Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Preparation of Papers for IFAC Conferences & Symposia: Ontology-based Methodology for Collaborative Process Definition of Enterprise Networks

V. Rajsiri*. JP. Lorré.*
F. Bénaben. H. Pingaud****

**EBM WebSourcing, Ramonville St-Agne, 31520*
France (Tel: 33-(0)5-61-28-56-11; e-mail: netty.rajsiri, jean-pierre.lorre@ebmwebsourcing.com)
***Ecole des Mines d'Albi-Carmaux, Centre de Génie Industriel, Albi, 81000*
France (e-mail: benaben, pingaud@enstimac.fr)

Abstract: This paper presents a knowledge-based methodology dedicated to automate the specification of virtual organization collaborative processes. Our approach takes as input knowledge concerning collaboration coming from involved organizations and produces as output a BPMN compliant process. The collaborative network ontology consists of (i) collaboration attributes, (ii) description of participants and (iii) collaborative processes inspired from the enterprise Process Handbook (MIT). This OWL ontology coupled with a reasoning engine will be used by a collaboration aided design tool (CDT) provided by EBM WebSourcing.

1. INTRODUCTION

Nowadays companies tend to open themselves to their partners and enter in one or more networks in order to have access to a broader range of market opportunities. The heterogeneities of partners (e.g. location, language, culture, information system), the long-term relationships and establishing mutual trust between its partners are the ideal context for the creation of collaborative networks. The interoperability is a possible way toward the facilitation of integrating networks (Konstantas et al., 2005) (Vernadat, 2006).

General issue of each company in collaboration is to establish connections with their partners. Partners have no idea about what their collaboration will exactly be but they know what they are waiting for from the collaboration. This means that partners can express informally and partially their collaboration requirements (knowledge). But, how to make these requirements more formalized and completed?

In principle, partners collaborate through their information system. The concept of collaborative information system (CIS) has been evolved to deal with the interoperability issues. According to (Touzi et al., 2006), this concept focuses on combining the information systems of different partners into a unique system.

Developing such a CIS concerns the transformation of a BPMN collaborative process model into a SOA (Service Oriented Architecture) model of the CIS. This is based on the Model Driven Architecture (MDA) approach (Millet et al., 2003), as discussed in (Touzi et al., 2007). The BPMN supports the Computation Independent Model (CIM) of the MDA, while the SOA-based CIS supports the Platform Independent Model (PIM) of the MDA.

Consequently, our research interest concerns the CIM model. The main focus is to formalize the informal and partial knowledge expressed from the partners in form of BPMN relevant process. However, how do we obtain the BPMN? The answer is shown as follows:

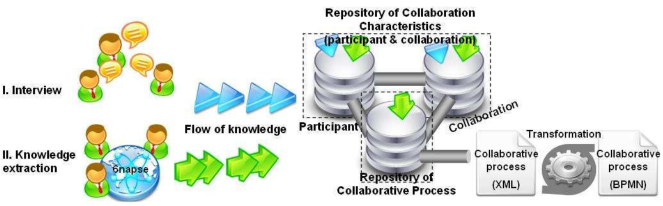

Fig. 1. Our approach for defining a BPMN

The schema above shows our approach composing of (i) two gathering methods: interview and knowledge extraction, (ii) two repositories: collaboration characteristics (participant and collaboration) and collaborative processes, and (iii) a transformation.

The approach starts at gathering knowledge by interviewing or extracting from a platform called 6napse. This knowledge will be classified and kept in corresponding repositories. Main difference between these two gathering methods is that the interview provides knowledge about the participants (e.g., name, role, business, service) and their collaborations (e.g., relationship, common objective) for the characteristic repository, while the extraction from 6napse provides not only the same knowledge as interview, but also the collaborative process (e.g., CIS, CIS services). Both repositories allow to analyze, keep and construct knowledge in form of collaborative process.

