# SemBP
# Semantic Technologies for Business Process Management

(Research report of M.Missikoff)


## Abstract

The proposed research program aimed at investigating methods, infrastructures, and tools dedicated to the semantic enrichment of Business Processes Management (BPM); all BPM phases are considered: design, integration, enactment, interoperability, monitoring, evolution, verification.
The semantic enrichment of Business Process (BP) modeling tools requires the development of a BP ontology (BPOnto). BPOnto contains the conceptual base (concepts hierarchy, relationships, and axioms) that refer to BP components, to be modelled and enacted.
The addressed problem area is very broad; it ranges from process knowledge extraction to process components identification, from BPOnto modeling to BP diagramming, and techniques to realise a bridge between traditional BP modeling tools and the SemBP infrastructure.
This SemBP approach can greatly support BP reuse, BP components storage (e.g., sub-processes and web services), semantic annotation, and advanced semantic search. This latter activity guarantees that the retrieved components are semantically compatible with the requirements, then, the requirements can be effectively used by the hosting BP (let alone technical issues, such as interface reconciliation).
It is also clear that the SemBP approach appears particularly suitable to support business people in defining the BP, and in the progressive transformations of the latter, towards a rigorous and, eventually, executable specification. This means that the SemBP approach aims at solving a critical issue: the Business/IT-Alignment problem represented by the alignment of business requirements, generated by the business people, and the technological artefacts (e.g., software applications) developed by IT people, starting form such requirements.
The final outcome will be in the direction of creating a SemBP infrastructure that will support (and be hidden by) a traditional BP modeling tool (such as Intalio BPMS Designer or uEngine), in order to allow business people to transparently use semantic technology solutions.


## 1. Introduction

Traditionally, Business Process Management (BPM) has been perceived as divided into two distinct areas[1]:

- **BPM as a discipline** of business management, supporting business organizations aiming at the standardization and the continuous optimization of operational processes, obtaining a large impact on achieving corporate performance goals;
- **BPM as a technology** of software production, providing IT organizations with a set of tools to model, deploy and execute processes that include human and system tasks (e.g., workflows) or that span across different business applications and require a broad set of integration capabilities (e.g., messaging, transformation, adapter technology); the current commercial state of the art is well-known as Enterprise Application Integration (EAI).

Both cases represent challenging activities that require highly skilled experts. Furthermore, today in the two areas (business and IT) the experts operate without a systematic interaction and cooperation, causing the well-known problem of Business/IT misalignment. One of the causes is that the two communities (business and IT experts) tend to use different, often incompatible tools and methods, sometimes even speaking different jargons. We believe that an ontological approach to BP modeling will significantly support the resolution of the above problems, by providing a shared, "neutral" representation to be used by the two communities.

Semantic enrichment methods aim at connecting an intuitive BP diagram to a semantically formal ontology. It is possible to apply semantic technologies to a number of tasks related to BP modeling. Below we list a few of these.

- Analysis of a business scenario, for a better understanding, for exchanging knowledge with other humans, for improving the way the organization operates (e.g., by using BP reengineering - BPR);
- Building a repository of BP models (or fragments of them) for future reuse;
- Support to the design of enterprise BPs, also by reusing the resources collected in the BP models repositories;
- Search & retrieval of BPs (or parts of them) to support the previous task;
- Search & retrieval of e-services, necessary to a full operational BP;
- Automatic execution of enterprise BPs, having identified and integrated the required e-services;
- Consistency checking of a BP model;
- Systematic BP reorganization and optimization (e.g., by using simulation and BP log mining and intelligence);

[1] https://www.sdn.sap.com/irj/sdn/weblogs?blog=/pub/wlg/8094

- Support to runtime monitoring of BP execution;
- Support to the integration of different local BPs in the context of a virtual enterprise (CBP: Collaborative BP, built from local BPs).

## 2. New trends in BP Management

Business process modeling is opening a new phase in the development of enterprise software applications (ESA), thanks to the recent proposals of standardization, such as BPMN (Business Process Modeling Notation) [0], the growing importance of Business Rules, with the proposal of SBVR (Semantic Business Vocabulary and Rules), and the emerging proposal of the Model Driven Architecture (MDA). Along the three lines a key role is played by OMG [0].

