2002s-64

An Evaluation of Inter-Organizational Workflow Modelling Formalisms

Aymeric Dussart, Benoit A. Aubert and Michel Patry

Série Scientifique
Scientific Series

Montréal
Juillet 2002

CIRANO
Centre interuniversitaire de recherche en analyse des organisations

CIRANO

Le CIRANO est un organisme sans but lucratif constitué en vertu de la Loi des compagnies du Québec. Le financement de son infrastructure et de ses activités de recherche provient des cotisations de ses organisations-membres, d'une subvention d'infrastructure du ministère de la Recherche, de la Science et de la Technologie, de même que des subventions et mandats obtenus par ses équipes de recherche.

CIRANO is a private non-profit organization incorporated under the Québec Companies Act. Its infrastructure and research activities are funded through fees paid by member organizations, an infrastructure grant from the Ministère de la Recherche, de la Science et de la Technologie, and grants and research mandates obtained by its research teams.

Les organisations-partenaires / The Partner Organizations

- •École des Hautes Études Commerciales
- •École Polytechnique de Montréal
- •Université Concordia
- •Université de Montréal
- •Université du Québec à Montréal
- •Université Laval
- •Université McGill
- •Ministère des Finances du Québec
- •MRST
- •Alcan inc.
- •AXA Canada
- •Banque du Canada
- •Banque Laurentienne du Canada
- •Banque Nationale du Canada
- •Banque Royale du Canada
- •Bell Canada
- •Bombardier
- •Bourse de Montréal
- •Développement des ressources humaines Canada (DRHC)
- •Fédération des caisses Desjardins du Québec
- •Hydro-Québec
- •Industrie Canada
- •Pratt & Whitney Canada Inc.
- •Raymond Chabot Grant Thornton
- •Ville de Montréal

ISSN 1198-8177

An Evaluation of Inter-Organizational Workflow Modelling Formalisms

Aymeric Dussart[†], *Benoit A. Aubert*[*] *and Michel Patry*[‡]

Résumé / ***Abstract***

Ce travail consiste en une évaluation des aspects dynamiques du language UML dans un contexte de workflow inter-organisationnel. Le choix du language par rapport à d'autres est motivé par sa richesse grammaticale lui offrant une très bonne adaptation à ce contexte. L'évaluation se fait par une validation ontologique basée sur les modèles BWW (Bunge-Wand-Weber) et par la réalisation d'un prototype de système de gestion de workflows inter-organisationnels. À partir des résultats convergents obtenus des deux différentes analyses, des améliorations au formalisme UML sont suggérées. D'un autre coté, les analyses divergentes suggèrent une possibilité de spécifier les modèles BWW à des contextes plus particuliers tels que ceux des workflows et permettent également de suggérer d'autres améliorations possibles au langage.

This paper evaluates the dynamic aspects of the UML in the context of inter-organizational workflows. Two evaluation methodologies are used. The first one is ontological and is based on the BWW (Bunge-Wand-Weber) models. The second validation is based on prototyping and consists in the development of a workflow management system in the aerospace industry. Both convergent and divergent results are found from the two validations. Possible enhancements to the UML formalism are suggested from the convergent results. On the other hand, the divergent results suggest the need for a contextual specification in the BWW models.

Mots clés : Ontologie, étude conceptuelle, validation du prototype, UML, méthodes et outils de développement IS.

Keywords: Ontology, Conceptual study, Prototype Validation, UML, IS development methods and tools.

† Robichaud Conseil and CIRANO, a.dussart@robichaudconseil.com.
* Associate Professor, HEC Montreal and Fellow, CIRANO, Benoit.aubert@hec.ca). Main contact author: HEC Montreal, 3000 chemin Côte Ste-Catherine, Montreal, (Canada) H3T 2A7. Telephone: 514.340.7307, Fax: 514.340.6132.
‡ Professor, HEC Montreal and Fellow, CIRANO, michel.patry@hec.ca.

Introduction

Transactions have been traditionally managed either through organizations or through markets. With advances in electronic commerce and in information systems, this distinction is getting blurred. For example, the last years have seen the development of electronic intermediaries, also known as electronic marketplaces (e-marketplaces), which aim at concentrating transactions made within, or across, industrial sectors through a limited number of virtual intermediaries. These virtual markets enhance transactional efficiency through the aggregation of trading partners (Lucking-Reiley and Spulber, 2001) and through a reduction in asymmetrical information. Both means have the potential to substantially reduce transaction costs (Garicano and Kaplan, 2000).

It is clear that electronic business has penetrated business to business (B2B) processes and consequently spurred a transformation of the traditional organizational boundaries (Zwass, 1998). Since technology has made possible the participation of several partners in shared business processes , these have been crossing organizational boundaries to an extent never experienced before (Van der Aalst, 2000).

Research on inter-organizational workflow technology is facing an important problem. It has essentially focused on technical issues and has almost ignored language structure (Van der Aalst, 2000). This is a classical case of a "technology seducer" problem, very present in the Information Systems (IS) discipline which has been criticized by Weber (1997).

This paper assesses the adequacy of representation languages for inter-organizational business processes. There is no question that having adequate language structures for representation is a fundamental requirement for adequate development. The evaluation methodology is based on ontology, using Wand and Weber models (1990), and prototyping. Since little empirical validation work has been done on Wand and Weber's models, this analysis will be combined with a prototypical validation that will consist in comparing the process language used in a workflow management system to the process language used for modeling business processes. By combining the two approaches, convergent results are expected to be found to validate the language.

The paper is organized as follows. First, workflows are defined. Then, a literature review is presented to introduce our ontological evaluation framework and to select a candidate language for thorough ontological and prototypical evaluation. Then, the ontological and the prototypical validation are developed. A discussion of the results follows using the convergent and divergent elements from both validations.

Definition of Workflows

With leading e-business software vendors such as IBM, BEA systems, Oracle, Vignette.com, and Microsoft (with Biztalk Server) offering workflow solutions, workflow technology can no longer be purely considered as hype. There are over 200 products available today (Van der Aalst, 2000). A workflow can be defined as: "*The computerized facilitation or automation of a business process, in whole or in part*" (WfMC, 1995, p. 6) and a Workflow Management System (WFMS) as: "*A system that defines, creates and manages the execution of workflows through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with workflow participants and, where required, invoke the use of IT tools and applications*" (WfMC, 1995, p. 6).

Originally, workflow appeared from attempts to automate administrative tasks by storing digital copies of bureaucratic documents such as invoices or customer letters (Chaffey, 1998). It has then evolved into a more complex tool for coordinating groups and insividuals working in organizations. Recently, workflow technology has been presented as a new way to support inter-organizational business processes (Gartner group, 1999, [i], [ii]).

The raison-d'être of workflow lies in its ability to automate business processes and consequently to improve operational efficiency. Chaffey (1998) mentions that workflow provides increased process efficiency through automation, process standardization, improved information availability, automated assignation of tasks to staff, and process monitoring through tools capable of measuring individual or team performance.

Three types of workflow are generally recognized in workflow practitioner-oriented literature (Leymann and Roller, 2000; Chaffey, 1998): **Ad Hoc workflows**, which possess a low potential to add value and which generally consist of non-repetitive tasks. **administrative workflows,** which are also of the low added-value type but which are composed of highly repetitive tasks. And, finally, **production workflows,** which are similar to administrative workflows but correspond to critical business processes for the organization with important value-added potential .

Inter-Organization Workflows

Recently, several e-Marketplaces have been facing difficulties in finding a profitable business model (Wise and Morrison, 2000). To attain economic viability, the e-marketplaces are now redefining their role from purely transactional systems towards more complex infomediaries (InternetWeek, 2000, [i]). Among the new services offered, B2B systems integration appears to be a priority. Indeed, a recent Information Week research (2001) showed that 60% of IT managers surveyed had integrated or were integrating with e-marketplaces for purchasing and sale activities.

To achieve integration, workflow technology is often offered as a possible solution (Van den Heuvel and Weigand, 2000). In fact, as electronic commerce (EC) is becoming the link between intra-organizational processes, workflow technology appears as a possible key to linking EC applications in a process-centered manner (Muth et al., 1999). Moreover, it is expected that by 2003 business process modeling capabilities will be the norm rather than the exception for e-Business solutions (The Hurwitz Group, 2000).

This is bringing new challenges. One of these is to enable workflow interoperability between partners. Research on this problem has for now mainly focused on technical issues and not on language structures (Van den Heuvel and Weigand, 2000). As stated by Van der Aalst (2000): “the semantics of the constructs needed to model inter-organizational workflows should be defined before solving the technical issues (which are mainly syntactical)” (p.68). This paper aims at bringing some elements of explanation to this problem by evaluating if the ontological validity of available formalisms is sufficient to represent workflows crossing organizational boundaries in the context of e-marketplaces.

The benefits of this research are numerous. First, there exists little or no efforts in the literature on the different workflow modeling formalisms for inter-organizational processes. Second, this work will bring more formal basis to the development of e-marketplaces. And finally, finding an adequate common language will allow to have a common denominator representation for translation from a language to another as defined by Curtis et al. (1992).

Evaluation Framework

> *"The idealist that does not distinguish a thing from any of its models cannot account for the multiplicity of schemata of one and the same thing. Consequently, he cannot understand the history of theoretical science, which consists partly in the replacement of some schemata by others."* Bunge, 1977, p.121

In this section, ontology-related definitions are first presented followed by an ontological evaluation framework for IS languages. Afterwards, several process formalisms are reviewed in order to select the most appropriate for an inter-organizational workflow context. Finally, a precise evaluation model combining both ontological and prototypical analyses is presented.

What Is Formalism?

The number of different business modeling languages and the necessity of interoperability between modeling tools has brought a debate on the needs for common languages, the importance of model engineering as a part of software engineering, and on the advantages of ontology-driven modeling (Bézivin, 1998). Ontology, in the context of business modeling, refers to meta-models that define or constraint the model.

For more consistency, a precise terminology as used in the work of Weber (1997) and in ontology-related literature will be defined and used for the remaining of this paper. The basic concept for a language consists in its **grammar**. A grammar can be defined as a set of constructs that include all fundamental objects of the language plus all higher-level constructs that can be generated using those objects. A grammar is composed of **grammatical constructs**. Constructs represent the building blocks of the grammar. Finally, grammars are used to generate **scripts**. Scripts represent a meaningful representation of reality. To evaluate a language, we need to determine if the grammar is appropriate for representation of the real-world phenomenon.

Wand and Weber (1990) have developed a set of models, based on the work on ontology by Bunge (1977, 1979). These models are referred to in literature as the Bunge-Wand-Weber (BWW) models. They have been used to evaluate different grammars such as data flow diagrams, entity relationship or object-oriented diagrams (Green and Rosemann, 2000).

