Business Commitments for Dynamic E-business Solution Management: Concept and Specification

Haifei Li, Jun-jang Jeng, and Henry Chang
IBM Thomas J. Watson Research Center
1101 Kitchawan Road, Route 134
Yorktown Heights, NY 10598, USA

ABSTRACT

Nowadays, enterprises have treated e-business as an integral part of their daily business operations. How to manage a dynamic e-business solution is an important IT issue facing IT departments. We have proposed an approach to manage a dynamic e-business solution through business commitments. In this paper, we describe the concept of business commitments. The similarity and difference among business commitments, SLAs, and contracts are discussed. A language called BPCL (Business Process Commitment Language) has been proposed to declaratively model the relationships among external and internal parties. The usage of BPCL in the context of BPSM (Business Process Solution Management) is described. The purpose of BPCL is to provide a policy-based and model driven approach to business process solution management for business process monitoring and control. The specification is included as an appendix at the end of the paper.

Keywords: dynamic e-business, solution management, business commitment, business process management, business process monitoring.

1. Solution Management

With the rapid advancement of dynamic e-business technology, organizations are not longer satisfied with isolated e-business applications and have the heavy burden of application integration. Corporate customers prefer to have an industry solution that is customized for their needs and ready to be used. IT consulting institutions, like IBM Global Services, have the growing pressure to deliver domain-specific solutions on time and at a low cost. A dynamic e-business solution refers to an integrated set of applications and procedures that constitute cross-enterprise business processes such as customer relationship management (CRM) and supplier chain management (SCM). The key enabler of the next generation solution management allows customers monitor and manage their assets from the view of business processes that subsume traditional system management, application management, and network management. It provides end-to-end business process management for domain-specific solution through policies and mechanisms on core process real-time monitoring, exception healing and repair, alert and report infrastructure, process event infrastructure, monitor/configuration agent deployment, solution management dashboard, real-time business action invocation, trading partner business process commitment management, and intelligent off-line operations such as decision support for optimization of sourcing selection and supply chain inventory, and predictive/proactive business process performance optimization.

The concept of solution management evolves from the needs of coordinating the execution of multiple applications where audit trails and exceptions generated from applications must be managed from a business-level view. Therefore, the traditional notion of solution management is more or less limited to applications and business processes within a single enterprise. When the rapid development of e-business applications, there is a growing need to manage the relationship among trading partners. Traditional contract management or service level management deals with trading partners individually; therefore a global view is missing. The final result is a sub-optimal solution. There is a need to collect the relationships among trading partners and interactions among internal parties, and to manage them globally. In this paper, we introduce a new approach to tackle the challenge of building effective solution management platforms. In particular, the concept of Business Commitments will be presented and its corresponding language called BPCL (Business Process Commitment Language) will be explained. It will be shown Business Commitments and BPCL provide a path toward model-based management for dynamic e-business solutions.

There is some existing work addressing the issue of managing external relationships. WSLA [1] and tpaML [2] are two examples. However, their focuses are on one-to-one individual relationships. BPCL proposed in this paper is able to manage the one-to-many relationship. Because of the inter-dependencies among parties, one-to-many relationship is more than a simple aggregation of one-to-one relationship. Related industrial standards include ebXML BPSS [3], WSCL [4], BPML [5], WSFL [6], and XLANG [7]. [8] gives an overview of various standards related to process modeling and e-business automation. [9] has an introduction to business process management from an industry perspective.

The rest of the paper is organized as follows. Section 2 describes the concept of business commitment. Section 3 describes key business process indicator, which is introduced to isolate the monitoring and control of business process from the underlying implementation. Section 4 discusses the structure of BPCL and Section 5 discusses the usage of BPCL in Solution Management. Section 6 concludes the paper. Key part of BPCL in XML Schema is presented in Appendix.

2. Business Commitments

According to Merriam-Webster's collegiate dictionary, commitment is "an agreement or pledge to do something in the future." Business commitments are broadly defined as commitments related to businesses. Commitments can be between trading partners (called external commitments), or between internal parties within a business (called internal commitments). The definition of business commitments captures not only the current stable states ("agreement") but also the future actions ("to do something") and constraints (not to do something); therefore it is an appropriate concept to describe certain types of business relationships and interactions that may require both agreements and actions from participating parties. A set of business commitment establishes the agreement of a management platform to its customers regarding how their artifacts are to be managed in this platform. In our opinion, the concept of "business commitments" nicely fits into the business process solution management (BPSM) tools that likely manage multiple business processes and multiple BPSM platforms. BPCL formally describes the concept of Business Commitments. BPCL specifies the syntax and semantics of business commitments. BPCL can be used by a BPSM platform to configure, monitor, and control business processes based on business commitments.

