Classification and Verification of Semantic Constraints in ebXML BPSS

Jong Woo Kim[1*] and Hyoung Do Kim[2]

[1] College of Business Administration, Hanyang University, 17 Haengdang-dong, Seongdong-gu, Seoul, 133-791, South Korea, kjw@hanyang.ac.kr

[2] Department of Management Information System, Hanyang Cyber University, HIT Building, 17 Haengdang-dong, Seongdong-gu, Seoul, 133-791, South Korea, hdkim@hycu.ac.kr

Abstract

The ebXML (Electronic Business using eXtensible Markup Language) Specification Schema is to provide nominal set of specification elements necessary to specify a collaboration between business partners based on XML. As a part of ebXML Specification Schema, BPSS (Business Process Specification Schema) has been provided to support the direct specification of the set of elements required to configure a runtime system in order to execute a set of ebXML business transactions. The BPSS is available in two stand-alone representations, a UML version and an XML version. Due to the limitations of UML notations and XML syntax, however, current ebXML BPSS specification is insufficient to specify formal semantic constraints of modeling elements completely. In this study, we propose a classification schema for the BPSS semantic constraints and describe how to represent those semantic constraints formally using OCL (Object Constraint Language). As a way to verify a Business Process Specification (BPS) with the formal semantic constraint modeling, we suggest a rule-based approach to represent the formal constraints and to use the rule-based constraints specification to verify BPSs in a CLIPS prototype implementation.

Keywords: ebXML, BPSS, business process specification, semantic constraints, constraint verification

1. INTRODUCTION

ebXML (Electronic Business using eXtensible Markup Language) initiated by OASIS and UN/CEFACT is a set of specification to support modularized electronic commerce framework [ebXML 2002a; ebXML 2002b]. The goal of ebXML is to provide global electronic commerce environment based on XML regardless of size of organizations or geographic location. As a part of ebXML, Business Process Specification Schema (BPSS) provides a standard framework for business process specification. The ebXML BPSS provides the semantics, elements, and properties necessary to define business collaboration between business partners. Using the ebXML BPSS, users can extract

* Corresponding Author

and format the nominal set of element necessary to configure an ebXML runtime system in order to execute a set of ebXML business transactions. The result of business scenario definition is called an ebXML Business Process Specification (BPS) which is the input to configuration files for ebXML Business Service Interface Software. ebXML BPSS consists of two versions of Business Process Specification Schema; UML (Unified Modeling Language) version and XML version [ebXML 2002a]. The UML version of the ebXML BPSS is merely a UML Class Diagram. It is not intended for the direct creation of ebXML BPSs. Rather, it is a self-contained statement of all the specification elements and relationships required to be able to create an ebXML compliant BPS. The XML version of the ebXML BPSSs provides templates for XML-based specification of ebXML BPSs, and is the final destination to define BPSs. Thus a user may either create a BPS directly as an XML document, or may chose to use some other means of specification first and then apply transformation rules to arrive at XML version.

Though UML version of ebXML BPSS has been proposed to clarify all the specification elements and relationships required to create ebXML compliant BPS, it is insufficient to provide formal definitions of the specification elements and relationships because it provides only formal semantics that can be specified in UML Class Diagram. In the case of standard UML docunment, in order to specify UML modeling elements and their relationships formally, UML meta model is described using UML Class Diagram and OCL (Object Constraint Language) [OMG 1999; Rumbaugh et al. 1999]. Like standard UML document, it is ultimately required formal specification of semantic constraints among specification elements in ebXML BPSS using OCL. If semantic constraints in BPSS are specified formally, they can be used to verify accuracy and completeness of ebXML BPSs that are specified using ebXML BPSS.

In this paper, we propose a classification schema of semantic constraints within ebXML BPSS modeling elements and their relationships, and present formal representation of these constraints using OCL. A prototype constraint verifier using CLIPS (C Language Integrated Production System) which is a rule based language is also described briefly in this paper [CLIPS 2002]. The paper is organized as follows. Section 2 describes the review of related works including ebXML BPSS and OCL. In section 3, a classification schema of semantic constraints in BPSS is presented, and OCL-based formal specification of semantic constraints is described. Section 4 presents a BPS verification approach based on the formal representation of the semantic constraints. Section 5 includes some conclusion remarks.

