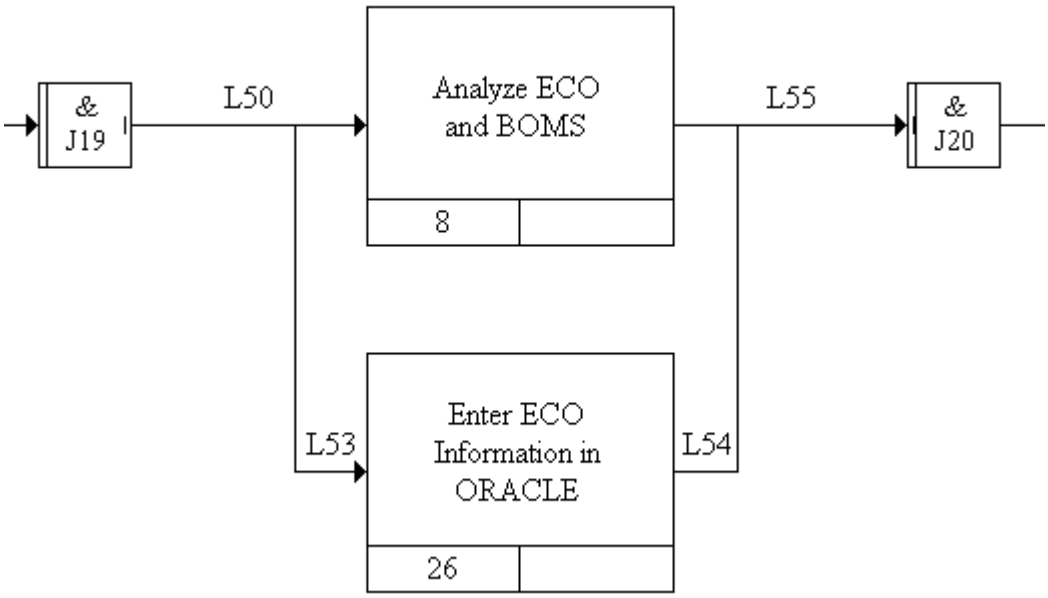


Figure 2. Section of Process Model

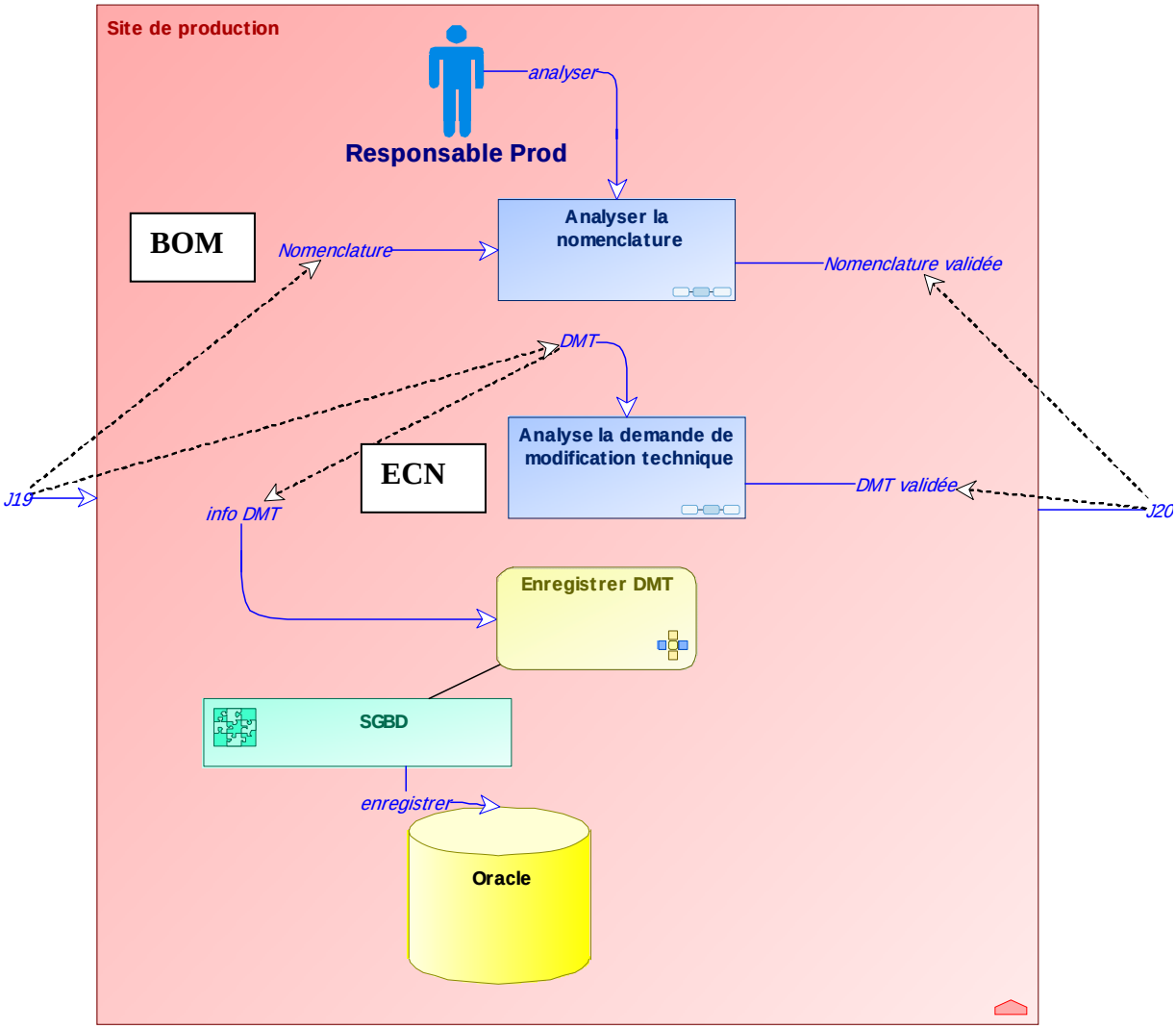


Figure 3. Section of Process