The three BWW models consist in the *representation* model, the *state-tracking* model, and the *decomposition* model. Each of these models aim at evaluating the goodness of an IS deep structure: that is to say how well a machine or a script is associated to a user's model of reality. The representation model provides a way to determine if (1) an IS grammar contains all the necessary constructs needed to represent any phenomenon in the real world and (2) whether any grammatical construct can be unambiguously interpreted. The state-tracking model essentially focuses on dynamics. It consists in evaluating how well changes in the real world will be transposed in the information system. It therefore focuses on dynamics. The good decomposition model tries to determine whether a script can be structured (decomposed into subsystems) in a way that will be easier to understand.

This paper focuses on the representational model for several reasons. First, prior work on process languages has focused on the representational model and we wish to take into account possible comparisons with other evaluations. Second, for the evaluation of a language, the representation model is clearly the most relevant theoretical tool.

For the representation model, the premise is that any modeling language should offer the necessary grammatical constructs to represent all the "things" in the real world that might be necessary for the analysis and design of an information system. Therefore, the representation model is composed of a list of ontological constructs to which the grammatical constructs of our formalism are to be compared for an ontological analysis.

For representation, two criteria are evaluated: Ontological *completeness* and Ontological *clarity*. Ontological completeness can be defined as follows. With O_c being a set of ontological constructs, G_c a set of constructs in an IS grammar, and *f* being the mapping from O_c to G_c, the grammar G_c is ontologically complete with respect to O_c if *f* is total. Otherwise, we are facing a case of ontological incompleteness. More simply, ontological completeness is attained if there exists a grammatical construct that can be used to represent each ontological construct.

Ontological clarity refers to how clearly a real world phenomenon can be represented by the ontological constructs. Wand and Weber (1990) have identified three cases that can jeopardize the ontological clarity of a language: construct overload, construct redundancy, and construct excess. Construct overload is a situation in which a grammatical construct refers to more than one ontological construct. Construct redundancy is a situation in which more than one grammatical construct can be used to represent a single ontological construct. Construct excess is the situation in which a grammatical construct does not fit a corresponding ontological construct.

Multiple Grammar Evaluation

There are situations in which an Information Systems analyst would use multiple grammars to represent the real world. For example, if he uses the Unified Modeling Language (UML), he would have to use the different grammars included in the different diagrams of the language. It may be in order to compensate for the weaknesses present in the initially chosen grammar. Green (1996), in a study of 168 users of Computer-Aided Software Engineering (CASE) packages, found that users were five times more likely to use multiple grammars than a single one. Weber and Zhang (1996) hypothesized that users rely on multiple grammars in order to minimize ontological overlap (this is illustrated in Figure 1). This hypothesis was supported by Green's (1996) findings.

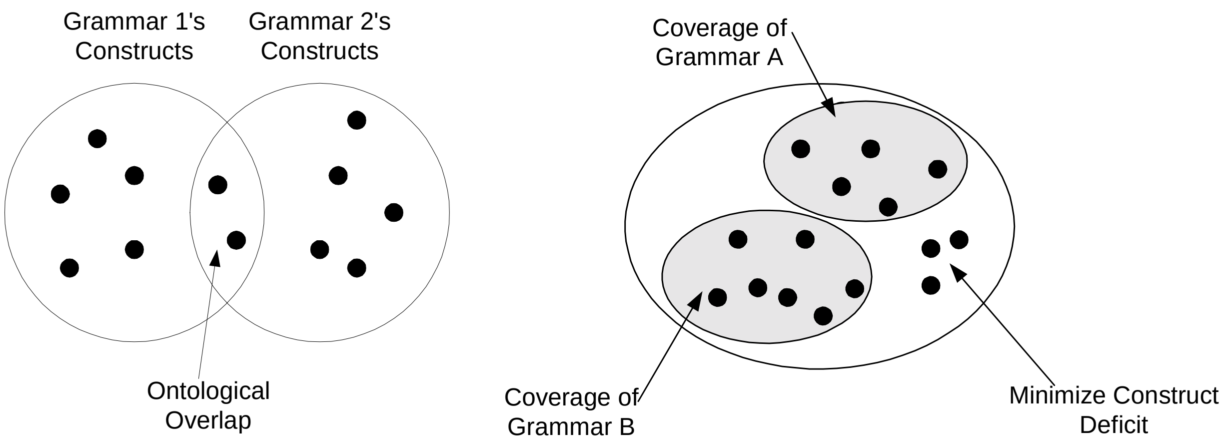


Figure 1 Minimizing Ontological Overlaps – Maximizing Ontological Completeness

Apart from minimizing the ontological overlap, Green (1996) also identified the goal of achieving maximum ontological completeness. Users should choose their grammars in a combination in order to leave the smallest possible number of ontological constructs uncovered by grammatical constructs. Figure 1 illustrates this objective.

Table 1, reproduced from Green and Rosemann (2000) reviews ontological analysis done on modeling grammars.

Table 1 – Ontological Analysis – Related Work – From Green and Rosemann (2000)

Study	Type of Grammar					Ont. Comp.	Ont. Clar.
	Traditional	Structured	Data-centered	O-O	Process		
Wand and Weber (1989)		X (DFD)	X (ER)			Yes	Yes
Wand and Weber (1993)			X (ER)			Yes	Yes
Sinha and Vessey (1995)			X (Relational)			Yes	Yes
Weber and Zhang (1996)			X (NIAM)			Yes	Yes
Weber (1997)			X			Yes	Yes
Green (1997)	X	X	X			Yes	Yes
Parsons and Wand (1997)				X		Yes	Yes
Opdahl And Henderson-Sellers (1999)				X (OML)		Yes	Yes
Green and Rosemann (2000)					X	Yes	Yes

Green and Rosemann (2000) present the only process related grammar for BWW analysis used in the ARIS toolset, the event-driven process chain (EPC). In this evaluation, all four situations of ontological deficiencies were identified, raising concerns by the authors of possible misspecifications in the BWW models. Those misspecifications were identified as a possible over-engineering of the model: it could include constructs that are not relevant to process modeling, the fact that the BWW evaluation does not take into account the objectives of the modeling grammar during ontological analysis suggesting a need for an individualization of the model, and finally a need to extend the BWW model with enterprise-modeling related constructs (Green and Rosemann, 2000).

These concerns motivate the use of a prototype for completing the ontological evaluation. Since this study will adopt a research methodology that combines ontology and prototyping, the next section will identify the best adapted language for such a dual evaluation.

Language Selection

Based on existing literature, a list of six criteria has been selected. The first three -- formal basis, executability, and visualization-- relate to business processes modeling in general, while the last three,-- representation of distinct organizations, modeling document exchange, and representation of the three dimensions of workflow-- relate more precisely to the context inter-organizational workflows. These two groups are discussed in sequence.

The first criterion relates to formality. Curtis et al. (1992) define a formal language as being a language "enactable on a machine". Therefore, a strictly formal language will have a complete mathematical semantic defining it in order to be understood by the machine. Moreover, a formal language has the advantage of a theoretical framework for analysis and representation (Basu and Blanning, 2000). The second criterion relates to the executability of the language. An executable language is a language that can be simulated. Simulation offers the possibility to support the verification of the formal description with respect to correctness, consistency, completeness, absence of deadlock, and alike (Benyoucef and Keller, 2000). Visualization is another criterion. It is generally accepted that visual information is better understood by humans and can improve human intuition and understanding about the process (Sutton et al., 1995).

For this study, three specific criteria have been added to take into account the inter-organization context. An e-marketplace being an intermediary between multiple buyers and sellers (Choudury et al., 1998), the modeling language will have to be able to *represent distinct organizations*. The *modeling of document exchange* relates to the actual tendency of linking intra-organizational processes through the exchange of XML documents to form a global B2B inter-organizational process (Skinstad, 2000 ; Skonnard et al., 2000 ; RosettaNet, 2000). Such processes where each partner takes care of a specified part of the process are defined as loosely coupled inter-organizational processes (Van der Aalst, 2000). And last but not least, the *representation of the three dimensions of workflow* corresponds to the foundations on which this work is based on, that is to say that workflow technology is the key to creating efficient e-business processes. There is a consensus today on the three dimensions that define a workflow (Leymann and Roller, 2000; Van der Aalst, 1998): the business process, representing what is to be done in terms of activities, the IT resource, which will be used in order to automate the tasks, and the organization (which will perform the activity) or the cases (when will the task be performed). Therefore, workflows are often represented using a three-dimensional space model called W^3 (what, who, which or when).

Several formalisms were candidates for the evaluation. Petri Nets are known for their well-known rigorous semantic. The WfMC formalism is the only consortium-led language to exist today. UML enjoys actual popularity and is an object-orientated paradigm. The ANSI formalism boasts a diagrammatic nature and is used in simulation software as IGrafix and

Process. Finally, the SAP formalism is overwhelmingly popular in business process reengineering with the SAP R/3 ERP package. Appendix 1 presents short descriptions of each formalism.

Table 2 presents a summary of the evaluation. The sign “+” means that the criterion is fully respected, while the “–“ means that it is not. A question mark means that it was not possible to determine whether the formalism meets or not a given criterion. The table clearly shows that both the WfMC and the EPC formalisms fail most criteria. The ANSI formalism, by not being formal and not being able to represent a particular invoked application, also has major limitations; which leaves only the Petri Nets and UML along with the classical debate between formal strictness and efficient diagramming in business process modeling.

Table 2 – Comparison of Five Workflow Modeling Languages

	Formal Basis	Executability	Visualization	Distinct Organizations	Document Exchange	W^3
Petri Nets	+	+	+	+	+	-
WfMC	-	-	+	-	-	+
UML	?	+	+	+	+	+
ANSI	-	+	+	+	+	-
EPC	-	-	+	-	-	-

UML was finally chosen to pursue the analysis. This choice is essentially motivated by disciplinary reasons. It is of greater interest to evaluate a language whose strength resides in its representational richness. Although UML possesses a less formal basis than Petri Nets, it allows the representation of the dimensions of workflow that are essential to have an efficient model. The Petri Nets are fundamentally too restrictive since they do not allow the representation of the IT resources used to automate an activity.

Therefore, in the remaining of this paper we will try to answer the following question: Is UML powerful enough from an ontological and practical point of view for the representation of workflows crossing organizational boundaries?

The grammar that will be used for evaluation and modeling purposes in this paper is the Unified Modeling Language (UML) in its basic form, as described in the UML user guide by Booch et al. (1999). In the current study, we will focus on the use of three diagrams: the activity, the state, and the sequence diagrams. This choice is consistent with the previous literature on workflow modeling using activity charts and state charts prior to their inclusion in the UML (Muth et al., 2000). Appendix 2 offers a short description of the three diagrams.

Evaluation Model

A model is, by definition, a simplification of the reality (Booch et al., 1999), that is to say a description of a real world extant. Figure 2 represents these concepts. Adequate modeling requires completing a good representation of the reality. Good representation means having a valid link between reality and its representation, which includes a valid user's model of the real world (Weber, 1997).