2. Related Works

2.1 Business Process Specification Schema in ebXML

In ebXML BPSS, a Business Processes is a detail descriptions about how tranding partners take on roles, relationships, and responsibilities to facilitate interaction with other tranding partners is shared collaborations [ebXML 2002b]. EDIs (Electronic Data Interchanges) that have been widely used in business-to-business transactions merely concentrate on automating document processing, so they pay little attention to simplify documents and have limitations to reuse document definitions. Currently, XML-based document standardization has actively discussed, which includes xCBL[xCBL.org 2002], UBL[OASIS 2002], and OAGIS BOD[OAGI 2002]. To build more interoperable and reusable electronic commerce infrastructure, however, we need to extend the level of reuse to business scenarios (processes) not only to business documents. That is, business processes need to be modeled and reviewed together with document structures, and business transactions among business partners are executed and managed based on the modeled processes.

Systematic analysis and design are required to define business scenarios and related documents in ebXML framework. UMM (UN/CEFACT Modeling Methodology) which provides a procedural model to analysis and to define business processes and information (documents) is recommended to use in analysis and design processes [UN/CEFACT 2001]. Extending UML, UMM meta model provides all types of modeling elements which needs to be identified to analyze business processes in ebXML framework [ebXML 2001b]. As a subset of UMM, BPSS provides a set of elements to define business collaborations, which are necessary to execute business processes. There are three ways to define BPSs using BPSS. First, BPSs can be defined using business process analysis worksheets and guidelines which had been developed to support the process to define business processes and information model based on UMM [ebXML 2001b]. In this case, business process model and information model are defined using worksheets, and the part of models which are corresponding to BPSS are extracted and are registered in a repository. BPSs can also be defined using either UML or XML version of BPSS. Regardless the way to define BPSs, the final format of BPSs that will be registered in a repository is XML version. Since two different versions of a BPS, XML version and UML version can be converted mutually, in this paper, we mainly focus on constraints verification on UML version BPSs.

2.2 Object Constraint Language

Object Constraint Language (OCL) is a formal language to specify constraints within object-oriented framework [OMG 1999; Rumbaugh et al. 1999]. In UML, OCL is used to describe formally the semantics of modeling elements in UML. That is, semantic constraints that cannot be described in UML diagrams are specified using OCL. OCL can also be used to specify domain specific constraints in analysis and design phases using UML. OCL had been developed as a business modeling language in IBM since contemporary formal languages are difficult to understand without mathematical background and are not easy system analysts to use. In UML, OCL is used to following purposes; (1) to specify invariants (constraints that should be satisfied always) of classes and types in class model, (2) to define type invariants of stereotypes, (3) to describe preconditions and postconditions of operations and methods, (4) to describe guard conditions, (5) to use as a navigation description language, and (6) to specify constraints of operations.

3. Constraint Classification and OCL-based Specification

3.1 WellFormedness Rules in ebXML BPSS

Latest available version of ebXML BPSS is version 1.05 which had been published in June, 2002. In ebXML BPSS specification version 1.05, the semantic constrains of modeling elements are described descriptively. In the previous version, i.e. version 1.03, however, modeling elements are described by four parts, super class, tagged values, associations, and WellFormedness Rules (WFRs). In the version 1.03, there are 11 WFRs are associated with modeling elements. The detail description of WFRs can be found in Table 1. In the ebXML BPSS document, additional 17 WFRs are described, which is the same in version 1.03 and 1.05. The 17 WFRs are also included in Table 1 with identification numbers from 12 to 28. The identification numbers of the WFRs in Table 1 are assigned according to the sequences in the formal ebXML BPSS document by authors of this paper.