2.1 Business Commitments, Contracts and SLAs

One question that naturally comes to readers' mind is what is the distinction among Business Commitments, Contracts, and Service Level Agreements (SLAs). Informally, contracts are legal documents specifying the duties and obligations of parties involved in a deal. Contract is a generic term for agreements with a legal flavor. For example, the implication of signing a contract is that "if party A breaks the contract, party B may take party A to a court". An SLA is a contract between a service requestor and a service provider that specifies the minimal acceptable levels for the service. SLA is one type of contracts with a business and quantitative (reflected in the term "level" in SLA) flavor.

Business commitment is a management agreement at the business process level with explicit actions attached to it. The concepts of business commitments, contracts, SLAs are related, but with different focus. Business commitment is the best concept for BPSM because of its focus on actions. These actions will be taken to configure, control, and monitor the business processes.

2.2. BPCL creation

Apparently, most business commitments are derived from contracts and/or SLAs. Some business commitments come from contracts and their relationships that are influenced by the perspectives of the owner of solution management platform. It implies that a solution manager should monitor and control not only the execution of an individual contract, but also the relationships among these contracts. In BPCL, the relationships among contracts are captured as *inter-contract clauses*.

Figure 1 shows the procedure for creating a BPCL. We assume that Party1 is the owner of solution management platform and it negotiates the management agreements with multiple (three in this example) parties. The result of the negotiation between Party1 and Party4 is SLA1 (assuming Party1 and Party4 have negotiated a service agreement). The results of the negotiation between Party1 and Party3, and the negotiation between Party1 and Party2 are Contract2 and Contract3 respectively (assuming these two negotiations are about general business contracts). Party1 may have its internal SLAs/Commitments that describe the obligations of various internal departments. Since SLA1, Contract2 and Contract3 are results of separate negotiations, they are fed into a process called Inter-Contract Analysis to generate possible inter-contract clauses. Inter-contract clauses are combined with internal SLAs/Commitments, SLA1, Contract2 and Contract3 (these are all from Party1's perspective) to form the BPCL. The procedures described in Figure 1 are manual. How to automate these procedures is an important topic but beyond the scope of this paper.

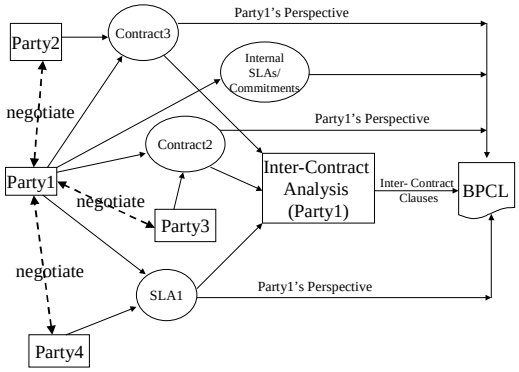

Figure 1. Steps for Creating a BPCL

3. Key Business Process Indicators

One of the major concepts embodied in BPCL is the *key business process indicator* (KBPI). A KBPI is an important business process data that manifests the status of the whole business process in business sense. KBPIs are a group of the subjects that a solution manager will monitor and control. It is obvious that not all the data generated by a business process is a KBPI. KBPIs are defined by the business analysts that are familiar with the domain of the managed business process solutions.

Key Performance Indicator (KPI) is a term used in areas like balanced scorecard, business intelligence (www.cognos.com), and supply chain management to describe the important data/parameters that can be used to measure (i.e., indicate) the performance of an enterprise. One key characteristic of key performance indicator is that an indicator must be measurable. Otherwise, it is impossible to manage it. This key characteristic is inherited by the definition of KBPI.

Once KBPIs extracted from business processes have been determined, business analysts can build useful relationships

among contracts through KBPIs. When an e-business solution involves more than two parties and more than one process, the monitoring and controlling of the relationship is vital. One hypothetical business commitment could be: "if KBPI1 from business process 1 is greater than KBPI2 from business process 2, please notify the solution owner."