Model in another Tool and another language

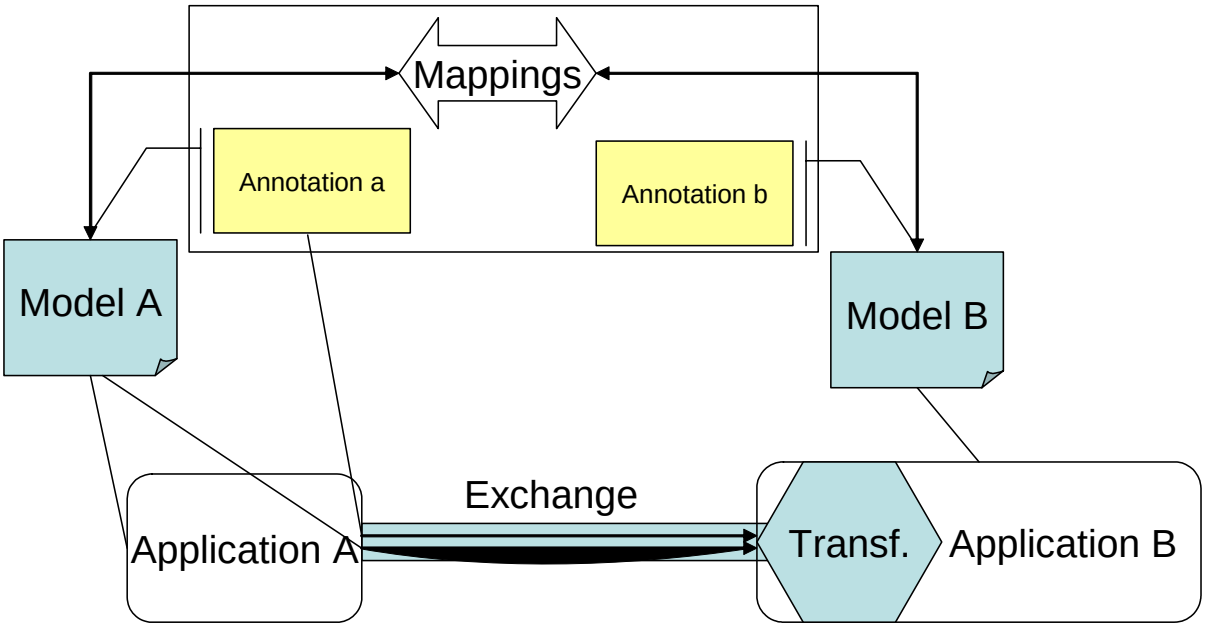


Figure 4. Model exchange and annotations (Boudjlida, *et al*, 2006)