Table 1-WellFormedness Rules in ebXML BPSS

Category	WFR ID	Description of WFR	Type of constraints
MultiPartyCollaboration	1	All multiparty collaborations must be synthesized from binary collaborations	CAM
BusinessPartnerRole	2	A partner must not perform both roles in a given business activity	CAM
Performs	3	For every performs performing an AuthorizedRole there must be a Performs that performs the opposing AuthorizedRole, otherwise the MultiParty Collaboration is not complete	CIM
AuthorizedRole	4	An AuthorizedRole may not be both the requestor and the responder in a business transaction.	CAM
	5	An AuthorizedRole may not be both the initiator and the responder in a binary business transaction.	CAM
CollaborationActivity	6	A binary collaboration may not re-use itself.	CIN
DocumentEnvelop	7	A Document Envelope is associated with exactly one requesting and one responding activity.	CIE
	8	IsPositiveResponse is not a relevant parameter on a DocumentEnvelope sent by a requesting activity	CAM
Transition	9	A transition cannot enter and exit the same state	CIN
Success	10	Every Binary Collaboration should have at least one success	CIM
Failure	11	Every Binary Collaboration should have at least one failure	CIM
BusinessTransaction	12	If non-repudiation is required then the input or returned business document must be a tamper-proofed entity	CAM
	13	If authorization is required then the input business document and business signal must be an authenticated or a tamper proofed secure entity.	CAM
	14	The time to acknowledge receipt must be less than the time to acknowledge acceptance if both properties have values. timeToAcknowledgeReceipt < timeToAcknowledgeAcceptance	CAS
	15	If the time to acknowledge acceptance is null then the time to perform an activity must either be equal to or greater than the time to acknowledge receipt.	CAS
	16	The time to perform a transaction cannot be null if either the time to acknowledge receipt or the time to acknowledge acceptance is not null.	CAM
	17	If non-repudiation of receipt is required then the time to acknowledge receipt cannot be null	CAR
	18	The time to acknowledge receipt, time to acknowledge acceptance and time to perform cannot all be zero.	SAR

	19	If non-repudiation is required at the requesting business activity, then there must be a responding business document	CIM
RequestingBusinessActivity	20	There must be one input transition whose source state vertex is an initial pseudo state.	CIE
	21	There must be one output transition whose target state vertex is a final state specifying the state of the machine when the activity is successfully performed	CIE
	22	There must be one output transition whose target state vertex is a final state specifying the state of the machine when the activity is NOT successfully performed due to a process control exception	CIE
	23	There must be one output transition whose target state vertex is a final state specifying the state of the machine when the activity is NOT successfully performed due to a business process exception.	CIE
	24	There must be one output document flow from a requesting business activity that in turn is the input to a responding business activity.	CIE
	25	There must be zero or one output document flow from a responding business activity that in turn is the input to the requesting business activity	CIE
RespondingBusinessActivity	26	There must be one input transition from a document flow that in turn has one input transition from a requesting business activity	CIE
	27	There must be zero or one output transition to a document flow that in turn has an output transition to a requesting business activity.	SMA SME
BusinessCollaboration	28	A Business Partner Role cannot provide both the initiating and responding roles of the same business transaction activity	CAM

3.2 Classification Schema of Semantic Constraints

Jacinto et al. classifies constraints in XML by four types; (1) constraints on the domain of values, (2) dependencies between two elements or attributes, (3) pattern matching against regular expressions, and (4) complex constraints [Jacinto et al. 2002]. However, since most WFRs in ebXML BPSS belong to complex constraints, we classify semantic constraints more detailed like Table 2.

Table 2-Classification Schema of Semantic Constraints

Group	Subgroup	Category	Description	Checking time
Simple	Attribute value	SAT (Type)	Constraints that restrict value type of an attribute without referring other values of attributes	VE
		SAR (Range)	Constraints that restrict value range of an attribute without referring other values of attributes	VE
	Multiplicity of Association	SMM (Minimal)	Constraints that request minimal number of links without referring other values of attributes	MF, ID
		SME (Exact)	Constraints that request exact number of links without referring other values of attributes	MF, ID, IA
		SMA (Maximum)	Constraints that limit maximum number of links without referring other values of attributes	IA
Pattern matching against a regular expression		PMR	Constraint that specify that values of an attribute should follow a particular syntactic rules. Example: time information specification	VE
Complex	Attribute value	CAS (Single instance's attribute value)	Constraints that restrict the value of an attribute and need to refer other attribute values in the same object	AE

