Semantic Business Process Management: A Vision Towards Using Semantic Web Services for Business Process Management

Martin Hepp[1,2], Frank Leymann[3], John Domingue[4], Alexander Wahler[1], and Dieter Fensel[1,5]

[1]*DERI Innsbruck Innsbruck, Austria*
[2]*Florida Gulf Coast University, Ft. Myers, FL, USA*
[3]*University of Stuttgart, Stuttgart, Germany*
[4]*Knowledge Media Institute, The Open University, Walton Hall Milton Keynes, UK*
[5]*DERI Galway, Galway, Ireland*
martin.hepp@deri.org

Abstract

Business Process Management (BPM) is the approach to manage the execution of IT-supported business operations from a business expert's view rather than from a technical perspective. However, the degree of mechanization in BPM is still very limited, creating inertia in the necessary evolution and dynamics of business processes, and BPM does not provide a truly unified view on the process space of an organization.

We trace back the problem of mechanization of BPM to an ontological one, i.e. the lack of machine-accessible semantics, and argue that the modeling constructs of Semantic Web services frameworks, especially WSMO [13, 14], are a natural fit to creating such a representation. As a consequence, we propose to combine SWS and BPM and create one consolidated technology, which we call Semantic Business Process Management (SBPM).

1. Introduction

Business Process Management (BPM) is the approach to manage the execution of IT-supported business operations *from a business expert's view* rather than from a technical perspective [1, 2]. However, the degree of mechanization in BPM is still very limited, creating inertia in the necessary evolution and dynamics of business processes, and BPM does not provide a uniform representation of an organization's process space on a semantic level, which would be accessible to intelligent queries and inferences. In other words, businesses have very *incomplete knowledge of* and very *incomplete and delayed control over* their process spaces.

In this paper, we show that (1) businesses have a need for a unified view on business processes (both process models and process instances) in a machine-readable form that allows querying their process spaces by logical expressions corresponding to business semantics, (2) businesses lack such a machine-readable representation of their process space as a whole on a semantic level, and (3) the lack of such a representation is a major obstacle towards mechanization of BPM. With mechanization, we mean the reduction (not necessarily the *elimination)* of human intervention in associated tasks.

Additionally, we point out that (4) Semantic Web and Semantic Web services (SWS) technology provide suitable knowledge representation techniques. As a consequence, we (5) propose to combine SWS and BPM and create one consolidated technology, which we call Semantic Business Process Management (SBPM). SBPM aims at supporting both agile process implementation and querying the business process space by logical expressions, e.g. in order to identify activities relevant for compliance with financial or environmental regulations or in emergencies and so on.

1.1. Motivation

It has for long been common sense to first determine business requirements and then to derive IT implementations – in short, to develop software according to ideal processes as determined by managerial goals. In the early 1990s, Hammer and Champy created the term "Business Process Reengineering" [cf. 3], which brought business processes to the center of interest and lifted the subject of design from the supporting IT systems to business processes, i.e. to the perspective of business experts. However, the popularity of Business Process Reengineering did not change the underlying sequential paradigm of (1) analyzing the current state extensively, (2) creating the description of an improved state, and (3) modifying existing systems in an engineering-fashion to implement the necessary changes at a broad scope, although continuous process improvement (CPI) is beginning to pick up.

This strict sequential model of IT design in enterprises, however, has led to enormous problems, because organizations as living systems are in continuous change, which means that every requirements analysis can become partly outdated while we are working on the implementation in the next stage of the systems engineering process, and the longer the cycles take, the more a problem this becomes. This was a lesser issue when (1) the use of IT in organizations was limited (there were little "legacy" systems), when (2) market structures were more stable, and when (3) the level of integration with suppliers and customers was low. Nowadays, however, organizations are trying hard to continuously align their actual business processes, as executed by the multiplicity of systems, with the should-be processes as derived from managerial needs.

If companies are to survive in a dynamic environment, they are subject to competition in at least three dimensions (see Figure 1): Cost per process execution (y-axis), cost per process setup (z-axis), and delay of process setup (x-axis in the figure). Thus, they have to aim at getting close to the origin in this three-dimensional space. In other words, companies must be *efficient* (low cost per transaction in the operational stage), *agile* (low lag in setting up new or modified processes), and *able to evolve their process space in small movements* based on low lead costs for setting up new or modified processes.