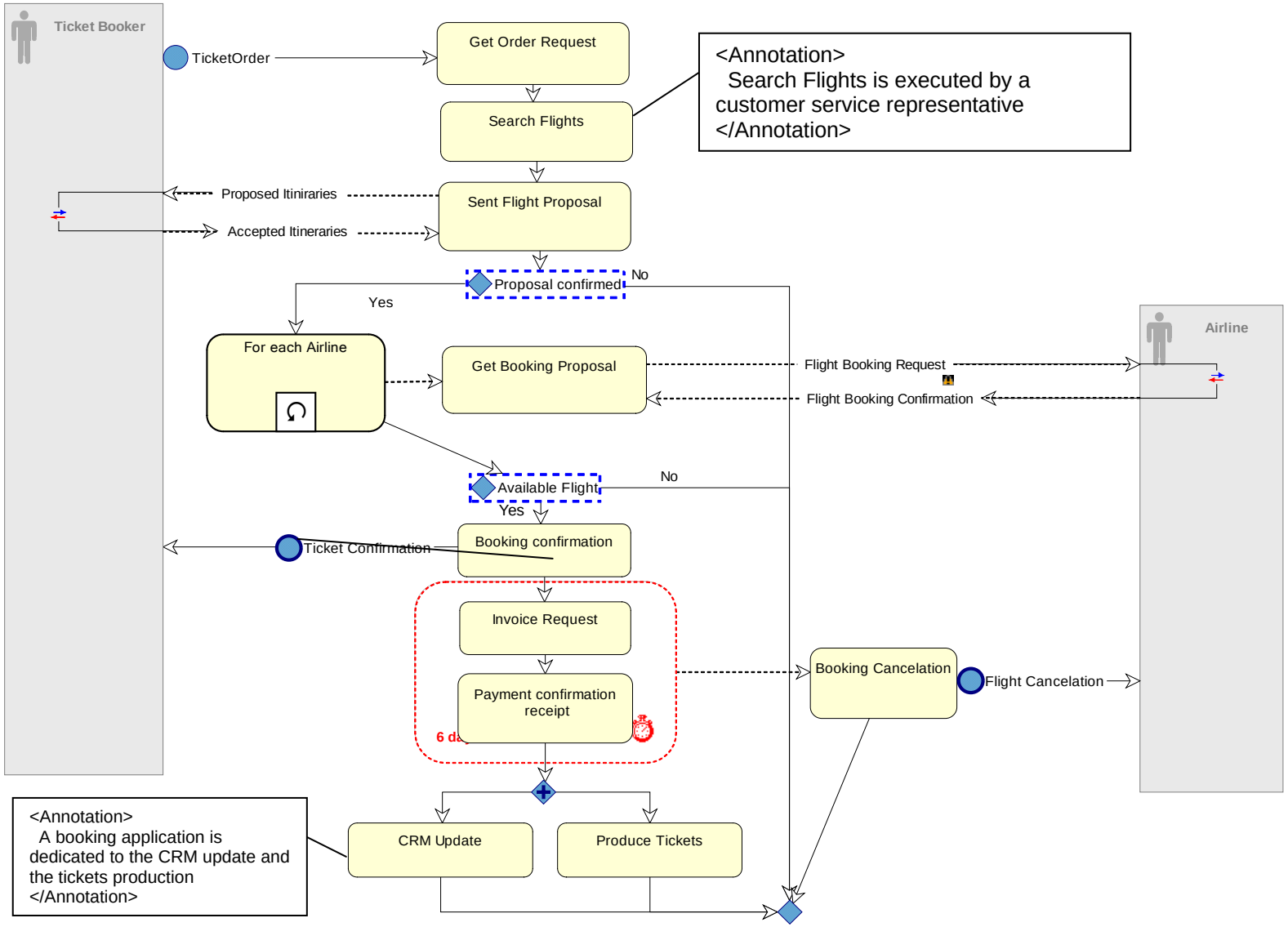


Figure 5. Ticket booking model using BPMN (Business Process Modelling Notation)