As discussed earlier, the BWW models are used for grammar evaluation. Unfortunately, it is difficult to determine in a BWW evaluation if it is the language that is faulty or the evaluation criteria. For example, Green and Rosemann (2000), in an analysis of the EPC, raised doubts about the validity of all the ontological constructs of the BWW model. Under a classical BWW analysis, the language would have been poorly evaluated while its industrial applications are very numerous. It is therefore important to consider a validation methodology that would complete a BWW evaluation.

To evaluate a grammar we therefore need to find a path from the grammatical constructs to the reality as illustrated in Figure 2. A formalism being a language used to model reality, applying the formalism and testing it in a practical manner is the only way to validate unambiguously the abstraction of the model. UML being only partially formal with the State diagrams and executability still being in an early development stage, a prototyping approach is clearly the most adequate and will therefore be used as a second evaluation method.

In this research, we have the precisely defined language features of the UML and we will use a model based on its grammar to write the process program, a workflow management system. Going from the model to the system will allow us to identify if the models are clear enough for a successful development of the system. Afterwards, a reverse iteration will be made in order to see if there could be any lack of information between the completed process program and the initially designed models.

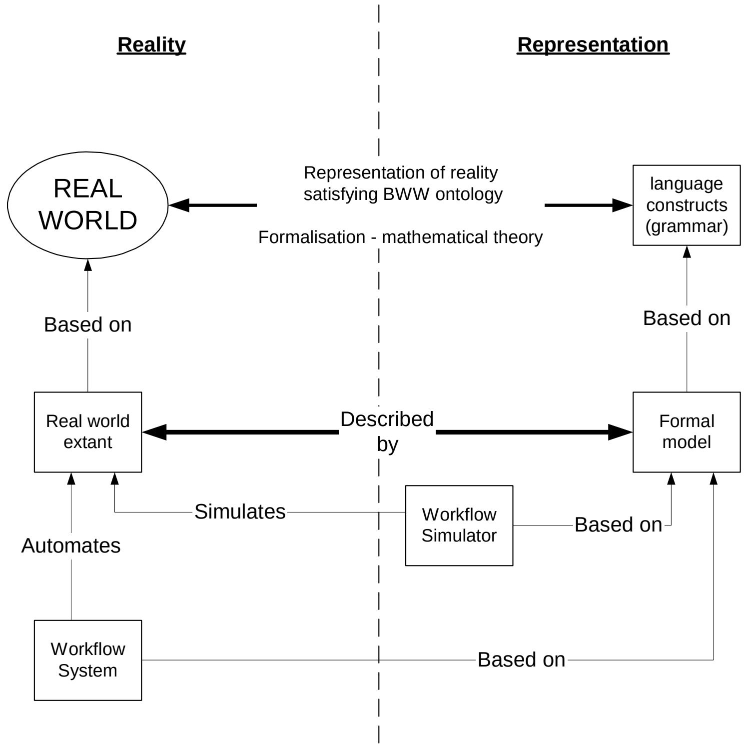


Figure 2 Validating Language Constructs

By combining both the BWW analysis and the prototypical analysis, a convergence between the two sets of results can be expected. Indeed, if a prototypical analysis concludes that there was a lack of information within the models for the development of the system, it would confirm a BWW theoretical conclusion of ontological incompleteness. In our context, the ontological constructs are replaced by the constructs of a process program that aims at automating a business process. Of course, the same language features (or grammatical constructs) are kept for both analysis and are those of the UML.

There still exists a possibility of a double bias in the analysis, that is to say that both the ontological analysis and the prototypical analysis would bring convergent, but biased results. Yet, combining the two analyses is a step towards a more valid and more accepted BWW analysis. It is also a step towards a confirmation of the doubts that Green and Rosemann (2000) raised about a possible over-engineering of the BWW models in their analysis of the EPC. Figure 3 presents our complete language evaluation model, which is the main the methodological contribution of this paper .

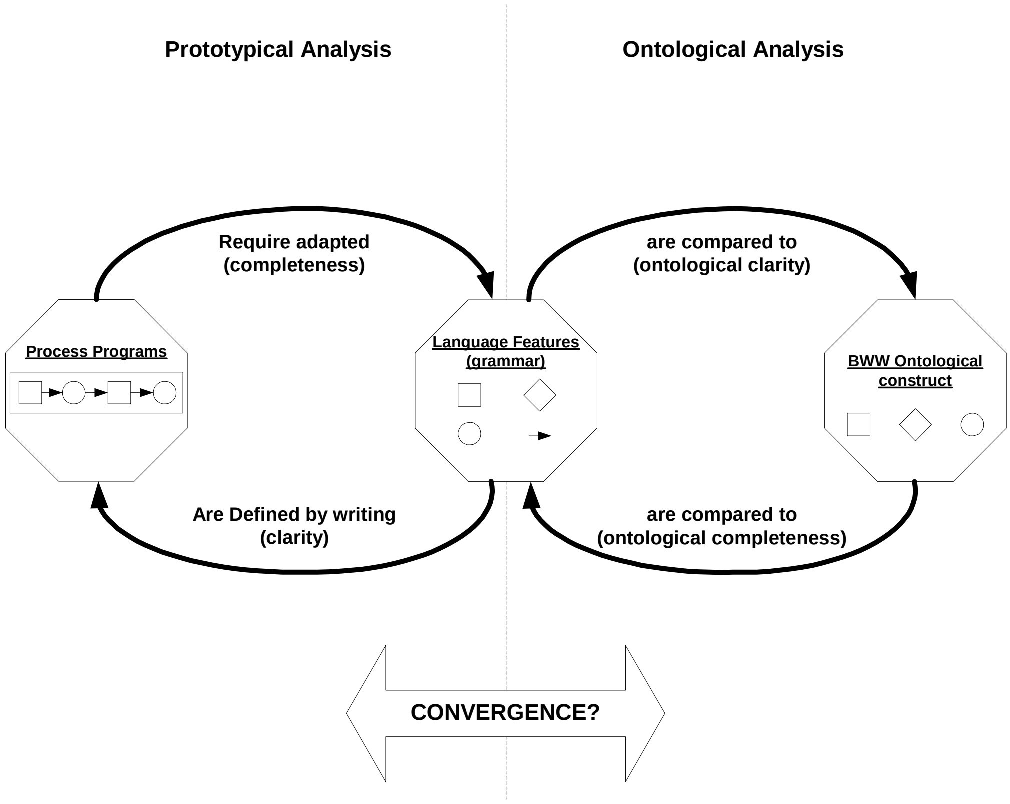


Figure 3 Language Evaluation Model

Using this framework, two analyses need to be conducted. The first analysis is purely theoretical and is an application of the BWW representational model presented earlier. The second analysis is a practical application using a prototype system. Once both analyses are completed, a comparison between the conclusions of the two will be conducted. If convergent results are found, then it will be possible to formulate possible improvements for the UML. If the results are divergent, questions will have to be raised about the validity of the ontological constructs.

Ontological Validation

The methodology for the BWW analysis is similar to previous BWW evaluations presented in Table 1. It consists in using the definitions of all ontological constructs and to find a possible mapping with the grammatical constructs of the language under evaluation. Afterwards, a forward analysis from the grammatical constructs to the ontological constructs is done to evaluate ontological clarity. The backward analysis will check for ontological completeness. If deficiencies are found, their possible consequences will be identified.

Matching Grammatical and Ontological Constructs

The elementary unit in the BWW model is the “Thing”. This elementary ontological construct can be associated to the object in our three diagrams. Contrary to the EPC, the activity chart can show the transformations made on objects during activities and therefore solve a case of ontological incompleteness.

An activity in the activity diagram will sometimes involve transformations made on objects. We will therefore interpret it as a Property in general for the object. This is relevant with Green and Rosemann’s (2000) analysis of the EPC that interpreted the function in the EPC as a property in general too.

Class and Kind are respectively represented in the UML in the class diagram with the class and the generalisation constructs. This diagram represents the static aspects of a system and is therefore not included in the analysis. Consequently, the absence of direct match will not be considered as an ontological incompleteness.

States of the thing are represented by the state of the object in the activity diagram or by the state construct in the state diagram. A state machine in the state diagram represents the Conceivable State Space, defined as all the states that a thing may ever assume. A Lawful State Space can be represented in a state diagram using substates. Stable States and Unstable States can respectively be represented by the final state or the initial state in a state diagram.

Events are represented as the trigger for a transition in the state diagram. But events can also be represented as an activity in the activity diagram. There is no grammatical differentiation for External events and Internal events but the use of the Uses Cases for human-machine interaction diagram or the use of stereotypes could help make the differentiation possible. The Conceivable Event Space can be observed on the state machine of a thing by looking at all transitions triggers. There exists no construct for a poorly-defined event and well-defined events use the same grammatical construct as a normal event.

Transformations are represented by an activity in the activity diagram. Lawful transformations are represented by guard conditions on transitions. There is no grammatical construct for Lawful event space.

History can be modelled using the shallow history state construct in the state diagram. Acts on cannot be represented in the same way as it is defined in the definitions of the ontological constructs but could eventually be associated to the composition relationship in the class diagram, for example, in a composition relation between a thing “Activity” and a thing “Project”

Coupling relationships between objects (“things”) in the system can be represented using messages in the sequence diagram. In the case of workflow management, it is the coupling between actors, organizational units or organizations (between the swimlanes in the activity diagram) that are most interesting to illustrate cross-organizational workflow. A System can be represented using the sequence diagrams. Indeed, if multiple objects are involved, dividing the system won’t eliminate the existing couplings between those objects. It could also be represented using the package construct of the UML. The System composition is represented using the object construct. Once again, the System environment, that is to say external and internal things to the system, can’t be differentiated without a stereotype. The System structure is represented using the message construct in the sequence diagram.

Subsystems can be represented using a stereotyped package. Relationships of composition and generalization would show the System decomposition and the Level structure. Unfortunately, the package and the relationships are not part of the three views that originally defined our language.

Table 3 – BWW Representation Model Analysis for Dynamic Aspects of UML

Ontological Construct	**Activity diagram**	**State diagram**	**Sequence diagram**	**Other views**
Thing	Object Swimlane	Object	Object	
Property ◦ In Particular ◦ In General ◦ Intrinsic ◦ Mutual ◦ Emergent ◦ Hereditary ◦ Attributes	Activity Swimlane			
Class				Class (class diagram)
Kind				Generalization (Class diagram)
State	State of the object	State		
Conceivable State Space		State Machine		
State Law		State→Transition→State		
Lawful State Space		Substates		
Process	Activity diagram Activity			
Event	Activity	Trigger		
Conceivable Event Space		All triggers		
Transformation	Activity			
Lawful Transformation: ◦ Stability Condition ◦ Corrective Action	Guard conditions on transitions			
Lawful Event Space				
History		Shallow history state construct		
Acts On				
Coupling: Binding Mutual Property			Messages	
System			Sequence Diagram	Package with <<system>>
System Composition			Object	
System Environment			<<Stereotype>>	
System Structure			Messages	
Subsystem				Package with <<subsystem>>
System Decomposition				Composition
Level Structure				Generalization
External Event		<< Stereotype>>		
Stable State		Final State		
Unstable State		Initial State		
Internal Event		<< Stereotype>>		
Well-Defined Event				
Poorly-Defined Event				

Results of the Ontological Evaluation

The complete ontological evaluation has been transcribed in Table 3. For an analysis of ontological completeness, it seems that several constructs can't find representation in any views: Lawful event space, Acts on, Well-defined event, Poorly-defined event. Consequently, from a purely theoretical point of view, for workflow modeling, the UML must be considered as ontologically incomplete.