4. Structure of BPCL

A BPCL document contains a set of inter-related components.

1. *Party*: party information. The descriptive information about parties participating in the solution management.

2. *BusinessProcess*: an abstract description of business processes. Its main parts are processID, description, party owning the process, participant parties, BPMType, etc. BPMType is a structure to specify the type of BPM system.

3. *KBPI*: key business process indicator. These are important parameters that indicate the status of the business process.

4. *BusinessEvent*. Events provide a triggering point for evaluating the logic expressions inside individually business commitments.

5. *BusinessCommitment*: The main parts of BC are BCIdentifier, TriggeringEvent, Validity, logic expression on KBPI, Initiator, Receiver, Action, and AltAction. Action is the action(s) to be taken when the logical expression is evaluated to be true. AltAction is the alternative action(s) to be taken when the logic expression is evaluated to be false.

5. Usage of BPCL in Solution Management

Business analysts of the solution owner compose business commitments. Once commitments have been defined, they are fed into a component called "BPSM Policy Analyzer" to generate a set of platform-independent BPSM policies for monitoring, evaluation, and actuation. In addition, solution architect defines *BPM Execution Model* based on the characteristics of the underlying platform. The execution model and policies are fed into a component called BPSM Model Analyzer to generate the management schema, which is the input to the configuration manager. The configuration manager configures various components according to the management schema. Figure 2 shows the BPCL in a solution management platform.

6. Conclusion

We have introduced the new concept of Business Commitment in this paper. BPCL is declarative in nature so that business analysts can easily describe their management commitment without concerning the low-level BPM implementation. BPCL adopts a global view of managed artifacts so that global optimization is possible. Since business commitments are formalized in the language, they can be manipulated to achieve automated deployment of management components unto BPSM platform.

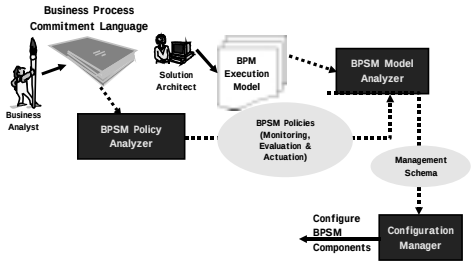

Figure 2. BPCL in a solution management platform

References

[1] Heiko Ludwig, Alexander Keller, Asit Dan, and Richard King, "A Service Level Agreement Language for Electronic Services," Proceedings of the 4th International Workshop on Advanced Issues of E-Commerce and Web-based Information Systems (WECWIS 2002), Newport Beach, CA, 2002.

[2] Asit Dan, Daniel M. Dias, Robert. Kearney, Terry C. Lau, Thao N. Nguyen, Francis N. Parr, Martin W. Sachs, and Hidayatullah H. Shaikh, "Business-to-business integration with tpaML and a business-to-business protocol framework," IBM Systems Journal Vol. 40, No. 1, 2001.

[3] ebXML BPSS, "ebXML Business Process Specification Schema," available at http://www.ebxml.org/specs/ebBPSS.doc, 2002.

[4] WSCL, "Web Services Conversation Language," available at http://www.w3.org/TR/wscl10/, 2002.

[5] BPML, "Business Process Modeling Language," available at http://www.bpmi.org/specifications.esp, 2002.

[6] WSFL, "Web Services Flow Language," available at: http://www-4.ibm.com/software/solutions/webservices/pdf/WSFL.pdf, 2002.

[7] Satish Thatte, "XLANG: Web Services for Business Process Design," available at http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm, 2002.

[8] Selim Aissi, Pallavi Malu, Krishnamurthy Srinivasan, "E-Business Process Modeling: The Next Big Step," IEEE Computer, May 2002.

[9] CSC's Research Services, "The Emergence of Business Process Management," available at http://www.bpmi.org/bpmi-downloads/CSC_RS_BPM_Report.pdf," 2002.