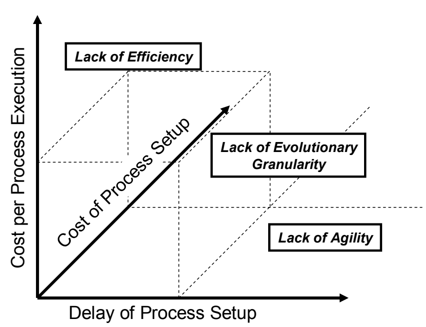

Figure 1. The three dimensions of enterprise performance from a process space view

Such enterprises that meet these requirements operate well and reside in the dashed cube in the figure. If, on the contrary, the cost per transaction is too high, there is a lack of efficiency (space above the cube); if it takes too long to set up a new process, there is a lack of agility (space right to the cube), and if the lead cost for setting up a new process is too high, the organization is unable to set up processes for small business opportunities or minor improvements (space behind the cube).

Now, although a significant part of the process is already stored in computer systems (e.g. in the form of process models, code fragments as activity implementations, data structures, data, system links, etc.), both querying and manipulating the process space regularly requires human labor. Obviously, there is a functional bottleneck between (1) the *business perspective* on operations and (2) the *actual execution* of operations on IT systems [cf. 2]. Figure 2 illustrates this IT/business divide. The upper triangle depicts the perspective of business experts and the lower triangle represents the actual implementation, which includes all computer systems and man-machine teams.

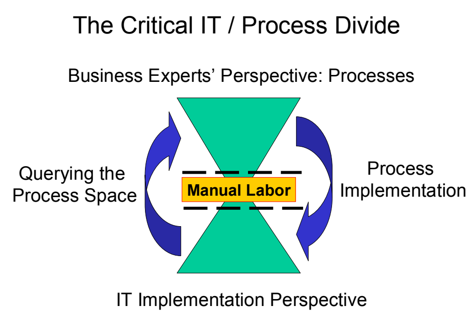

Figure 2. The bi-directional IT / process divide (derived and extended from http://www.bpmi.org)

The transition between those two spheres is very narrow, as there is no automated mediation between them. In other words, *the fundamental problem is that traversing from one sphere to the other requires manual labor in any of the two directions, i.e. both for querying and manipulating the process space:* If a manager needs to know all billing processes, systems analysts have to try to create an inventory of any such processes; and if a manager needs a new billing process for a new product or service, software engineers have to transform the management requirements into an IT implementation. This leads to the situation that business-process-related activities are, amidst a wealth of IT, surprisingly centric to human labor, and thus slow, costly, and imperfect.

This gap has been targeted by the emerging field of Business Process Management (BPM) [1, 2]. BPM aims at providing tools and techniques that support the modeling, management, and monitoring of operations on a business process level, while automatically mapping this high-level perspective to the actual implementation being executed on the multiplicity of systems. BPM modeling tools usually put a strong emphasis on the graphical representation of processes, augmented with middleware for workflow and, often, Enterprise Application Integration (EAI) functionality. In brief, BPM is a promising new area that provides a high-level perspective on business processes inside an organization. However, its current implementation does not overcome the underlying limitation that *the business process space inside the organization as a whole is not accessible on a semantic level,* especially because business process modeling languages like BPEL4WS [4] are an insufficient means of capturing and representing such a domain of discourse.

In our opinion, Business Process Management will come closer to its promise if it provides mechanization support for traversing the IT/business divide in both directions far more than it does today, e.g. for answering queries like "Can we set up a billing process that completes in less than 0.3 seconds and costs less than $0.10 per transaction?" or enacting new process instances according to a machine-readable representation of a *goal*, and not only according to representations of a *process orchestration* as in BPEL4WS. One major obstacle to this mechanization is that both the business experts' perspectives on business processes and the IT implementation sphere are *not accessible at a semantic level and thus to machine reasoning.* Overcoming this will help organizations achieve the desired effectiveness, agility, and ability to exploit small opportunities – in other words, to be located inside the target cube of Figure 1.

1.2. Approach

Semantic Web technology, namely ontology languages, repositories, reasoners, and query languages, provides scalable methods and tools for the machine-accessible representation and manipulation of knowledge. Semantic Web services (SWS) make use of Semantic Web technology to support the automated discovery, substitution, composition, and execution of software components, namely Web services. Our idea is to combine SWS and BPM, and to develop one consolidated technology, which we call Semantic Business Process Management (SBPM).

The structure of this paper is as follows: In section 2, we identify the two ways of accessing the process space of an organization as either querying or manipulating this space, and present brief use cases for those two types of actions. In section 3, we outline the idea of SBPM. Section 4 summarizes our findings.