Moreover, many examples show that the UML for workflow modeling is not ontologically clear. Indeed, we face construct overload for the activity construct in the activity diagram that can represent a transformation, a process, a property in general, or an event. Construct overload was also observed for the swimlane of the activity diagram that can represent either a thing (such as an organization) or a hereditary property of the thing (a user of the organization). We also face construct redundancy in the case of the Process ontological construct that can be either represented by a complete activity diagram or by the activity construct in an activity diagram. In the case of the activity diagram, construct excess can also be identified since the branching construct could not find any matching ontological construct.

Also, since we are using multiple grammars in the analysis, it is necessary to evaluate ontological overlap between the different views of the UML. Unsurprisingly, they mainly concern the activity diagram for which there exists many overlaps with the state diagram. The activity diagram was the last added diagram in the UML and consisted in bringing a "process" view to information systems. Ontological analysis shows that it does not integrate perfectly with the other views. Clearly, the goal of minimizing ontological overlap is not attained here.

The consequences of those deficiencies are not negligible for the systems analyst. First, he may not possess all necessary constructs to complete his models. Second, some confusion may arise between different constructs because of the overload and scripts could therefore be interpreted differently from an analyst to another. Finally, the analyst may be tempted to use only the activity diagram because it covers most of the necessary ontological constructs.

But these harsh conclusions for the UML need to be softened for several reasons. First, we need to be cautious towards the ontological incompleteness conclusion. Using multiple views, this incompleteness has been minimized to only four constructs that are not necessarily essential to workflow modeling and this could confirm a conclusion by Green and Rosemann (2000) who raised the question of a possible over-engineering of the BWW model and a need for a contextual individualisation of the model. Second, the construct redundancy that has been identified refers to the possibility of having different levels of abstraction for the activity diagram. While this may look confusing at first, adequate stereotyping on different activity diagrams could clearly identify at what level we are. Third, the construct excess refers to the absence of an ontological construct that identifies branching. Intuitively, this is an ontological construct that would definitely be essential to any workflow modeling grammar.

Now that conclusions have been raised from the ontological analysis, a prototypical analysis will complete the evaluation.

Prototypical Validation

Methodology

The prototypical analysis consists in modeling a B2B business process and to automate it using a WFMS. It is precisely the transposition of the model in the system that is analyzed. Indeed, the purpose of the study is to determine if the models are clear and complete enough for the successful development of a WFMS.

The research context chosen is the aeronautical industrial sector and, more precisely, the exchange of quality control documents between manufacturers and their numerous sub-contractors. The names of the clients and suppliers are voluntarily not mentioned for confidentiality reasons. The existing inter-organizational processes dramatically lack automation and it is therefore anticipated that important economies of scale could me made by using a market aggregator that would automate B2B processes in a workflow-oriented manner.

We therefore need a development model that: (1) uses UML for modeling and (2) aims primarily at defining standard B2B processes. The development model for the RosettaNet

Consortium that has successfully implemented B2B standard for over 60 companies in the IT and electronics components industry (Internetweek, 2000, [ii]) was a natural candidate. The model aims at creating Partner Interface Processes (PIP) that defines standard interfaces for developers. A PIP is composed of a new "generic" B2B process, a dictionary of common properties for the industry and of XML document type definitions (RosettaNet, 2000).

Business Model

"As is" process. Using corporate documentation, a preliminary blueprint of the B2B workflow was drawn and presented to five different experts or managers in aerospace quality control. Two were working for two different large manufacturers while the three others were working for different sub-contractors. Some minor modifications were made on the business blueprint and complementary corporate documentation was sometimes collected during the meetings. With the modifications suggested by the respondents and the supplemental corporate information obtained, a final blueprint of the business process was finally drawn. It is presented in the Appendix 3.

The process starts with a supplier having produced a given number of items ordered by a manufacturer. There is an optional quality inspection to be made on a randomly chosen item in the shipment if the supplier produces it for the first time or if modifications were made in the manufacturing process. This inspection leads to the writing of a first article inspection report that is kept in the supplier's documentary vault while another copy is sent with the shipment. The failure of this inspection is not included in the boundaries of our studied process because it involves another process of B2B communication to determine the reasons of non-compliance.

For every shipment, the supplier must complete a mandatory inspection that consists again in choosing randomly an item in the shipment and to inspect it. This leads to the writing of a certificate of conformity, also called a certificate of compliance. Once again, a copy is kept in the supplier's documentary vault while another copy is joined to the shipment. If the item is found to be non-compliant with the manufacturers requirements, a supplier report of non-conformity (RNC) is sent to the manufacturer describing the defect and asking for a study of the non-conformity. If the manufacturer accepts the non-conformity, he sends back the

RNC mentioning that the article is accepted “as is”. Otherwise, the RNC is sent back mentioning that the article is rejected. Sometimes, a certificate of acceptance or a certificate of rejection can replace the RNC. Once again, a copy of both of these documents is kept in the vaults of both the supplier and the manufacturer.

When the shipment arrives at the manufacturer, quality control documents are inspected. If the supplier has a sufficiently good rating for the manufacturer, inspection at reception can be skipped. Otherwise, another inspection is made and, if it is successful, the received items are placed in the inventory. If the inspection finds a defect, all the received part are immediately placed in quarantine and the non-conformity is studied. A RNC is filled and if the item can be accepted “as is”, it is placed in the inventory. The refusal of the item is not included in the boundaries of this process because it involves another complex process of repairing; reworking or modifying the item according to supplemental analysis made and would unnecessarily complicate the case of study.

“To be” process. The business process analysis phase aims at creating a new generic “to be” process modeled using the UML formalism. Several governmental and quality control agencies impose both the process and the exchanged documents, and therefore little modifications were possible. In fact, it was found that the two manufacturers had very similar processes with their subcontractors and almost similar quality control documents. The redesign of the process includes a third party (the e-hub) and was made using the basic guidelines of business process reengineering (BPR) as presented in Hammer (1990). While the application of those guidelines may be considered as being a non-exhaustive method for workflow analysis and redesign, it is important to remember that the objective of this work is to evaluate an inter-organizational modeling language and not the applicability of BPR methods in an inter-organizational context. The modified business process is presented in the Appendix 4.

The new process is organized around the outcome: having a shipment of compliant parts in the manufacturer’s inventory. The original process was already very outcome-focused but involved non-value added activities such a filling, storing or sending paper-based documents through multiple communication channels. In the new process, document-related activities only involve information capturing on a web-based interface, thus reducing the number of channels to one. All other activities aim at filling the final outcome of the process. This is very well transposed in the models. Indeed, as presented in Appendix 4, the primary control variable of the flow is the state of the article shipment item, which is purely outcome-related. The second point, which consists in having the users of the outcome to perform the process, was already well respected in the initial process. Indeed, if the outcomes are separated as having (1) a compliant shipment of items sent to the manufacturer and (2) an inspected shipment of items in inventory, both activities are performed by those who will use the outcomes. Another guideline consists in treating geographically dispersed resources as if they were centralized. The new process is to be executed by a centralized WFMS and will coordinate work as if it was done in a single organization. Moreover, all documents are to be stored in a centralized location. Through aggregation, the system aims at minimizing quality control costs. The decision points are all located where the work is done except for the "study non-conformity" activity. But again, this activity cannot be modified because it has to be completed by the manufacturer. Finally, information capture is now done at the source.

System Development

For the implementation framework phase, the XML document format used consisted of those already present in the WMFS software package. The Dictionaries step was not completed because it is not relevant for this study. A prototypical process model as presented in Pfleeger (1998) was followed. Every revision to the model was considered as a possible completeness problem (if the model lacks the information) or clarity problem (if we have to hesitate between constructs in the system).

To automate or semi-automate this inter-organizational workflow, the Weblogic Process Integrator (WLPI) of BEA SYSTEMS was used. It was decided to have only one workflow engine to support the process instead of multiple engines in every organization for several reasons. First, business integration in the aeronautical industry will have to include integration with existing ERP systems of manufacturers that already support some form of documentary management, therefore making the installation of a local WFMS clearly unappealing for them. A central WFMS sending XML documents readable by different forms of legacy systems makes the solution more acceptable to any business partner. Second, for smaller firms, the purchase of a WFMS is way beyond budget. A web-based interface for document management with routing controlled by an outsourced WFMS is much more appropriate to their context. Third, using an intra-organization-like architecture in an inter-organizational context greatly reduces the risk associated with the prototype development while still being adequate for the study.

This system is based on the Weblogic application server that enables the use of Java 2 Enterprise Edition (J2EE) specifications such as Java Server Pages (JSPs), Enterprise Java Beans (EJB) or Java Messaging Services (JMS). At the core of the system is the Weblogic Process Integrator Server, a workflow engine dynamically executing workflows defined in the workflow studio using a flowchart tool. For each activity defined in the workflow, several tasks can be completed such as sending an electronic mail message, launching an application or sending an XML document to a client application. Workflow models, instances and variables are stored in a relational database.

For the prototyping context, the WFMS is used as a work coordination tool, which is generally the main task of a WFMS (Chaffey, 1998). Each organization uses a client application that (1) gives reminders to complete an inspection or to fill a document and (2) asks specific questions on the result of evaluation or of an inspection for adequate workflow routing. Communication between the workflow engine (the electronic intermediary) and the different client applications (the organizations) is made through the exchange of XML documents. The workflow engine predefines the document type definitions used.

To define a workflow, WLPI uses a workflow definition meta-model presented in Figure 4. The execution logic is represented using eight grammatical constructs defined as nodes in the meta-model. Each node can invoke different actions such as invoking an application, sending a reminder or sending an XML document.