Defining these two repositories requires implementing a knowledge-based methodology. This methodology uses ontology and reasoning to automate the specification of collaborative processes. The ontology covers the collaborative network domain which maintains the repositories of collaboration characteristics and collaborative processes, as shown in Fig.1. The reasoning methodology establishes the interactions between the repositories in order to fulfill the building of collaborative processes.

The paper is focused firstly on introducing the collaborative network ontology and the reasoning methodology. Secondly, an application of the knowledge-based methodology will be presented by using the interview gathering method.

2. KNOWLEDGE-BASED METHODOLOGY

In Artificial Intelligence, according to (Grimm et al., 2007), knowledge representation and reasoning aim at designing computer systems that reason about a machine-interpretable representation of the world, similar to human reasoning. A knowledge-based system maintains a knowledge base which stores the symbols of the computational model in form of statements about the domain, and it performs reasoning by manipulating these symbols.

Our knowledge-based methodology lies on the above approach in order to deal with the collaborative process design.

In principle, the methodology of collaborative process design in our case starts at analyzing the input knowledge regarding collaboration requirements of the participants and ends at providing a related BPMN collaborative process. After manipulating the methodology, what we are waiting for at the output are network participants, exchanged data, business services and coordination services. These elements are essential for designing BPMN collaborative process. Thus, to make the methodology able to produce these elements, we need (i) to define an ontology and rules describing the collaborative network domain and (ii) to use an inference engine to deduce these modelling elements from the input knowledge.

In this section, we will present the ontology of collaborative network. Then, the reasoning methodology will be presented step by step.

2.1 Collaborative Network Ontology (CNO)

An ontology is a specification of a conceptualization (Gruber, 1993). It contains a set of concepts relevant in a given domain, their definitions and inter-relationships.

To define domain and scope of an ontology, (Natalya et al., 2001) suggested starting by answering several basic questions which concerns for example, the domain of interest, user and expected result of the ontology. Often developing an ontology is akin to defining a set of data and their structure for programs to use. Problem-solving methods and domain-independent applications use ontologies and knowledge bases built from ontologies as data.

The domain of interest for developing an ontology that we focus is on the collaborative network domain especially for designing collaborative process. The knowledge base built from this ontology will cover the two repositories shown in Fig.1. It will be used in some applications by the consultants of EBM WebSourcing to suggest their clients a collaborative process relevant in a given collaboration behaviours.

There are three key concepts underlying the collaborative network ontology (CNO) which are (i) the participant concept, (ii) the collaboration concept and (iii) the collaborative process concept.

What we need to define in an ontology is not only the concepts, relations and properties, but we need also to define rules that reflect the notion of consequence. The followings are some examples of rules in the collaboration domain: If decision-making power is equal and duration is discontinuous then topology is peer-to-peer or if role is seller then participant provides delivering goods.

The following paragraphs describe these three concepts with their relations, properties and rules.

The **participant concept**, see Fig. 2, interests in the descriptions about participant. It concerns the characterization criteria of collaboration (Rajsiri et al., 2007a).

A participant provides several services at high level (discussed in the collaborative process concept) and resources (e.g., machine, container, technology), plays proper roles (e.g., seller, buyer, producer) and has business sectors (e.g., construction, industry, logistic).

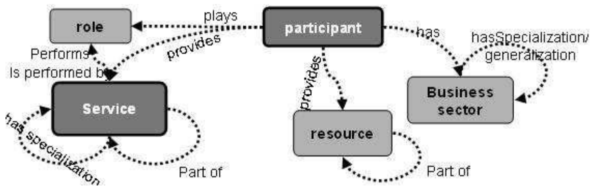

Fig. 2. RDF graph representing the participant concept

From the above figure, reasoning by deduction can be occurred for example between role and service. The related services will be derived from a given role and vice-versa. For example, if role is computer maker then its services are making screen, making keyboard…

The **collaboration concept**, see Fig. 3, concerns the characterization criteria of collaboration (Rajsiri et al., 2007a) and also integrates the collaborative process meta-model (Touzi et al., 2007). Common objective, resource, relationship and topology are the characterization criteria, while CIS and CIS services are a part of the collaborative process meta-model.