In parallel, semantic technologies are addressing the modeling of behavioral aspects. Proposals like WSMO [17] and OWL-S [16] are attracting the attention of researchers. However, such proposals have been conceived within a "replacement" rather than as a "subsidiary" perspective. In fact, the proposed semantic solutions tend to cover all the BP lifecycle, from the initial design, to the refinement, the discovery and composition of services, until the actual execution. In this research we prefer to investigate the second perspective, where semantic technologies are placed side by side to business oriented solutions, such as the above mentioned OMG proposals.

Coming back to the MDA paradigm, it proposes 3 levels where BP models play a central role:

- Computational independent models (CIM), where business experts provide a first (informal) description of a BP (e.g. by using EPC [0]);
- Platform independent models (PIM), where business experts work together with IT experts to build a procedural specification of the process, in a rigorous way (e.g., by using BPMN), but leaving out low level technical details;
- Platform specific models (PSM), developed by IT experts, where all the technical details are introduced to achieve a complete specification, (e.g., by using BPEL4WS [4]), ready to be executed by a suitable engine (e.g., ActiveBPEL2) [0], invoking the required services.

The OMG-MDA proposal has been mainly conceived to support a layered development of enterprise software applications, however for what concerns BP lifecycle, and in particular BP development and evolution, it does not propose any specific approach. In a dynamic enterprise, BPs need to be periodically revised and updated. Such BP evolution may be necessary for different reasons, for instance: poor functional performances (e.g., a process takes too much time or fails to fully achieve what expected), poor non-functional performances (e.g., security and privacy are not sufficiently guaranteed), new company policies (requiring the update of non conformant BPs).

## 3. State of the Art

The literature reports a large number of relevant results in the areas of process modeling, rules representation and management and, in particular, results addressing the analysis of activities sequencing. However, we wish to point out that we avoided sophisticated approaches, such as those based on temporal algebra (e.g., based on the Allen's work [25]) or temporal logic [26], and its variations (such as Event Calculus [27] and Situation Calculus [28].) Instead, here we approached the analysis of BP and the sequencing of their activities by starting with the simple notion of precedence (e.g., requesting_quote precedes issuing_order). Another important element is represented by branching and decision points in the workflow. Here we intend to start by considering the nature of the branching, namely: *and*, *or*, *xor*. The conditional expression determining the truth can be considered in a second phase. The simplified approach assumed in this work is already sufficiently expressive to produce a few interesting results and, at the same time, leaves space for future work where we plan to adopt a richer modeling paradigm.

As anticipated, in the last period there has been an intense research activity on methods for process ontologies. Well-known proposals are OWL-S and WSMO. But those are more seen as comprehensive frameworks for service-oriented business ontologies. More focused proposals, in the area of ontology-based BP, are OWL-T, OWL-P, and oXPDL.

OWL-T [6] (T stands for Task) is a method for coding an ontology in OWL, expressing user demands (tasks) at a high-level of abstraction, without dealing with the technical details of the underlying infrastructure. The OWL-T meta-model is characterized by a hierarchy of task types: Atomic, Composite, Simple, Complex. Each task is described in terms of properties and components. In particular, functional properties allow inputs, outputs, preconditions, post-conditions, preferences and effects to be represented.

---

[2] www.activebpel.com

For its characterization, OWL-T is particularly suitable for the generation of executable processes. The OWL-T developers aim to transform tasks into executable processes by employing some automatic methods of service composition. Our objective is rather to support the design and progressive refinement phases.

OWL-P [7] (where P stands for Process) proposes an approach for business process modeling and enactment, based on a combination of protocols and policies. The key idea is to capture meaningful interactions as protocols. OWL-P is an ontology framework for protocols coded in OWL. OWL-P describes concepts such as roles, messages exchanged between the roles, and declarative protocol rules. The main computational aspects of protocols are specified using the Semantic Web Rule language (SWRL) [8] for defining rules which allows implication rules over entities defined as OWL-P instances to be specified.
OWL-P will be carefully analysed, but the preliminary impression is that this approach requires a significant amount of details even in the preliminary design steps. In contrast, our objective is characterised by an iterative refinement process, where the first phases require simple specifications, in the form of BP sketches.

oXPDL [9] is a process interchange ontology formalism based on the standardized XML Process Definition Language (XPDL) [10]. oXPDL explicitly represents the semantics of a process model defined according to XPDL in a Web ontology language. oXPDL also focuses on reusing and integration of existing standard and ontologies such as SUMO [11], eClassOWL [12], RosettaNet [13] and PSL [14]. This proposal is very interesting and will be also carefully studied. Again, our idea is to start with a simpler framework, tightly connected with a traditional, BP diagrammatic environment, to provide an effective semantic annotation.