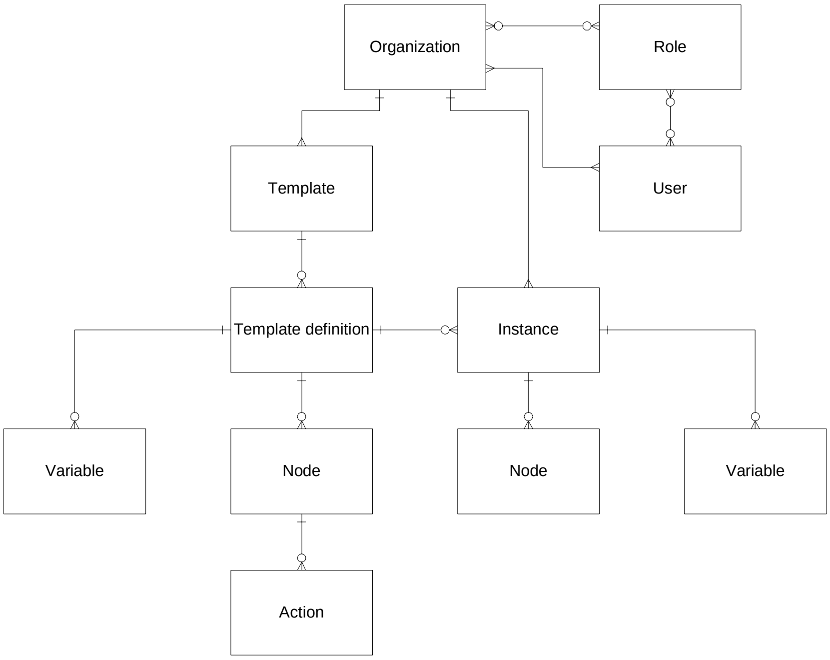
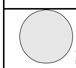
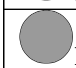
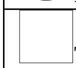
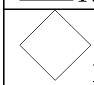
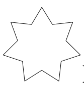
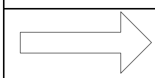
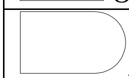


Figure 4 Weblogic Process Integrator Workflow Meta-model

For parameterisation, the Workflow studio is used to describe in a flowchart manner the workflow to be automated. It uses eight grammatical constructs that are briefly described in Table 5 with their equivalences in the UML.

Table 5 – Weblogic Process Integrator Grammatical Workflow Constructs

PROCESS PROGRAM CONSTRUCT	VENDOR DESCRIPTION	UML EQUIVALENT		
		Activity Diagram	State Diagram	Sequence Diagram
 Start	Indicates the start of a workflow	Initial State		
 Done	Indicates the end of a workflow	Final State		
 Task	Defines a task in a workflow	Activity		
 Decision	Represents a condition in the workflow that evaluates to be true or false	Sequential Branch		
 Event	Represents an event that can be triggered either internally or externally by an XML message. Sub-actions can be performed and/or workflow variables can be set as the result of the trigger of the event	Activity	Trigger	
 Connector	Used to connect workflow nodes. The arrow directs you to the subsequent task in the flow.	Triggerless transition		
 Or	Allows joining of one or more task, decision, or event with an OR condition	?		
 And	Allows joining of one or more task, decision or event with an AND condition	Join		

Prior to parameterisation, tests were made to verify that the engine could adequately execute a basic workflow and could send simple XML documents to the workflow clients. Afterwards, the models were used to set the WFMS to our context using the translation present in Tables 5-7. The observations made during the system development follows.

Table 6 – Non-diagrammatical Constructs Used in Weblogic Process Integrator

Construct	Representation in UML
Organization	Swimlane
User	Swimlane
Role	Swimlane
Workflow variable	Object State

Table 7 – Some Possible Actions for Tasks

Actions	Representation in UML
Call program	Object with stereotype
Sending an XML document	Object with stereotype
Send an e-mail	Object with stereotype
Assign task to user	Activity

Clarity

The clarity analysis consists in transposing the model into the WFMS to identify ambiguities. Two cases of ambiguity were identified. The first case concerns the activity construct in the activity diagram. It was ambiguously used as an event construct too because no grammatical construct exists for an event in the activity diagram of the UML while these constructs were distinct in the process program. For example, receiving an XML document is modelled as an activity in our models while it is an event in the process program. Clearly, mistakes could be made while transposing the model in a WFMS because room is left for interpretation.

The second ambiguity concerns the swimlanes of the activity diagram. Our process program required making a clear distinction between organizations, roles and users. The swimlanes of our model were not sufficiently precise to make such distinctions. In the case of workflow management, more precision is needed and adequate meta-modeling appears unavoidable to clearly identify the relationships between users, their roles and their organizations. Unsurprisingly, such precisions are made in RosettaNet UML extensions (1999).

Completeness

The completeness analysis consists in comparing the process program requirements for adequate execution of the process to the language features we had for modelling the workflow. Only one case of incompleteness was observed. As defined in Booch et al. (1999): “A Stereotype is an extension of the vocabulary of the UML, allowing you to create new kinds of building blocks similar to existing ones but specific to your problem” (p.78). Therefore, for every process program construct that lacked a precise grammatical symbol, we could freely define a new construct to represent it. For our prototype system, we developed stereotypes for each possible action for an activity as represented in Table 7.

However, since there does not exist a symbol for the “exclusive or” join, we could not define a stereotype to represent it. This is clearly a completeness deficiency in the activity diagram. Indeed, joining in the activity diagram can only be made on an “AND” basis and not an “EXCLUSIVE OR” basis. This observation was not made during the parameterisation of the system but when determining the translation scheme presented of Table 5.

Final Remarks

A final observation made during the development of our prototype is the little use we made of the views other than the activity diagram. In fact, this is not very surprising since the flowcharting tool used in WLPI in very similar to the activity diagram.

In this analysis, we identified cases of clarity and completeness problem and made an observation on the use of multiple views. We will now compare these results with those of the ontological analysis and discuss convergent or non-convergent results.

RECONCILIATION AND DISCUSSION

In this section, the results obtained from the ontological and from the prototypical evaluation are compared. We will first evaluate completeness issues, followed by clarity problems, and finally by grammatical overlaps.

Completeness

From the ontological evaluation, we concluded that the UML was ontologically incomplete because it lacked the lawful event space, Acts on, Well-defined event and the Poorly-defined event constructs. Those completeness problems were not observed for the development of the prototype system. In fact, those constructs are fundamentally philosophical and have little to do with workflow modeling. They most probably illustrate a case of contextual over-engineering of the BWW models for a situation as specific as cross-organizational workflows and illustrate the need for a contextual specification of the models.

In the prototypical evaluation, we lacked a construct of "Exclusive Or" for joining two activities. This result cannot be compared with the ontological analysis because branching in the UML had no ontological equivalent in the BWW models. But clearly, such a joining is essential in process modeling and this illustrates once again the need for a specification of the BWW models so that it can include branching. It also illustrates the need to add an "Exclusive Or" construct in the activity diagram.

Clarity

We observed in the ontological evaluation that the activity construct in the activity diagram brought a construct overload problem. This result is convergent with the prototypical analysis in which we faced confusion between the activity and the event construct when transposing our workflow model into the WFMS. Clearly, the activity diagram lacks an event construct and further specifications of the language should aim at including this construct.

The construct redundancy problem of the Process ontological construct, which could either be represented by a complete activity diagram or by a single activity in an activity diagram, was not a particular problem for the development of our system. In fact, this result is explainable by the fact that the WFMS imposes indirectly the appropriate level of abstraction of the task as it coordinated the work of individuals.

Finally, during both ontological and prototypical validation, we faced clarity problems with the swimlanes of the UML activity diagram that could ontologically represent a thing, or a hereditary property of that thing and, in a more practical context, users, roles and organizations. Further specifications of the UML should aim at defining a more precise semantic for the swimlanes in the activity diagram that could permit the representation of organizational hierarchical levels.

Multiple Views

The ontological evaluation revealed that the activity diagram had several overlaps with other views in the UML. During the development of the prototype, little use was made of diagrams other than the activity diagram. These results are clearly convergent. It illustrates once again how the activity diagram integrates poorly with other views of the language.

It is difficult here to suggest possible improvements because reducing overlaps could also lower the complete ontological completeness of the language. In fact, further improvements should aim at both reducing overlaps while maintaining the overall ontological coverage, which can be considered as very satisfactory for the UML.

Conclusion

To this day, research on inter-organizational workflows has essentially focused on technical aspects of inter-operability between WFMS. In fact, very little work has been done in order to define a precise semantic for inter-organizational business modeling. This paper intended to bridge that gap by finding a solution to this problem from an IS perspective.

To provide a framework for this research, we chose to rely on the work of Wand and Weber (1993). This paper aimed at determining if the ontological validity of available formalisms was sufficient to represent workflows crossing organizational boundaries. A review of several formalisms revealed that the UML fulfils essential representation criteria related to B2B workflows. Moreover, it possesses several extension possibilities that makes it a powerful –and popular, language for business modeling.

Three contributions can be stressed out. First of all, this work presented a more rigorous methodological framework for ontological grammar evaluation than previous studies by combining an analysis using the BWW representation model with a prototypical analysis. Prior research had raised doubts on the validity of the BWW model in workflow-modeling contexts by assuming that the tested grammar had little deficiencies. Clearly, a more rigorous methodological framework was needed in order to discuss the validity of the BWW models. By using the ontology of a WFMS in addition to the BWW ontology, conclusions drawn out of convergent results from both analyses can be considered more rigorous.

Second, by using this new methodological framework, little ontological deficiencies were identified in the UML. This result could mainly be attributed to the extension capabilities of the grammar. Nevertheless, some room for improvement has been identified and specific enhancements were suggested. The most challenging concerns the overlaps of the activity diagram with other views in the UML. Clearly, this view, which is essential for workflow modeling, does not fit well with the other views of the popular language, and this problem could worsen the current hesitancy for developers to use a workflow paradigm for IS development.

Third, the two analyses confirmed the need for the development of specific ontologies for workflow modeling. There is undeniably room for both a universal ontology for the representation of real-world phenomenon such as the BWW models and for more specific contextual ontologies, which could also be based on the BWW ontology. In fact, the BWW representation model is probably too fundamental for a precise context such as cross-organizational business process modeling. Indeed, while it was first concluded that the UML had ontological deficiencies, our models were sufficient for the successful development of a prototype system in the aerospace industry.

From these contributions, directions for future research can be identified. While this work credited the UML with little deficiencies, some of the aforementioned suggested improvements could boost the already-high quality of its grammar. Another interesting research direction could be the definition of common extensions for the community of WFMS developers. Indeed, while the extension mechanisms are a powerful tool against ontological deficiencies, such extensions need to be defined in a matter that is fully understandable by all business partners. To this day, only extensions to the UML for business modeling at large and not workflow management have been defined.

Such a contextual approach could also be a research track for ontological evaluation models. Further research in this area could aim at defining a particular ontology for workflow modeling based on the BWW models and on the meta-models from several WFMS. There are over 250 WFMS systems available on the market, which would allow a large sample of study. With a precise ontology for workflow, universal extensions could be defined and therefore ease the task of making UML models a possible direct input for WFMS parameterisation.

REFERENCES

American National Standard, ANSI X3.6-1970, "Flowchart Symbols and their Usage in Information Processing".

Basu A., Blanning R., A Formal Approach to Workflow Analysis, Information Systems Research, 11(1), pp. 17-36, March 2000.