1.3. Related Work

Our work is related to the following research directions:

Workflow management: For an overview of production workflow management including the role of business processes and their modeling, see for example [5]. [6] discusses the impact of using workflow technology on the creation of applications. [7] describes an overall environment for modeling, testing, deployment, running, and analyzing applications based on business processes (i.e. the lifecycle).

Business Process Management: The vision of BPM is outlined in [1, 2]. [8] sketches the role of business processes as an artifact in software engineering. [9] discusses the use of business processes in cross-enterprise interactions. [10] positions Web services and business processes as the basis for future application structures.

Semantic Web Services: OWL-S [11] is a comparatively narrow framework and ontology for adding semantics to Web service descriptions. WSMF is a more comprehensive framework [12], which has been further developed to the Web Service Modeling Ontology (WSMO). The core specification of WSMO can be found in [13], a brief introduction is given in [14]. WSML [15] is a family of fully-fledged ontology representation languages that supports WSMO. IRS-III [16] and the Web Service Execution Framework WSMX [18] are two reference implementations of WSMO.

SWS are currently subject to intense research, especially in the DIP project[1], and it is thus outside the scope of this paper to summarize all related work.

Business Process Modeling Languages and Standardization Initiatives: The BPM and Web services communities have yielded a wealth of languages and standardization approaches that aim at describing business processes, especially from the perspective of Web services orchestration. The most prominent examples are BPEL4WS [4], BPML [17], XLANG [19], WSCI [20], and WSFL [21]. In short, all those languages focus on the representation of a limited part of the process space, namely the patterns of message exchange (choreography) and the control

[1] http://dip.semanticweb.org

and data flow in the combination of multiple Web services (orchestration).

Mining the process space: One major challenge towards the vision of SBPM is to capture the process space automatically. There are at least two earlier works that can be built on. Reverse Business Engineering [22] is a methodology and family of toolsets that read out transactional data and program module usage in ERP systems, namely SAP R/3 and mySAP, in order to analyze the process space of an organization. Additionally, [23, 24] describe the usage of data mining technology for deriving process models form historical information, i.e. a new kind of analysis technique in the business process lifecycle.

2. The Process Space of an Organization

In this section, we discuss querying and manipulating as the two fundamental forms of accessing the process space of an organization, and present brief use case scenarios for such actions. We argue that all management tasks related to the process space of an organization can be traced back to just those two fundamental types of usage.

2.1. Querying the Process Space

In management science, decision making is a core discipline, and the main challenge for good decision making is having access to all required information. This might sound like a triviality, but in fact reveals that querying the process space is a very important task. We understand a query as (1) a machine readable representation of (2) a logical expression that (3) defines a subset of all facts in the universe of discourse (i.e., the process space) and (4) is used as request for returning all known facts, including *implicit* facts, that match this logical expression.

We envision the following examples of queries as reflecting very typical managerial information needs:

- *"List all business processes that depend on system x."*
- *"Do we have a cost approval process for items below $ 200?"*
- *"Do we have inter-organizational processes that involve company y?"*
- *"How many business transactions do we carry out with partner z on an average day?"*
- *"How many inventory management methods are currently in use?"*
- *"Can we compose a billing process model that complies with the attached specification and costs less than $ 0.1 per transaction?"*
- *"In which of our food manufacturing machines are we processing meat or raw eggs?"*

Such queries can be time-critical, e.g. in order to identify activities relevant for compliance with financial or environmental regulations or in emergencies. One can easily see that such types of queries cannot be supported by current process languages, because these were not designed to capture all relevant facts. Also, simple databases are not sufficient, because they do not include implicit information. Rather, fully-fledged knowledge representation techniques are needed. An obvious reason for this is that a huge part of the facts needed to answer such queries will be *implicit* information. For example, we might have a database of all food processing machinery and this might even contain the type of food processed, but this does not allow searching for generalizations (e.g. "Microsoft OS" as the super-category of various versions of MS Windows) or symmetrical relationships (if we know that system 1 is connected with system 2, then we implicitly know that system 2 is also connected with system 1).

The ability to answer such queries spanning the whole process space requires

(1) a machine-accessible representation of all relevant facts (concepts, instances, and axioms) on the implementation and execution level and

(2) a machine-accessible representation of the queries.

At first view, pure Semantic Web techniques, namely ontologies, repositories, and reasoners, are sufficient. In other words, it would be sufficient to "ontologize" the process space. However, a very important type of queries are in the form "can we enact / compose a process that does xyz?" This is a typical SWS discovery and composition scenario.