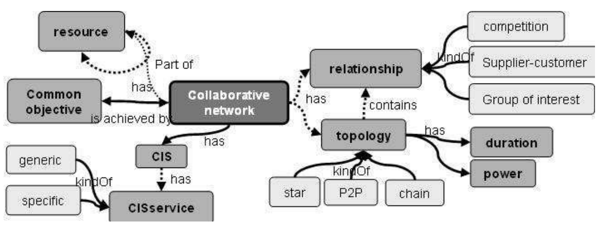

Fig. 3. RDF graph representing the collaboration attributes

A collaborative network has a common objective (e.g., group same products to buy together) and a CIS. A CIS has its own CIS services which can be generic (e.g., send documents/mails) or specific (e.g., select supplier service). A network can have several topologies which can be star, peer-to-peer, chain or combination of these three structures. Topology has duration and decision-making power characteristics. Central, equal or hierarchic making power is a decision-making power. Duration can be continuous or discontinuous. A topology contains relationships which can be group of interest, supplier/customer or competition.

To deduce topology, we define some rules, for example, if decision-making power is equal and duration is discontinuous then topology is peer-to-peer.

The **collaborative process concept**, see Fig. 4, is an extension of the concepts developed by the MIT Process Handbook project (Malone et al., 1999).

Services (e.g., computer manufacturing, software development) express competencies of participant at high level. A service can be divided into business service and coordination service. Business service (e.g., assemble components of computer) explains task at functional level. Service can deduce the business services that correspond to it. For example, if service is making keyboard then business services are assembling circuit board, testing board…

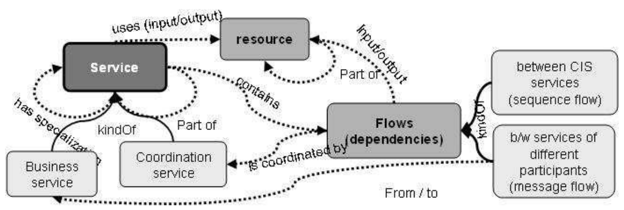

Fig. 4. RDF graph representing the service

The concept of dependencies (flows) of resources between two services is also included. Each dependency can be associated a coordination service (e.g., manage flow of material from a business service to another). This means that a coordination service is in charge of managing a dependency. For example, we can use the forwarding document coordination service to manage a dependency containing bill from service A to service B.

To get the three above concepts together, we express as shown in Fig.5.

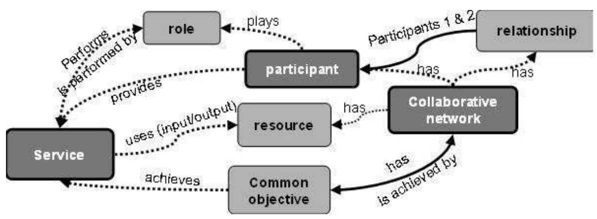

Fig. 5. Union of the participant, collaboration and collaborative process concepts.

Collaborative networks usually have several participants, resources, relationships and a common objective. Common objective achieves services which use resources. Services are mostly performed by roles of the participants. A relationship gets two participants together which its type is depended on the roles of the participants (e.g. if two participants play seller and buyer roles, the relationship will be supplier/customer).

Once the CNO has been informally defined, we need to formalize it with rigorous syntax and semantic language. OWL (Web Ontology Language), a W3C recommendation, is the most recent development in standard ontology languages. There are three OWL versions but the most appropriate one in our case is OWL-DL (Description Logics) because it adapts to automated reasoning. OWL-DL guarantees the completeness of reasoning (all the inferences are calculable) and logics.

For using this language, we need an editor to create ontology's elements (classes, relations, individuals and rules). We use the Protege which is an open-source OWL editor developed by Stanford University (Protege, 2000).