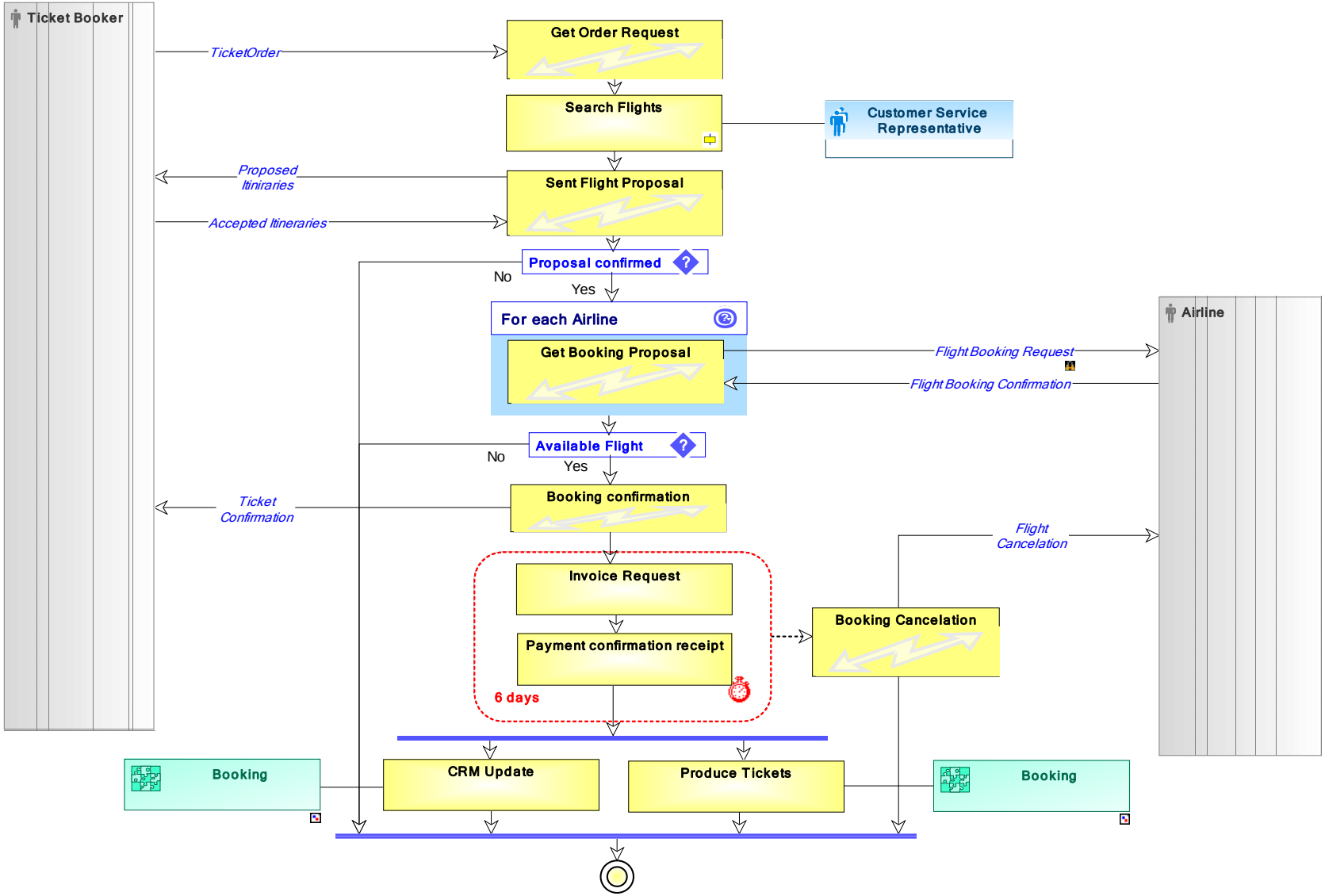


Figure 6. Ticket booking model using MEGA Process language

REFERENCES

Berio G., et al. (2002). *Requirements analysis: initial core constructs and architecture*, UEML TN IST 2001 34229. http://www.ueml.org, 2002.

Bertalanffy L. V. (1969). *General System Theory: foundations, development, applications.* George Braziller, New York.

Boudjlida N., Dong C., Baina S., Panetto H., Haussmann K., Tomas J., Abian M., Nunez M.J. (2006). Deliverable DTG4.1: A practical experiment on semantic enrichment in a homogeneous environment. In *Interoperability Research for Networked Enterprises Applications and Software Network of Excellence, n° IST* 508-011, July

Chen, D., and Vernadat, F.B. (2002). Enterprise Interoperability: A Standardization View. In *Proceedings of the IFIP International Conference on Enterprise Integration and Modelling Technology (ICEIMT'02)*, Kluwer Academics Publisher, Valencia, Spain, April 24-26 2002.

Clark, T. and R. Jones (1999). Organisational Interoperability Maturity Model for C2, In *Proceedings of the Command And Control Research And Technology Symposium (CCRTS)*, June 29 - July 1, Newport, RI, USA

deSilva, D. and deSilva, D. (1995). How to gain a competitive edge in international business through inter-cultural communication power. In: *Proceedings of Pan-Pacific Conference XII: A Business, Economic, and Technological Exchange*. May 29-June 1, 1999. Dunedin, New Zealand.

Dutta, S., T. Gulledge, and M. Tarimcilar (1997), Strategies for implementing knowledge based systems. *IEEE transaction on Engineering Management*, **44/1**, 79-91.