2.2. Manipulating the Process Space

The second form of accessing the process space is manipulating it. Examples are to create a new business process, modify an existing process, or put an outdated process out of production.

We envision the following examples of requests as reflecting very typical managerial process space manipulation needs:

- *"Compose a process model that achieves the attached goal."*
- *"Set up a billing process model by checking availability of all services that need to be invoked."*
- *"Create a billing process instance."*
- *"Replace process fragment A in all processes by process fragment B, if doable."*

- *"Execute process A every time process B is executed (completed, terminated)."*

Such functionality requires the same representations as described in section 2.1 plus at least

- the ability to actually invoke the represented functionality, e.g. via Web service calls,
- a component that can resolve a given request into an orchestration, and
- a workflow engine than can execute the resulting orchestration.

Basically, current BPM techniques and business process languages can cover part of these requirements. For example, one can define the orchestration of a business process in BPEL4WS and pass this to an execution environment which will actually enact and execute this process. However, we do not only want to enact processes for that we already have the description of the orchestration and for which we know the components (and know that they are available), but we also want to be able to describe goals and leave it to the BPM environment to figure out whether and how this can be implemented.

3. Semantic Business Process Management

Current BPM is not built on expressive, logic-based representation techniques, and thus fails at making the whole business process space accessible to intelligent queries and machine reasoning. This insufficient degree *of machine-accessible representations of the processes and data about processes* inside organizations creates the following problems:

(1) **Low degree of automation in the modeling stage:** The actual setup of business processes according to managerial needs is mainly done manually, often involving numerous consultants.

(2) **Modeling delay:** The dynamic composition or modification of business processes is labor-intensive, increasing the time to market and reducing an organization's agility.

(3) **Cognitively inadequate complexity:** The lack of a clear separation between business goals and implementation details makes the management of business processes overly complex.

(4) **Process blindness:** Managers and other business experts cannot quickly determine whether a specific process can be composed out of existing basic activities, nor can those stakeholders query the process space within their organization by logical expressions. Thus, checks for process feasibility (e.g. prior to the launch of new products or services) or compliance (e.g. Sarbanes-Oxley, ISO, etc.) are still to be done manually by business analysts.

(5) **Intransparent conflicts and interdependencies:** It is impossible to use machine reasoning in order to identify potential side-effects of modifications.

As a consequence, we propose to combine SWS and BPM and yield one consolidated technology, which we call Semantic Business Process Management (SBPM). SBPM takes the following approach: We

(1) represent and semantically describe each process and its encompassed basic activities inside an organization as a SWS in a process repository;

(2) capture the complete IT landscape (e.g. hardware, operating systems, manufacturing equipment) in the form of an ontology, because this will be necessary for checking whether a modification of the process space (e.g. activation of a new program component) is compatible with the available infrastructure (e.g. available amount of storage space);

(3) gather domain knowledge (e.g. technical constraints, business rules) and store it in the form of axioms formulated in a rule language (which can be part of the ontology language);

(4) map transactional data from the various systems (e.g. ERP) to a virtual instance store in the sense that this data is accessible to reasoning tasks without the need to replicate it;

(5) express queries in an ontology query language;

(6) model business experts' needs as goals, and

(7) use a SWS execution environment for the mediation between business goals and queries, and the actual process space.

This fundamental idea is shown in Figure 3.

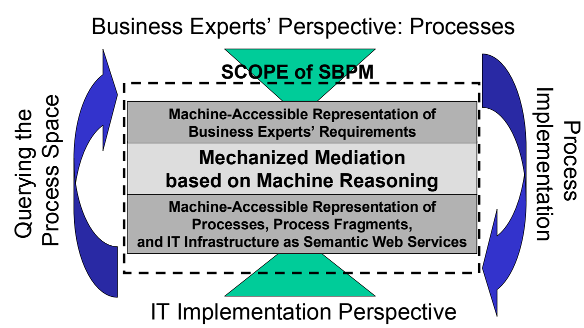

Figure 3. Semantic Business Process Management

We are currently in the process of creating a WSMO/WSMX-based use case and proof of concept for the telecommunications industry, and are working on a comprehensive stack of standards for SBPM.