BEA Systems, BEA Weblogic Process Integrator Studio User Guide, Edition 1.1, October 2000.

Benyoucef M. and Keller R., An Evaluation of Formalisms for Negotiations in E-Commerce, In Proceedings of the Workshop on Distributed Communities on the Web, pages 45-54, Quebec City, QC, Canada, June 2000.

Bézivin J., Who is afraid of ontologies? In OOPSLA ’98 Workshops, Vancouver, Canada, October 1998.

Booch G., Rumbauch J., Jacobson I., The Unified Modeling Language User Guide, Addison Wesley, 1999.

Bunge M., Treatise on Basic Philosophy: Volume 3: Ontology I: The Furniture of the World, Reidel, Dordrecht, Holland, 1977.

Bunge M., Treatise on Basic Philosophy: Volume 4: Ontology II: A World of Systems, Reidel, Dordrecht, Holland, 1979.

Chaffey D., Groupware, Workflow and Intranets: Reengineering the Enterprise with Collaborative Software, Digital Press, 1998.

Choudury V., Hartzel K., Konsynski B., Uses and consequences of Electronic Markets: An Empirical Investigation in the Aircarft Parts Industry, MIS Quarterly, pp 471-507, December 1998.

Curtis B., Kellner M., Over J., Process Modeling, Communications of the ACM, 35(9), pp75-89, September 1992.

Curran T., Ladd A., SAP R/3 Business Blueprint, Prentice Hall, 2000.

Evans A., France R., Lano K, Rumpe B., Developing the UML as a Formal Modeling Notation, from UML: Beyond the notation. International Workshop Mulhouse France, 1998.

Gartner Group [i], E-Business to Workflow: Wake Up!, October 7th, 1999.

Gartner Group [ii], Why E-Business Craves Workflow Technology, December 1999.

Garicano L., Kaplan S.N., The effects of business-to-business e-commerce on transaction costs, Working Paper 8017, National Bureau of Economic Research, 2000.

Green P., An Ontological Analysis of Information Systems Analysis and Design (ISAD) Grammars in Upper CASE Tools, Unpublished Ph.D. dissertation, Department of Commerce, The University of Queensland, 1996.

Green P., Rosemann M., Integrated Process Modeling: An Ontological Evaluation, Information Systems Journal, 25(2), pp.73-87, 2000.

Hammer M., Reengineering Work: Don't Automate, Obliterate, Harvard Business Review, pp104-112, July-August 1990.

Hurwitz Group, Hurwitz TrendWatch, B2Bi Mantra: Business Process and Workflow, August 2000.

InternetWeek [ii], RosettaNet Sharpens Focus, November 6th, 2000.

InternetWeek [i], Wanted: A Net Marketplace Model That Works, November 10th, 2000.

InformationWeek, The B-To-B Integration Equation, February 12th, 2000.

Intellicorp, Business Process Design with the SAP R/3 Reference Model and Object-Oriented Information Engineering, White Paper, 1996.

Keller, G., Nuettgens, M., Scheer, A.-W., Semantische Prozessmodellierung auf der Basis "Ereignisgesteuerte Prozessketten (EPK)", in: Scheer, A-W. (Hrsg.), Veroeffentlichungen des Instituts fuer Wirtschaftsinformatik. Heft 89. Saarbruecken 1992.

Kennedy Carter, An overview of the xUML System Development Process, White Paper, 2000. Du web: http://www.kc.com

Leymann F., Roller D., Production Workflow: Concepts and Techniques, Prentice Hall, 2000.

Lucking-Reiley D.H, Spulber D.F., Business-to-Business Electronic Commerce, Journal of Economic Perspectives, 15(1), pp. 55-68, 2001.

Muth P., Weissenfels J., and Weikum G.. What Workflow Technology Can Do For Electronic Commerce. Technical report, University of the Saarland, Department of Computer Science, Saarbrucken, Germany, 1999.

Opdahl A.L., Henderson-Sellers B., Grounding the OML Metamodel in Ontology, Working Paper, pp. 1-59, University of Queensland, Brisbane, 1999.

Parsons J., Wand Y., Using Object in Systems Analysis, Communications of the ACM, 40(12), pp. 104-110, 1997.

Peterson, J. L., Petri net theory and the modelling of systems, Prentice-Hall, 1981.

Pfleeger, S. L., Software Engineering, Theory and practice, Prentice Hall, 1997.

Popper K., La Connaissance Objective, Champs Flammarion, 1979.

Reisig, W., Petri nets, an introduction, EATCS Monographs on Theoretical Computer Science, Springer-Verlag, 1985.

Rittgen P., Modified EPC and their Formal Semantics, Institüt für Wirtschafsinformatik, October 1999.

Rivard S., Talbot J., Le développement de systèmes d'information, Presses de l'Université du Québec, 1998.

Rosemann M., Green P., Enhancing the Process of Ontological Analysis – The "Who cares" Dimension, in Proceedings of the Information Systems Foundations Workshop, Sydney, Australia, 1999.

RosettaNet, RosettaNet Implementation Framework, November 1999.

RosettaNet, UML Extension for e-Business Partner Interface Process Modeling, Version 0.5 Draft, 1999.

Sinha A.P., Vessey I., End-User Data Modeling: An Ontological Evaluation of Relational and Object-Oriented Schema Diagrams. Working Paper, Indiana University, Indiana (1995).

Skonnard A., Laskey B., Biztalk Server 2000: Architecture and Tools for Trading Partner Integration, MSDN Magazine, May 2000.

Skinstad, R., Business process integration through XML, Netfish Technologies , 2000. Du web: http://www.infoloom.com/gcaconfs/WEB/paris2000/S10-03.HTM

Van der Aalst, The Application of Petri Nets to Workflow Management, The Journal of Circuits, Systems and Computers, 8(1), pp 21-66, 1998.

Van Der Aalst W.M.P., Formalization and Verification of Event-driven Process Chains, Information and Software Technology, 41(10), pp 639-650, 1999.

Van Der Aalst W.M.P., Loosely Coupled Interorganizational Workflows: modelling and analysing workflows crossing organizational boundaries, Information and Management, 37, pp. 67-75, 2000.

Van Den Heuvel W.-J., Weigand H., Cross-Organizational Workflow Integration using Contracts, In Proceedings of OOPSLA 2000 Workshops, Minneapolis, USA, 2000.

Wand Y., Weber R., An ontological evaluation of systems analysis and design methods. In Information Systems Concepts: An in-Depth Analysis, Falkenberg E.D., Lindgreen P., editors, pp. 79-107, North-Holland, 1989.

Wand Y., Weber R., An Ontological Model of an Information System, IEEE Transactions on Software Engineering, 16(11), pp. 1282-1292, 1990.

Wand Y., Weber R., On the Ontological Expressiveness of Information Systems Analysis and Design Grammars, Journal of Information Systems 3(4), pp 217-237, 1993.

Wand Y., Weber R., On the Deep Structure of Information Systems, Information Systems Journal, 5(3), pp. 203-223, 1995.

Weber R., Ontological Foundations of Information Systems, Accounting Research Methodology Monograph No.4, Coopers and Lybrand, Australia, 1997.

Weber R., The Information Systems Discipline: The Need for and Nature of a Foundational Core in Proceedings of the Information Systems Foundations Workshop, Sydney, Australia, 1999.

Weber R., Zhang Y., An analytical evaluation of NIAM's Grammar for Conceptual Schema Diagrams, Information Systems Journal, 6(2), pp.147-170, 1996.

Wise R., Morrison D., Beyond the Exchange: The Future of B2B, Harvard Business Review, pp. 86-96, November-December 2000.

Workflow Management Coalition, The Workflow Reference Model, January 1995.

Workflow Management Coalition, Interface 1: Process Definition Interchange, January 1999.

Zwass V., Structure and Macro-level Impact of Electronic Commerce: From Technological Infrastructure to Electronic Marketplaces in Kenneth E. Kendall (ed.), Emerging Information Technologies, Thousand Oaks, CA: Sage Publications, 1998.

Appendix 1 – Description of the Formalisms

Petri Nets

First invented by Karl Adam Petri in the early 60s, the Petri Nets formalism was originally a language that presented the possibility of modeling parallel treatments in a system instead of only sequential (Pfleeger, 1997). The graphical notation composed of only three elements was quickly used to model business processes and has now been formalized to model workflows (Van der Aalst, 1998). All constructs in a Petri Net can be demonstrated mathematically and furthermore, the formalism can illustrate cases by the use of tokens that move through the net (Peterson, 1981). The use of these tokens therefore makes the net executable and very well adapted to simulation (Reisig, 1985). Also, the adaptability of Petri Nets for loosely coupled inter-organizational workflows has been demonstrated by Van der Aalst (2000). Unfortunately, the limitation of the number of elements that compose the Petri Nets diagrams restricts the number of workflow dimensions illustrated to two: the IT resources used cannot be represented. (Van der Aalst, 1998).

WfMC

The Workflow Management Coalition is a grouping of companies that recognizes common characteristics to the different Workflow Management (WFM) products on the market and, using these characteristics, aims at enabling the different existing WFM products to work together using an identical set of standards (WfMC, 1995). The objective is to attain a high level of interoperability between WFMS, to ease their integration with other systems such as document management (or electronic marketplaces in our case of study), and, ultimately, to make more effective the use of WFMS.

To define workflow processes, the WfMC uses a basic meta-model composed of a set of objects that represents simple processes. Type of workflow, activity, transition condition, workflow relevant data, role, and invoked application are the six defined object types. The list isn't exhaustive since objects can be added for further functionality like for vendor specific extensions. The following figure is a representation of the model:

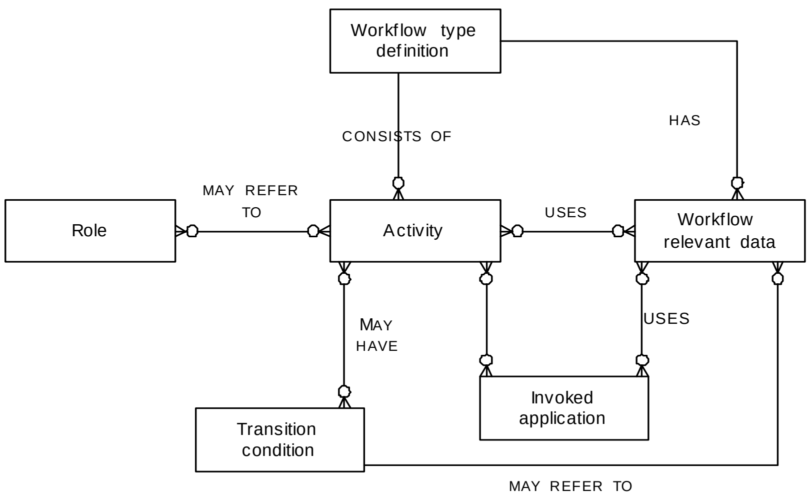


Figure 5 Basic Process Definition Meta-model

As illustrated by the meta-model, the formalism can represent the three dimensions of workflow. It could also be extended with an "Organization" object and a "Document" Object to concord with the other criterias of our context of study (WfMC, 1999).