		CAM (Multiple instances' attribute value)	Constraints that restrict the value of an attribute and need to refer other attribute values in other objects	AE
	Instance cardinality	CIM (Minimum)	Constraints that request minimal number of instances of a class or an association, and need to refer values of other attributes	FE, MF, ID
		CEI (Exact)	Constraints that request exact number of instances of a class or an association, and need to refer values of other attributes	FE, MF, ID, IA
		CIA (Maximum)	Constraints that limit maximum number of instances of a class or an association, and need to refer values of other attributes	FE, IA
		CIN (Non-existential)	Constraints that request non-existence of an instance depending to values of other attributes or existence of other instances	IA, VE

* Note: VE: When the value is entered or changed; MF: When the whole modeling is finished
IA: When an instance (or link) is added; ID: When an instance (or link) is deleted
AE: After all of related values are entered or some of them changed
RE: When the reference values are entered or changed

Simple constraints are constraints which need to refer only one attribute value or the mapping cardinality of one association to verify the constraints. There are two sorts of simple constraints about attributes. The first type of constraints restricts the value type of an attribute, which is denoted SAT (Simple:AttributeValue:Type) in this paper. The second type of simple constraints restricts the value range of an attribute, which is denoted by SAR (Simple:AttributeValue:Range) in this paper. Simple constraints about the mapping cardinality of associations restrict the number of instances participant in an association which is called links in UML, which include constraints about the minimal number of links (Simple:MultiplicityofAssociation:Minimal; SMM), constraints that specify the exact number of links to be satisfied (Simple:MultiplicityofAssociation:Exact; SME), and constraints that specify the maximum numbers of links that are allowed (Simple:MultiplicityofAssociation:mAximum; SMA). The classification of 28 WFRs are presented in the fourth column in the table of Table 1.

Pattern matching constraints against regular expressions are constraints that specify that values of an attribute should follow a set of particular syntactic rules. For example, in ebXML BPSS, *time* values such as time to acknowledge receipt and time to receipt should follow ISO-8601 which is a standard to describe time and date information [UN/CEFACT 1998]. That is, receipt date and date to acknowledge receipt can be specified like "P5D" (Within five dates from now) or "19940508/P1Y6M" (During one year and six months from May 8, 1994). Time information in BPS need to be verified whether it follows the syntactic rules of ISO-8601.

Complex constraints are constraints that need to refer at least two values of attributes or instances. Complex constraints are also divided by two subgroups; constraints for attribute values and constraints for restricting the number of object instances. The complex constraints for attribute values are divided by two categories; constraints that need to refer more than or equal to two attributes values in one object instance (Complex:AttributeValue:SingleInstance, CAS) and constraints that require to refer attributes values more than or equal to two object instance (Complex:

AttributeValue:MultipleInstances; CAM). Complex constraints which restrict the number of object instances are divided by four categories; (1) constraints that specify the minimal number of instances (Complex:InstanceCardinality:Minimal; CIM), (2) constraints that requires the exact number of instances (Complex:InstanceCardinality:Exact; CIE), (3) constraints that define the maximum number of instances (Complex:InstanceCardinality:mAximum; CIA), and (4) constraints that requires no instance of an object or an association in a particular situation (Complex:InstanceCardinality:Non-existencial; CIN).