2.2 Seven-Step Reasoning Methodology

Once the ontology has been edited in Protege via OWL-DL, we will reason on the ontology. To do this, we use the inference engine Pellet. Pellet is an OWL-DL inference engine open source, originally developed at the University of Maryland's Mindswap Lab (Sirin et al., 2007).

The ontology coupled with an inference engine will be used by the collaboration aided design tool (CDT). The CDT will be used as the application by the consultants of EBM WebSourcing to illustrate network and collaborative process to their clients. The idea is to show the progress while creating collaborative network and collaborative process at the end of each step of the reasoning methodology.

Before starting the reasoning methodology, an information gathering is required to be done. The information we would like to gather can be divided into two parts: (i) characteristics of network and (ii) participants' details which concern their skills and collaboration details concerning their roles and services in the network context. The questions are, for example, identify relationship between the participants, define the type of decision-making power and duration (continuous or not) for each relationship, describe services (or tasks) or roles which must have in the network, etc.

Once the CNO and the CDT are ready to be used as well as the required information is gathered, the queries are also needed to be defined and executed in order to extract data of the ontology. So, we use SPARQL which is developed primarily to query RDF graphs (Sirin and Parsia, 2007).

The querying and reasoning methodology consists of seven steps starting from characterizing collaboration until defining modelling elements one by one. The following paragraphs will be described the seven-step reasoning methodology with some examples.

Step 0 - Characteristic and topology: the characteristics of a collaborative network come normally when gathering information from the participants by interview.

The topology will be derived from duration and decision-making power, which are also provided by participants. For instance, if decision-making power is equal and duration is discontinuous then topology is peer-to-peer, if decision-making power is hierarchic for whatever duration be then topology is chain.

Step 1 – Business, role, service and business service of each participant: this step is dedicated to complete details of the participants concerning their businesses, roles, services and business services. The knowledge obtained while gathering provides information about business. Role and service are not compelled but at least one of them is required to be provided because they can be completed by each other by deduction. It means that related services will be derived from a given role and vice-versa. For example, if role is computer maker then services are making screen, making keyboard…

Once the services have been defined, what we have to do next is deducing all business services that correspond to the services. For example, if service is making keyboard then business services are assembling circuit board, testing board…

Step 2 - Dependency between two business services belonged to different participants: the key concept we are using is the dependency concept by considering possible combinations of services using resources (Crowston, 1994).

Once the business services have been defined to the related services for each participant, we query here all possible dependencies by filtering only services which their output resource is the same as the input used by other services when services are owned by different participants. For example, if the placing order service of a buyer produces a purchase order as output and the obtaining order service of a seller uses a purchase order as input then there is a dependency of resource between these two services.

Step 3 - Coordination and CIS services: the key concept we are using is the notion from coordination theory of (Crowston, 1994). This concept deals with the organizational view of collaborative process. The concepts of dependency and coordination are related because coordination is seen as a response to problems caused by dependencies. Or we can say that coordination can be defined as managing dependencies among services.

At this step, we search in the knowledge base the available coordination services which match to the dependencies found in Step 2. For example, the dependency for transferring raw materials from a supplier to a customer, the coordination service can be managing physical goods accessibility. If there are any dependencies which we cannot find an appropriated coordination service, we need to create a new one and stock it in the knowledge base.

Once we have got possible coordination services for every dependency, according to the definition of CIS service defined in (Touzi et al., 2007), we can say that a coordination service is equivalent to a CIS service.

Step 4 - Added-value service of CIS: this step is focused on completing collaborative process by verifying if all required services (based on the common objective of network) have someone to be in charge of. The key concept we are using here is the value chain (Porter, 1985). The value concept provides a list of services describing competencies at very high level and generic (e.g., vehicle manufacturing, software development).