Appendix: Key Part of BPCL in XML Schema

```
<!-- BPCLType definition    -->
<xsd:complexType name="BPCLType">
<xsd:sequence>
<xsd:element name="Party"
type="bpcl:PartyType" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="BusinessProcess"
type="bpcl:BPType" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="KBPI"
type="bpcl:KBPIType" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="BusinessEvent"
type="bpcl:BEType" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="BC" type="bpcl:BCType"
minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>

<!-- PartyType definition -->
<xsd:complexType name="PartyType">
<xsd:sequence>
<xsd:element name="PartyIdentifier"
type="bpcl:PartyIdentifierType"
maxOccurs="unbounded"/>
<xsd:element name="Contact"
type="bpcl:ContactInformationType"/>
<xsd:element name="RolePlayer"
type="xsd:string" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="name"
type="xsd:string"/>
</xsd:complexType>

<!-- abstract business process model  -->
<xsd:complexType name="BPType">
<xsd:sequence>
<xsd:element name= "ProcessID"
type="xsd:string"/>
<xsd:element name= "Activity"
type="bpcl:ActivityType" minOccurs="0"
maxOccurs= "unbounded"/>
<xsd:element name= "ControlFlow"
type="bpcl:ControlFlowType"
minOccurs="0" maxOccurs= "unbounded"/>
<xsd:element name= "DataFlow"
type="bpcl:DataFlowType"
minOccurs="0" maxOccurs= "unbounded"/>
<xsd:element name= "Description"
type="xsd:string"/>
<xsd:element name="OverviewURL"
type="xsd:anyURI"/>
<xsd:element name="ProcessOwner"
type="xsd:string"/>
<xsd:element name="ParticipantParty"
type="xsd:string" maxOccurs="unbounded"/>
<xsd:element name="BPM"
type="bpcl:BPMType"/>
</xsd:sequence>
</xsd:complexType>

  <!-- Key Business Process Indicator -->
<xsd:complexType name="KBPIType">
<xsd:sequence>
<xsd:element name="KBPIName"
type="xsd:string"/>
<xsd:element name="KBPIType"
type="xsd:string"/>      <xsd:element
name="ProcessID" type="xsd:string"/>
<xsd:element name="KBPICategory"
type="bpcl:KBPICategoryType"
minOccurs="0" maxOccurs="unbounded"/>
<xsd:choice>
<xsd:element name= "ProcessAssociation"
type= "bpcl:ProcessAssociationType"/>
<xsd:element name="EventName"
type="xsd:string"/>
<xsd:element name="InvocationURL"
type="xsd:anyURI"/>
<xsd:element name="ObjectMethodName"
type="xsd:string"/>
<xsd:element name="QueryString"
type="xsd:string"/>
 <!-- computing the value based on other
KBPIs -->
<xsd:element name= "Computation"
type="bpcl:FunctionType"/>
<!-- deriving value for a basic KBPI -->
 <xsd:element name="ValueDerivation"
type="bpcl:ValueDerivationType"/>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>

<!--   business event -->
<xsd:complexType name="BEType">
<xsd:sequence>
<xsd:element name="EventName"
type="xsd:string"/>
<xsd:element name="EventType"
type="xsd:string"/>
<xsd:element name="ProcessID"
type="xsd:string"/>

<!-- the following choice is event
source: Sender or Timer -->
<xsd:choice>
<xsd:element name="Sender"
type="xsd:string"/>
<xsd:element name="Timer"
type="bpcl:TimerType"/>
</xsd:choice>
<xsd:element name="Receiver"
type="xsd:string" minOccurs="0"/>
```

```
<xsd:element name="EventAttributes"
type="bpcl:EventAttributesType"
minOccurs="0" maxOccurs= "unbounded"/>
</xsd:sequence>
</xsd:complexType>

<!--   Business  Commitment -->
<xsd:complexType name="BCType">
<xsd:sequence>
<xsd:element name="BCIdentifier"
type="xsd:string"/>
<xsd:element name="TriggeringEvent"
type="xsd:string"/>
<xsd:element name="CommitmentLevel"
type="bpcl:CommitmentLevelType"/>
<xsd:element name="Validity"
type="bpcl:PeriodType"/>
<xsd:element name="Expression"
type="bpcl:LogicExpressionType"/>
<xsd:element name="Initiator"
type="xsd:string"/>
<xsd:element name="Receiver"
type="xsd:string"/>
<xsd:element name="Action"
type="bpcl:ActionType"
minOccurs="0" maxOccurs= "unbounded"/>
<xsd:element name="AltAction"
type="bpcl:ActionType"  minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
```