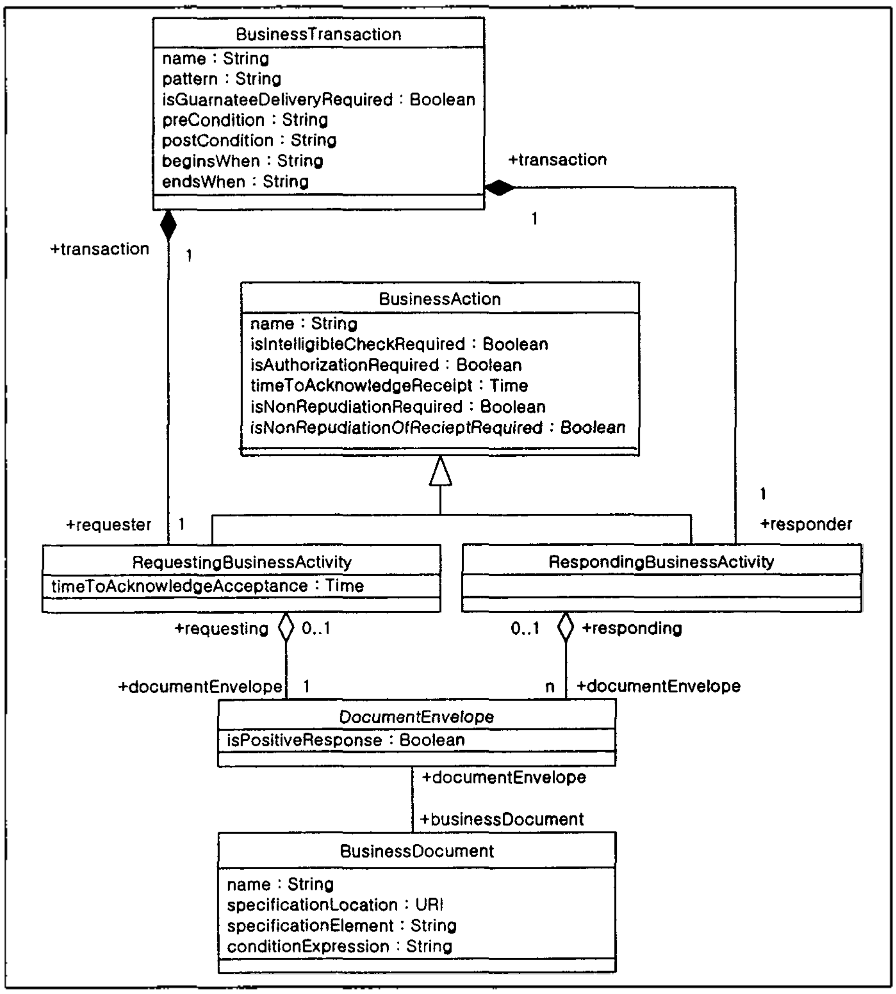

Figure 1-A partial class diagram including modeling elements in ebXML BPSS
(Source: [ebXML 2002a])

WFR 14 and WFR 28 are examples for complex constraints for attribute values. WFR 14 specifies that time to acknowledge acceptance should follow time to acknowledge receipt in class *RequestingBusinessActivity* of Figure 1. In that case, since the constraints is related with two attributes *timeToAcknowledgeAcceptance* and *timeToAcknowledgeReceipt* in the same object instance *RequestingBusessActivity*, it belongs to CAS(Complex:AttributeValue:SingleInstance) category. OCL description of WFR 14 is as follows.

[WFR 14]

content RequestingBusinessActivity **inv**:

timeToAcknowledgeAcceptance <> '' **and**

timeToAcknowledgeReceipt <> ''

implies

timeToAcknowledgeAcceptance > timeToAcknowledgeReceipt

WFR 19 is an example of complex constraints about instance cardinality, which implies that when an instance *RequestingBusinessActivity* is non-repudiation, the corresponding *RespondingBusinessAcitivity* must have at least one document (Refer Figure 1). In this case, in order to check whether the constraint is satisfied or not, we need to refer to the instances of two classes *RequestingBusinessActivity* and *RespondingBusinessAcitivity*, and an association between the two classes. Also, since the constraint restricts about the number of instances, the constraint belong to CIM (Complex:InstanceCardinality:Minimum). The description of WFR 19 using OCL is as follows. The 'size' in the following OCL statement is a predefined function in OCL, which returns the number of instances that participate in an association.