Firstly, we have to find services that would be achieved by the common objective of network. These will be all required services to be included in network. For example, if the common objective of a network is making computers to stock then the required services are buying computer parts, inbound logistics, assembling parts and warehousing. Secondly, we have to find if there are any services that no participant can perform. To do this, we have to compare all required services to the result of Step 1. If the participants can perform all required services, we can go to Step 5. If not, the services which no any participants can perform will belong to the CIS. Then, we need to query from the knowledge base to find whether there are any available business services which concern these services or not. If so, the consultants of EBM WebSourcing will select the suitable ones with the participants and add as CIS services. If not, we need to specify new business services.

Step 5 - Dependency between two CIS services: this step use the same concept as Step 2 but the services we focus here belong to the CIS.

Step 6 - Coordination and CIS services: this step use the same concept as Step 3 but the services we focus here belong to the CIS.

In this way, the ontology can reason by means of automated deduction about the collaboration domain, similar to the way a human would. The application of the reasoning methodology will be presented in Section 3 with a scenario.

3. EXAMPLE SCENARIO

To illustrate principles of knowledge representation in this section, we introduce an example scenario taken from a customer-supplier use case.

In the scenario, we will start by describing the collaborative situation which is our input knowledge. Then, the application of the methodology will be presented step by step and also shown its graphic result from the CDT.

3.1 Input Knowledge

In this scenario, we have three participants which their role is manufacturer (M), supplier (S) and warehouseman (W). They perform a supply chain network for manufacturing to stock. The business sectors are manufacturing and logistics. The relationships between participants are identified pair by pair as shown in the table 1. The information about decision-making power and duration are defined once the relationships have been identified, see the table 1. The attributes of relationship is shown in the figure 2 and of decision-making power and duration are shown in the figure 3.

Table 1. Type of relationship (R), decision-making power (P) and duration (D)

Participants	M	S	W
M		(R):Supplier-customer (P):Hierarchic (D):Continuous	(R):Supplier-customer (P):Hierarchic (D):Continuous
S	idem		N/A
W	idem	N/A	

From this knowledge, we can draw the network like this:

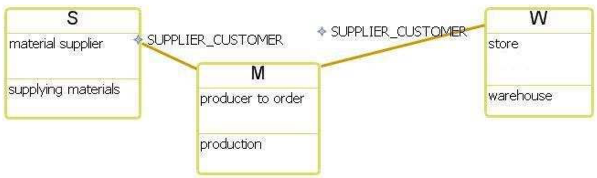

Fig. 5. Network of M, S and W drawn by the CDT.

3.2 Application and Results from Reasoning by Deduction at each step

Step 0: Characteristic and topology of network

From the information in Table 1, two topologies are supposed to be derived because there are two relationships (M-S and M-W). This network has two chain topologies because, from the deduction, the decision-making power is hierarchic and the duration is continuous.

Step 1: Business, role, service and business service of each participant

Business and role of each participant is provided as the input knowledge. Thus, in this use case, the related services are derived from the given roles. After that, we reason possible business services related to the derived services. For example, since S plays supplier role, so S provides material supplying service which has obtaining order, scheduling delivery, transferring material and receiving payment business services. The figure 9 shows this information:

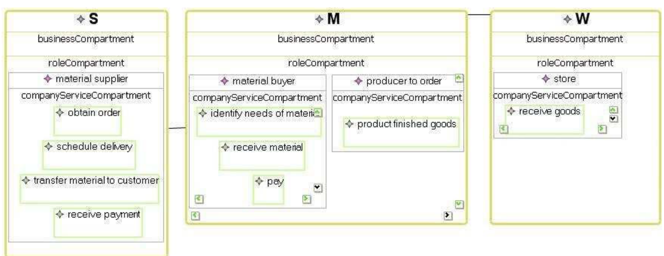

Fig. 6. Participants with their roles and business services.