4. Conclusion

We have shown that businesses lack a machine-readable representation of their process space as a whole on a semantic level, while they would benefit from such a unified view on business processes in a form that allows querying their process spaces by logical expressions. We also argued that the lack of such a representation is a major obstacle towards mechanization of BPM, and that Semantic Web and Semantic Web Services (SWS) technology provide suitable knowledge representation techniques. As a second step we proposed to combine SWS and BPM and create one consolidated technology, which we call Semantic Business Process Management (SBPM); described the required components and architecture for SBPM, and outlined how this architecture will allow mechanized mediation of the IT / business divide and will thus support both agile process implementation and querying the business process space by logical expressions, e.g. in order to identify activities relevant for compliance with financial or environmental regulations or in emergencies.

Acknowledgements: The work presented in this paper is partly funded by the European Commission under the project DIP (FP6 - 507483).

References

[1] H. Smith, P. Fingar: "Business Process Management. The Third Wave". Meghan-Kiffer Press, Tampa, FL, USA 2002.

[2] Business Process Management Initiative, http://www.bpmi.org.

[3] M. Hammer, J. Champy: "Reengineering the Corporation". Nicholas Brealey Publishing, 2001.

[4] T. Andrews et al.: "Business Process Execution Language for Web Services Version 1.1". Available at http://www.siebel.com/bpel.

[5] F. Leymann, D. Roller. "Production Workflow - Concepts and Techniques". PTR Prentice Hall, 2000.

[6] F. Leymann: "The workflow-based application paradigm". In: Proc. of the Workshop on Workflow Managment - State of the Art, Münster, Germany, April 10, 1996.

[7] F. Leymann, D. Roller: "Workflow based applications". IBM Systems Journal 36(1) (1997), pp. 102-123.

[8] F. Leymann: "On the interrelationship of workflow technology and other software technologies". In: Proc. of the 5th Intl. Conf. on the Software Process ICSP5 (ACM SIGSOFT), Chicago, Illinois, USA, June 15-17, 1998.

[9] F. Leymann, D. Roller, M.-T. Schmidt: "Flows and Web Services: B2B aspects of business process management". IBM Systems Journal 41(2) (2002).

[10] F. Leymann: "The Influence of Web Services on Software: Potentials and Tasks". In: Proc. of the 34th Annual Meeting of the German Computer Society, Ulm, Germany, September 20-24, 2004, Springer, 2004.

[11] OWL-S 1.0 Release, http://www.daml.org/services/owl-s/1.0/.

[12] D. Fensel, C. Bussler: "The Web Service Modeling Framework WSMF". Electronic Commerce Research and Applications, 1 (2) 2002.

[13] D. Roman et al.:"D2v1.2 Web Service Modeling Ontology (WSMO)". WSMO Final Draft April 13, 2005. Available at http://www.wsmo.org/TR/d2/v1.2/20050413/.

[14] C. Feier, J. Domingue: "D3.1v0.l2 WSMO Primer". WSMO Working Draft April 1, 2005. Available at http://www.wsmo.org/TR/d3/d3.1/v0.2/20050401/.

[15] J. de Bruijn et al.: "D16.1v0.2 The Web Service Modeling Language WSML". WSML Final Draft March 20, 2005. Available at http://www.wsmo.org/TR/d16/d16.1/v0.2/20050320/.

[16] J. Domingue, L. Cabral, F. Hakimpour, D. Sell, and E. Motta (2004): "IRS-III: A Platform and Infrastructure for Creating WSMO-based Semantic Web Services". In: Proceedings of the Workshop on WSMO Implementations (WIW 2004) Frankfurt, Germany, September 29-30, 2004, CEUR Workshop Proceedings, ISSN 1613-0073.

[17] A. Arkin: "Business Process Modeling Language". Available at http://www.bpmi.org/downloads/BPML1.0.zip.

[18] Web Service Execution Environment (WSMX). Available at: http://www.wsmx.org/

[19] S. Thatte: "XLANG. Web Services for Business Process Design". Available at http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm.

[20] W3C: "Web Service Choreography Interface (WSCI) 1.0", W3C Note 8 August 2002. Available at http://www.w3.org/TR/wsci/.

[21] F. Leymann: "Web Services Flow Language WSFL". Available at http://www.ibm.com/software/solutions/webservices/resources.html.

[22] IBIS Prof. Thome AG: „RBE Plus". Available at http://www.rbe-online.de.

[23] W.M.P van der Aalst, B.F. van Dongen, J. Herbst, L. Maruster, G. Schimm, A.J.M.M Weijters: "Workflow mining: A survey of issues and approaches". DKE 47 (2003).

[24] R. Agrawal, D. Gunopulos, F. Leymann: "Mining process models from workflow logs". In: Proc. of the Intl. Conf. on Extending Database Technology EDBT'98, Valencia, Spain, March 3-8, 1998.