[WFR 19]

content RespondingBusinessAcitivity **inv**:

self.BusinessTransaction.RequestingBusinessActivity.isNonRepudiationRequired='true'

implies self.DocumentEnvelope->size > 0

4. Rule-based Constraint Verification

The formal representation of semantic constraints can be used for verifying BPSs that are specified based on BPSS. In this section, we describe the overall structure of a CLIPS implementation to verify BPSs. The overall structure of rule-based verifier is shown in Figure 2. Classes and associations in class diagrams in ebXML BPSS are defined as templates in CLIPS. A specific BPS which is the description of a business transaction between particular companies is represented as facts in CLIPS based on the templates of classes and associations. 'Common sense model' includes OCL predefined functions and time specification rules, which can be used to define semantic constraints in CLIPS. ebXML constraints which are specified using OCL are represented in CLIPS rule format. The following is CLIPS rules to specify WFR 14 and WFR 19. In WFR 14, the 'previous' function used in condition phrase is included in time specification rules, which compares times and returns true if the first time is previous to the second one, or false if otherwise.

```
;WFR 14
(defrule checkWFR14
   (RequestingBusinessActivity (name ?name)
      (timeToAcknowledgeReceipt ?TimeToReceipt)
      (timeToAcknowledgeAcceptance ?TimeToAcceptance))
      (previous ?TimeToReceipt ?TimeToAcceptance))
   =>
```

a semantic constraints classification schema for modeling elements of BPSS. The classification schema includes three categories, simple constraints, pattern matting against regular expressions, and complex constraints. The simple and complex constraints are refined further to 12 categories. We also show that various kinds of constraints can be expressed using OCL (Object Constraint Language) with examples. Based on the formalized constraints, we present a rule-based verifier structure using a rule based language, CLIPS to check verification of specific BPSs.

References

[CLIPS 2002] _, *CLIPS Reference Manual (Version 6.20)*, http://www.ghg.net/clips/CLIPS.html, 2002.

[ebXML 2001a] Business Process Project Team, *ebXML Business Process Specification Schema (Version 1.03)*, UN/CEFACT and OASIS, May, 2001.

[eXML 2001b] Business Process Project Team, *Business Process Analysis and Guidelines v1.0*, UN/CEFACT and OASIS, May, 2001.

[ebXML 2001c] ebXML Technical Architecture Project Team, *ebXML Technical Architecture Specification v1.0.4*, UN/CEFACT and OASIS, February, 2001.

[ebXML 2002a] Business Process Project Team, *ebXML Business Process Specification Schema (Version 1.05)*, UN/CEFACT and OASIS, June, 2002.

[ebXML 2002b] _, "Enabling a Global Electronic Market," http://www.ebxml.org, 2002.

[Jacinto et al. 2002] Jacinto, Marta Henriques, Librelotto, Giovani Rubert, Ramalho, Jose Carlos Leite, Henrigues, Pedro Randgel, "Constraint Specification Languages: Comparing XCSL, Schematron, and XML-Schemas," *Proceedings of the XML Europe 2002*, Barcelona, May 2002.

[OAGI 2002] -, *Open Applications Group*, http://www.openapplications.org/, 2002.

[OASIS 2002] _, *OASIS Universal Business Language TC*, http://www.oasis-open.org/committees/ubl/, 2002.

[OMG 1999] OMG, *OMG Unified Modeling Language Specification (Version 1.3)*, OMG, 1999.

[Rumbaugh et al. 1999] Rumbaugh, James, Jacobson, Ivar, Booch, Grady, *The Unified Modeling Language Reference Manual*, Wesley, 1999.

[UN/CEFACT 1998] _, *Numerical Representation of Dates, Time, and Periods of Time (Recommendation No. 7, TRADE/WP.4/INF.108, TD/B/FAL/INF.108)*, UN/CEFACT, 1998.

[UN/CEFACT 2001], *UN/CEFACT Modeling Methodology (UMM)*, http://www.gefeg.com/tmwg/n090r10.htm, November 2001.

[xCBL.org 2002] _, *XML Common Business Library*, http://www.xcbl.org/, 2002