Step 2: Dependencies between business services of two different participants

Once the business services provided by the participants have been reasoned, all possible dependencies will be filtered by checking if the input of a business service is the same as the output of another business service when participants belonging to these services are different. For example, since the identifying needs of materials of M produces a purchase order which is required as input of the obtaining orders of S, there is a dependency of purchase order. The dependencies we have got here are shown below:

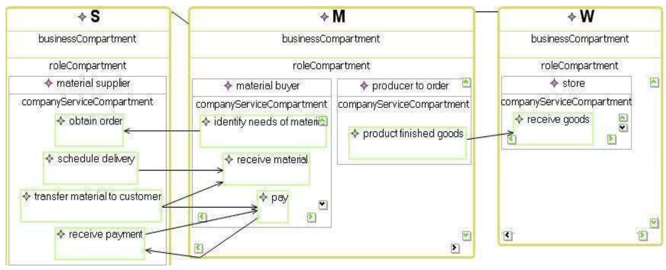

Fig. 7. Dependencies between business services

Step 3: Coordination services and CIS services

For every dependency found in the previous step, it needs to have at least a coordination service to manage it. For example, to manage the dependency containing bill from the transferring materials business service of S to the paying business service of M, we can use the forwarding document coordination service, a generic CIS service. The figure below shows this example:

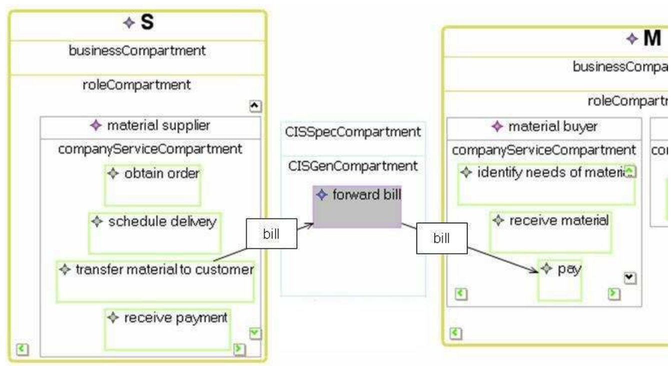

Fig. 8. An example of coordination service

Step 4: Added-value services of CIS

First of all, we need to find all services that the common objective has to achieve. Since the common objective in this case is to fulfil their supply chain for manufacturing products to keep in stock. The services required to have in the network are procurement, inbound logistics, operations, sales, outbound logistics and control and evaluation. Secondly, we need to find the services that the participants can perform. In this case, these are what the participants can do: procurement, inbound logistics, operations and sales. In comparison of what we have got (services required to achieve the common objective vs. services done by the participants), we have found the control and evaluation service that no any participants can do. Thus, the control and evaluation service will belong to the CIS.

The consultant of EBM WebSourcing found then from the knowledge base any available business services which concern the control and evaluation service. The selected business services are such as controlling payment delay and production plan… These will be created as the CIS specific services.

Steps 5 & 6: Dependencies between CIS services and Coordination services

We repeat the steps 2 and 3 once again to derive all possible dependencies between the CIS services and the relevant coordination services. At the end, the result is shown below:

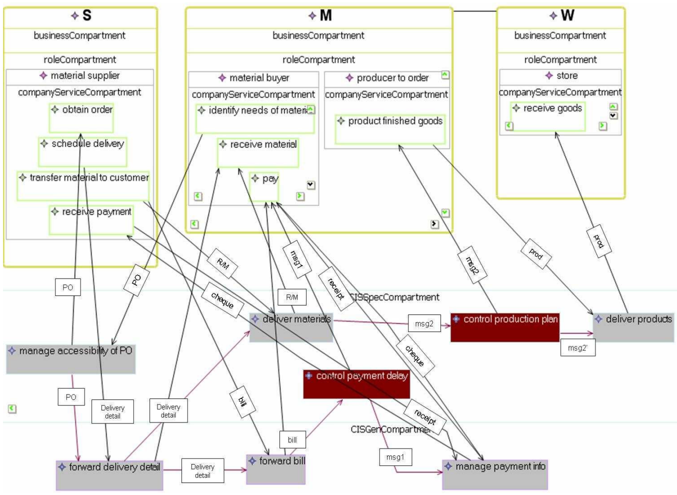

Fig. 9. A solution of collaborative process of the network.