Unfortunately, the WfMC formalism is a coded formalism for which there only exists a visual format under development. Moreover, it is not based on any mathematical fundaments, making the possibility to enact the language on a machine hypothetical.

UML

UML was born in 1996 from the unification of the Booch, OMT and OOSE modeling methods. It is composed of a set of diagrams of which the activity diagram is used to describe processes. Different objects involved in the process can be represented in the diagram. For workflow modeling, these objects could either illustrate the invoked application by an activity or the document flow. Furthermore, different organizations can be modeled using swimlanes in the diagram (Booch et al., 1999). UML is diagrammatic in nature and some executable versions such as xUML are being developed (Kennedy Carter, 2000).

It is more difficult to determine whether UML is formal or not. On one hand, a complete formal notation for UML does not exist to this day but numerous efforts are presently made to formalize this language (Evans et al., 1999). On the other hand, many of the existing diagrams in UML are based on other formalisms such as Petri Nets or State charts for which there exists a formal semantic (Basu and Blanning, 2000). It is therefore difficult to evaluate this criteria "booleanly".

ANSI

The ANSI formalism is a purely diagrammatic formalism (ANSI, 1970). It is better known for its use in business process simulation softwares such as Optima! and is therefore executable. Swimlanes, large corridor-like partitions of the diagram, can be used to represent different organizations and document exchange between them can be represented using the adequate symbols (Rivard and Talbot, 1998). Unfortunately, the symbols used do not allow the representation of a particular IT application needed to automate an activity. Moreover, by being purely diagrammatic, it is not built on any formal basis.

SAP formalism (EPC)

The event-driven process chain (EPC) method was originally described in a paper by Keller (1992). The objectives of this formalism were to make business processes clearly interpretable by computer science neophytes before transposing them in software (Chaffey, 1998). But the EPC formalism is now better known for its use in business process reengineering with the SAP R/3 ERP system. Indeed, to assist in the remodeling of operations among the organization, SAP provides more than 800 businesses "best-practices" based on "experience, suggestion and demands of leading companies in a wide range of industries" (Curran and Ladd, 2000, p.64). The following figure presents the EPC meta-model:

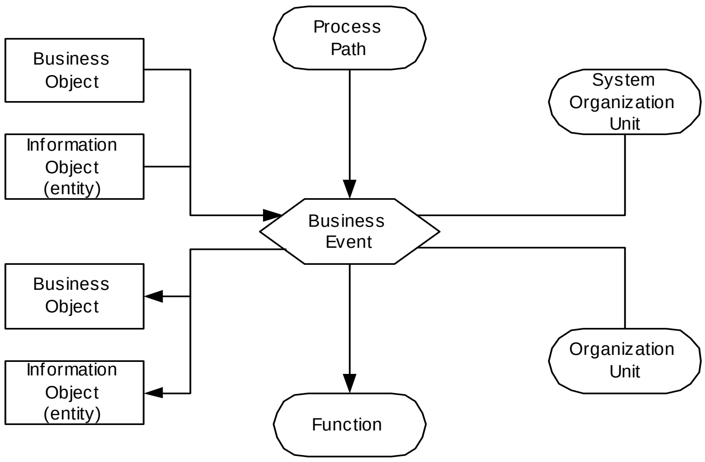


Figure 6 EPC Meta-model

The original EPC language is not formal but some efforts are made in order to provide a modified EPC based on Petri Nets in order to have a more formal semantic (Rittgen, 2000 ; Van der Aalst, 1999). The EPC was not designed in order to be executable on a software specification level but in order to be understandable by business professionals (Intellicorp, 1996). All workflow dimensions are respected as illustrated in the meta-model, but inter-organizational document exchange could hardly be illustrated. To solve this problem, SAP has developed a formalism called "business scenarios" but no detailed specifications have been published to this day.

Appendix 2: UML – Description of the three diagrams

The following figure illustrates a basic B2B workflow model. It is the process of receiving a confirmation from a seller after having placed an order. The process is partially automated: each activity triggers an IT application.

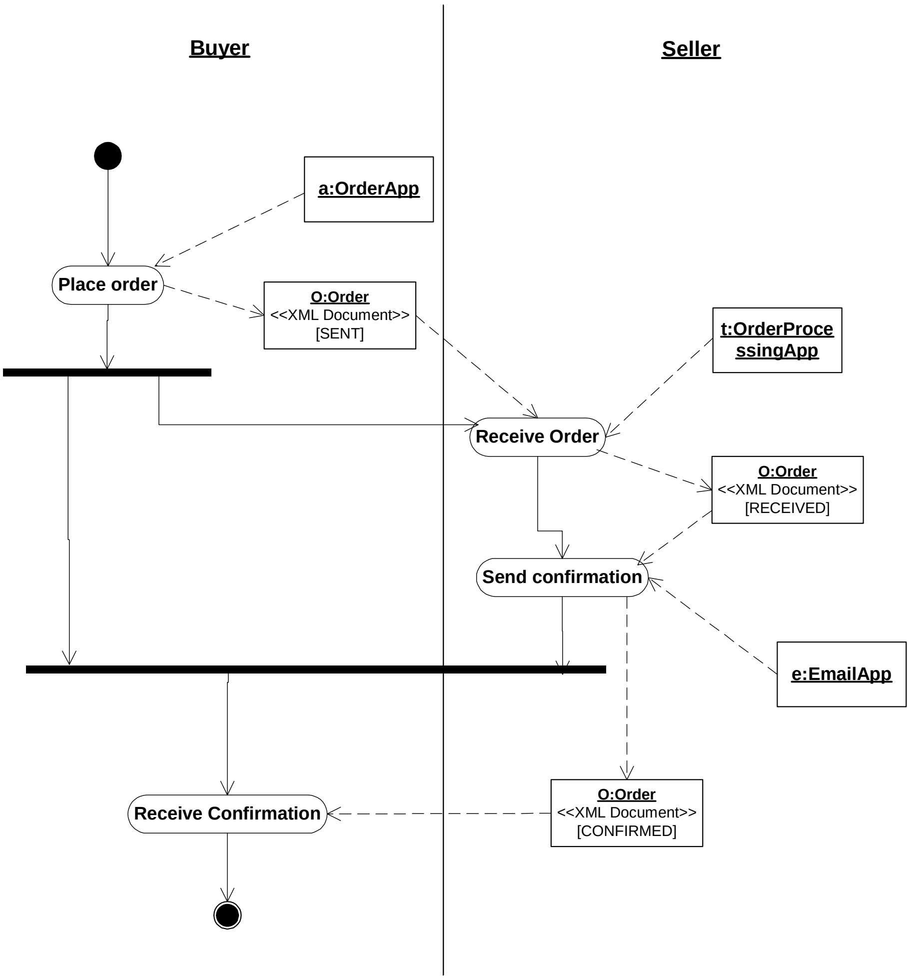


Figure 7 Workflow Model Using the Activity Diagram

The applications are represented using the objects involved for each activity. For example, a web-based order form is represented by the object a:OrderApp. The transmitted XML document is represented using the object o:Order with a stereotype defining its XML format and with a different status ([sent], [received] or [confirmed]) depending on its location in the flow. The order can therefore be represented using a state diagram as show in Figure 8.

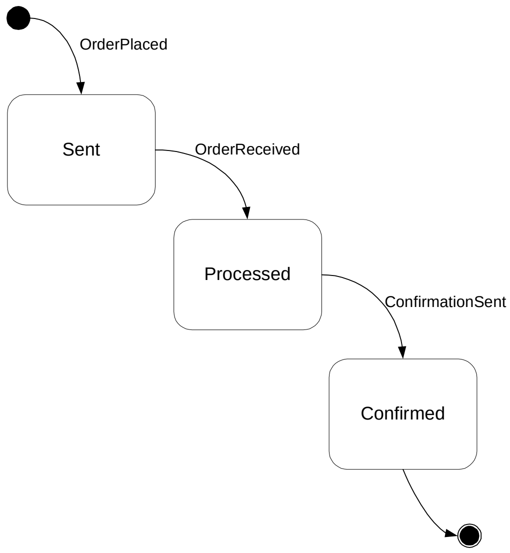


Figure 8 State Machine for the o:Order Object

The First advantage of the state machine for workflow management systems resides in its ability to represent the changing state of applications used over more than a single activity. Its other major advantage resides in its possibility to represent the control flow between activities. For example, the flow from the activity "place order" to the activity "receive order" requires the object o:Order to reach the state Sent. In more complex workflows, such a diagram offers the possibility of defining more complex transition conditions.

Figure 9 presents the last diagram, the Sequence diagram representing the flow of messages between the object Buyer and Seller.