Gruber T. R. A (1993). Translation approach to portable ontologies. *Knowledge Acquisition*, **5(2)**, 199-220.

Harding, J., B. Yu, and K. Popplewell (1999), Information modeling: an integration of views of a manufacturing enterprise. *International Journal of Production Research*, **37**, 12, 2777-2792.

Halevy A. Y. (2001). Answering queries using views: A survey. *VLDB Journal*, **10/4**, 270–294

EIF (2004), *European Interoperability Framework for pan-European eGovernment Services*, Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens (IDABC), November, Luxembourg

IEEE (1990). *IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries*. Institute of Electrical and Electronics Engineers. New York, NY: 1990.

IEC 62264 (2002). IEC 62264-1:2002. *Enterprise-control system integration, Part 1. Models and terminology*, ISO/IEC, Geneva

INTEROP NoE (2003). Interoperability Research for Networked Enterprises Applications and Software. Technical annex (restricted) IST FP6 Network of Excellence IST 2003-508011, November, http://www.interop-noe.org

ISO 14528 (1999). *Industrial Automation Systems – Concepts and rules for Enterprise Models*, TC 184/SC5/WG1, Geneva, Switzerland

ISO 16100 (2002). *Manufacturing Software Capability Profiling for interoperability, Part 1: Framework*, TC 184/SC5/WG4, Geneva, Switzerland, ICS 25.040.01

ISO EN DIS 19440 (2004). *Enterprise integration - Constructs of enterprise modelling*, Draft version, TC 184/SC5/WG1, Geneva, Switzerland

Koen, B. (2003). *Discussion of the Method: Conducting the Engineer's Approach to Problem Solving.* Oxford University Press. USA.

Lilleng, J. (2005). Towards Semantic Interoperability. *Proceedings of the IFIP/ACM SIGAPP INTEROP-ESA conference*, February 23-25, Geneva, Switzerland, Springer Science publisher.

Molina, A., Chen, D., Panetto, H., Vernadat, F., & Whitman, L. (2004). Enterprise integration and networking: issues, trends and vision. In *Proceedings of the IFIP International Conference on Enterprise Integration and Modeling Technology (ICEIMT'04), Knowledge Sharing in the Integrated Enterprise Interoperability - Strategies for the Enterprise Architect*, **183**, 303-313, Springer publisher, October 9th-11th, Toronto, Canada.

Molina, A., Aca, J., and Wright, P. (2005). Global collaborative engineering environment for integrated product development. In *International Journal of Computer Integrated Manufacturing* **18/8**, 635-651.

OASIS, (2002). Organization for the Advancement of Structured Information Standards. ebXML specification, February, http://www.oasis-open.org

Panetto H., Berio G., Benali K., Boudjlida N. and Petit M. (2004a). A Unified Enterprise Modelling Language for enhanced interoperability of Enterprise Models. *In Proceedings of IFAC INCOM 2004 Symposium, April 5th-7th, Bahia, Brazil*.

Panetto, H., Whitman, L.E. and Chatha, K.A. (2004b). Ontology for Enterprise Interoperability in the domain of Enterprise Business Modelling. In: *IFAC TC 5.3 International workshop on Enterprise Interoperability, Integration and Networking (EI2N'2005), Proceedings of the Worlshops of the IFIP/ACM SIGAPP INTEROP-ESA conference*, February 22nd, Geneva, Switzerland, 103-114, Hermès Science publishing, ISBN 1-903209-49-5

Petit M, and al. (2002). *State of the Art in Enterprise Modelling*, UEML TN IST 2001 34229, www.ueml.org

Tsagkani, C. (2005) Inter-Organizational Collaboration on the Process Layer. In *Proceedings of the IFIP/ACM SIGAPP INTEROP-ESA conference*, February 23-25,

Geneva, Switzerland, Springer Science publisher, ISBN: 1-84628-151-2

UEML. (2003). Unified Enterprise Modelling Language (UEML) Thematic Network. IST-2001-34229, http://www.ueml.org.

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

Vernadat, F.B. (2000). UEML: towards a unified enterprise modeling language. *International Journal Production Research*, **40**/**17**, 4309-4321.

W3C (2003). *Simple Object Access Protocol (SOAP) 1.2*, W3C specification, February, http://www.w3c.org

W3C (2004a), *XML, Extensible Mark-up Language*, W3C XML 1.1 recommendation, February, http://www.w3c.org

W3C (2004b). *Web Services Description Language (WSDL) 2.0*, W3C specification. August, http://www.w3c.org