We have to remark that the final collaborative process obtained at the end is just a solution for the given use case. It is always possible to have other solutions that correspond to collaborative behaviours of the participants more or less than the proposed one.

4. CONCLUSIONS AND ON-GOING RESEARCH

Since the output of the knowledge-based methodology is a collaborative process illustrated by the CDT, the obtained collaborative processes have to be transformed into a BPMN compliant process. As such a collaborative process model is an input of the CIS translator. Thus, the meta-model of collaborative process (Touzi et al., 2007) is required for accomplishing this transformation.

However, while conceptualizing the CNO, we have already integrated the meta-model of collaborative process. Thus, the collaborative process obtained from the knowledge-based methodology (Fig. 9) is really near the BPMN compliant process but still not complete. There are some missing elements such as gateways and events. These elements are needed to be added in actual collaborative processes because they can make process more dynamic.

Our current work is focused on adding the dynamic aspect to the actual reasoning methodology by taking into account event and gateway elements. Also, the actual knowledge-based methodology, including its concepts, rules and reasoning steps is needed to be finalized and validated. After that, we will handle the transformation of collaborative processes obtained from the methodology into BPMN collaborative processes.

REFERENCES

Crowston, K. (1994). A Taxonomy of Organizational Dependencies and Coordination Mechanisms (Working paper No. 3718-94): Massachusetts Institute of Technology, Sloan School of Management.

Grimm S., Hitzler P. and Abecker A. (2007). Knowledge Representation and Ontologies: Logic, Ontologies and Semantic Web Language, *Semantic Web Services*. Springer.

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

Konstantas D, Bourrieres JP, Léonard M and Boudjlida N. (2005). Interoperability of enterprise software and applications, INTEROP-ESA'05, Geneva Switzerland, Springer-Verlag.

Malone, T.W., Crowston K., Lee J., Pentland B. (1999) Tools for inventing organizations : Toward a Handbook of Organizational Processes, Management Science/Vol. 45, No. 3.; Process Handbook Online : http://ccs.mit.edu/ph/

Millet J. and Mukerji J. (2003). MDA Guide Version 1.0.1, available on http://www.omg.org

Natalya F.N. and Deborah L.M. (2001). Ontology Development 101: A Guide to Creating Your First Ontology. Stanford University, Stanford, CA, USA.

Protege (2000). http://protege.stanford.edu

Porter M. (1986). L'avantage concurrentiel, InterEdition, Paris, page 52.

Rajsiri V, Lorre JP, Benaben F and Pingaud H. (2007a). Cartography for designing collaborative processes. Interoperability of Enterprise Software and Applications

Rajsiri V, Lorre JP, Benaben F and Pingaud H. (2007b). Cartography based methodology for collaborative process definition, Establishing the foundation of collaborative networks. Springer. Pages 479-486.

Sirin E. and Parsia B. (2007). SPARQL-DL: SPARQL Query for OWL-DL. 3rd OWL Experiences and Directions Workshop (OWLED-2007).

Sirin E., Parsia B., Grau B.C., Kalyanpur A. and Katz Y. (2007). Pellet: A practical OWL-DL reasoner**,** Journal of Web Semantics, 5(2).

Touzi J, Lorre JP, Benaben F and Pingaud H. (2006). Interoperability through model based generation: the case of the Collaborative IS. Enterprise Interoperability.

Touzi J, Benaben F, Lorre JP and Pingaud H. (2007). A Service Oriented Architecture approach for collaborative information system design, IESM'07, Beijing China.

Vernadat FB. (2006). Interoperable enterprise systems: architectures and methods, INCOM'06 Conference, St-Etienne, France.