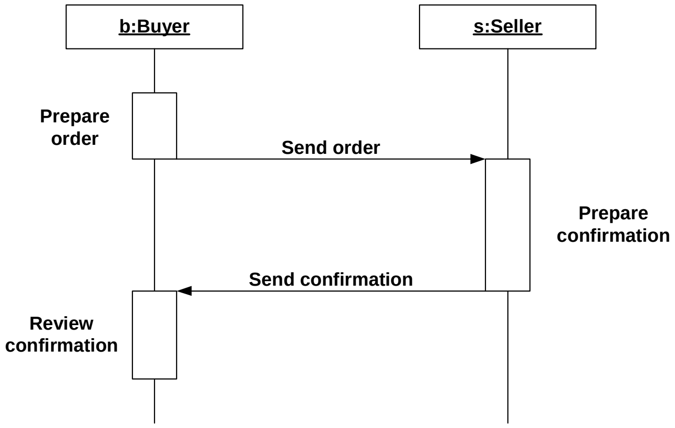
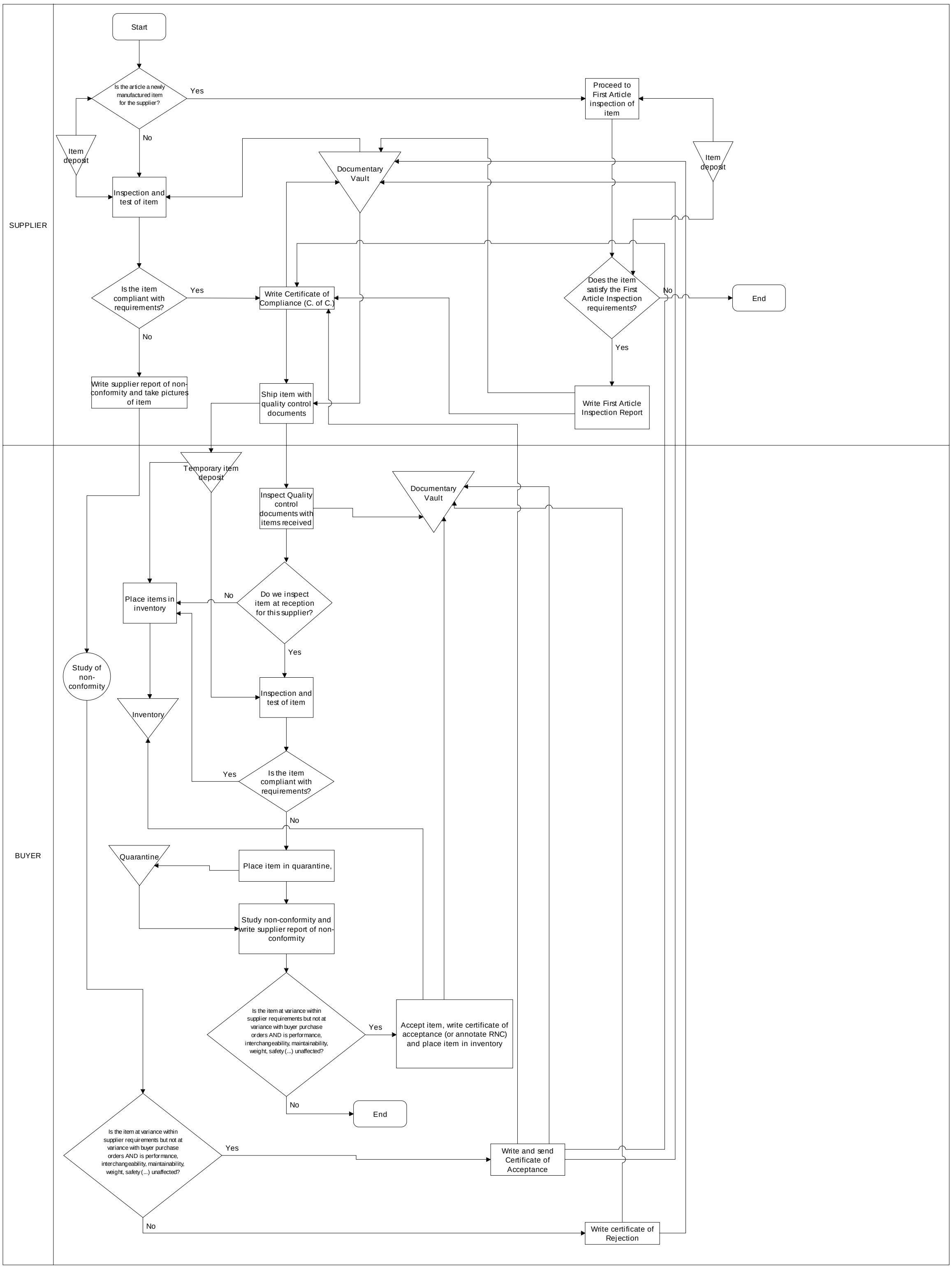
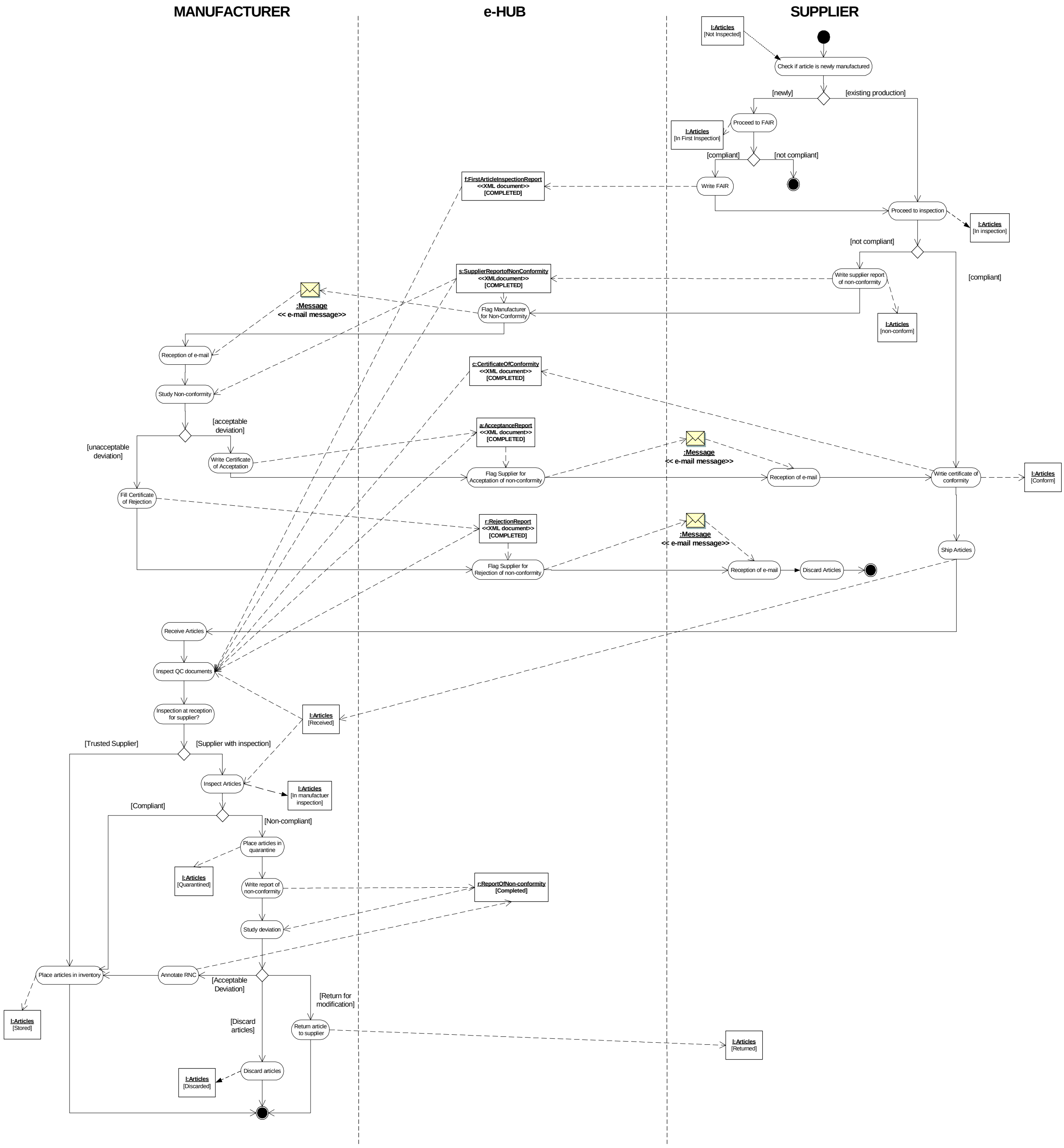


Figure 9 Interaction Between a Buyer and a Supplier Modeled Using a Sequence Diagram

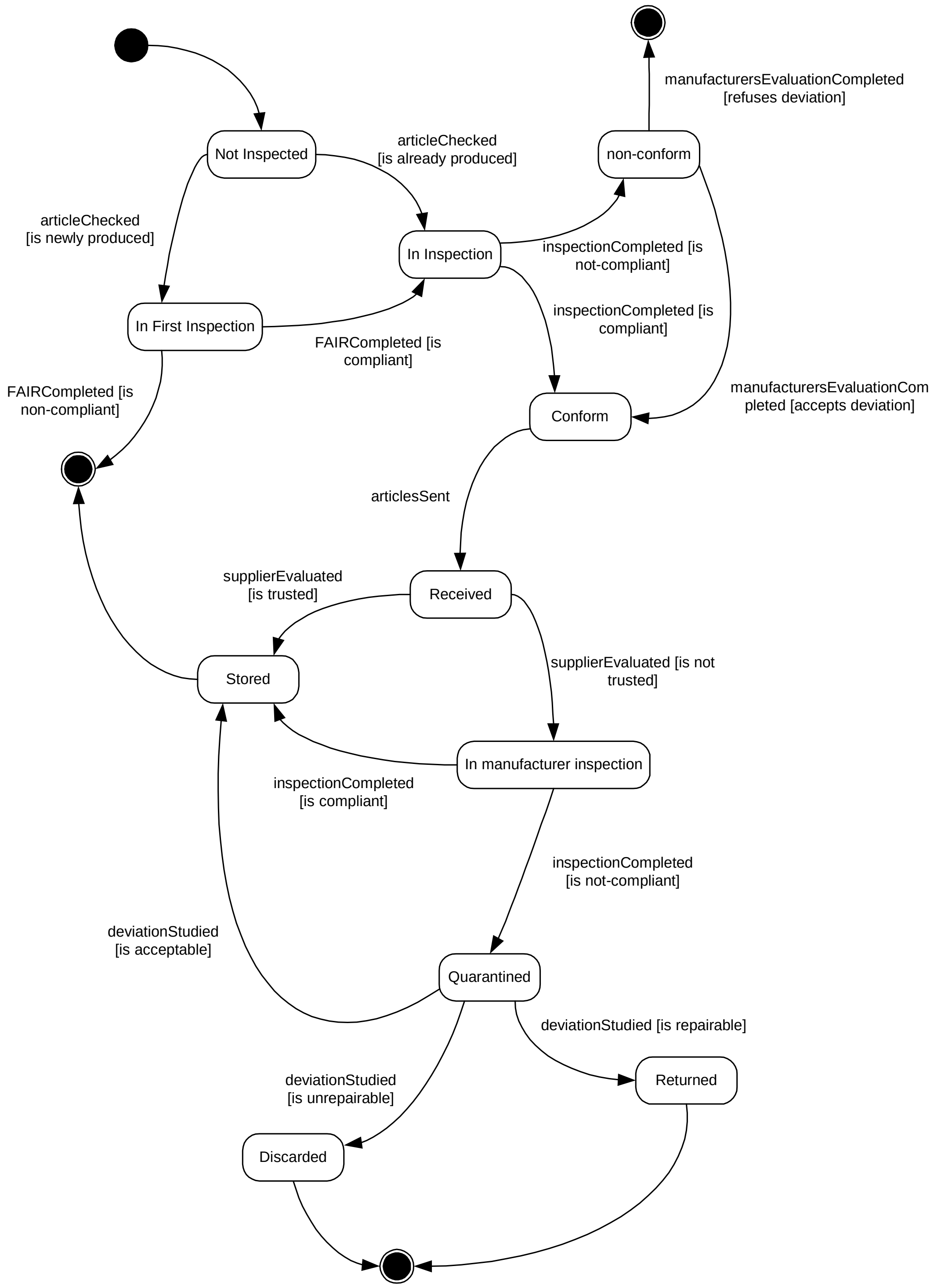
Appendix 3 Final “as is” Business Blueprint



Appendix 4 Proposed “to be” Process



Appendix 5 State Machine for the l:Articles Object



Appendix 6 Possible Interaction Diagram for the New Process