## 4. Objectives of the research

The research developed during the visit at the University of Paris, Sorbonne, had the objective of carrying on the study of BP ontologies and the possibility of a semantic enrichment of BP models developed by business people, by using diagramming tools (e.g., Intalio BPMS Designer.) To this end, the core of our investigation concerns the ontological primitives useful to build a BPOnto and the methods and tools for their management and evolution.

An Ontology is defined as *"a formal, explicit specification of a shared conceptualisation"* [1], or alternatively as follows *"an ontology can range in expressivity from a Taxonomy (knowledge with minimal hierarchy or a parent/child structure), to a Thesaurus (words and synonyms), to a Conceptual Model (with more complex knowledge), to a Logical Theory (with very rich, complex, consistent and meaningful knowledge)"* [18].

These definitions characterise ontology and in particular they underline that ontology is:

- The result of a consensus reached among a group of domain experts (shared conceptualisation) and, more in general, of a wider community (e.g., by using new emerging methodologies, such as the folksonomy)
- A formal specification, and as such it can be interpreted by a machine for reasoning and querying activities
- The result of a stepwise approach with intermediate results (i.e., Taxonomies, Thesaurus).

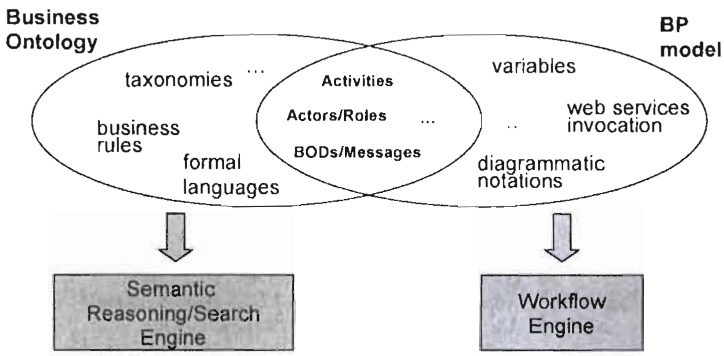


**Figure 1 – Relationships between a Business Ontology and a Business Process.**

On the other hand, "a business process or business method is a collection of interrelated tasks, which solve a particular issue." [3]. Therefore, one of the main efforts will also be the actual modeling of the ontological content of BPs.

These definitions highlight the different nature of the two basic approaches of our investigation: the diagrammatic approach in the bisiness context and the ontological modeling in the semantic context of BPs. Both approaches model a certain fragment of the reality, but if an ontology is descriptive, a BP has more a prescriptive and operational mission, since it aims, eventually, at the generation of a workflow and an execution form of tasks that compose it.
They can describe the same piece of reality from different perspectives and for different purposes but indeed with a certain degree of overlapping (see Figure 1). Consequently, it is useful to think about a certain level of interaction between BP diagrams and a BPOnto.

## 5. Research activities

The research activities have focused on two major areas.

### 5.1. Methods and tools for BP ontologies.

We have started from the existing results achieved in BP ontologies, and in particular from the first proposal of the BP ontology framework BPAL [11]. BPAL is an ontology modeling framework that allows a predicative specification of a BP Schema to be formulated. Below we report a sketchy introduction to BPAL[3]. It proposes to organize the ontological knowledge according to a few basic categories, listed below.

- Activities: *acv*;
- Precedence relations: *prec*;
- Decisions: *dec*;
- Messages: *msg*;
- Actors: *acr*;
- Events: *evt*.

Note that decision atoms can be specialized to AND, OR or XOR type: *adec, odec, xdec*. In BPAL we distinguish a BP Schema (BPS), which is a set of predicative atoms, from a BP Instance (BPI) represented by a chain of ground terms. A BPI originates from the actual execution of a BP Schema and each element of the chain (activity instance) corresponds to an execution of an activity (variable) in BPS.

Along this line, the research activity intends to systematise the original BPAL proposal and to test it in a trial case.

### 5.2. Scenarios for a semantically enriched BP modeling environment

The second research line is more on the technology side, aimed at exploring the feasibility of a Semantic Bridge, to be realised between the two BP management environments of interest: diagrammatic and ontological. There are several approaches in tackling the coupling of the two. Below we sketchily outline some different scenarios, with different levels of interaction, requiring alternative architectures, where cooperation between a diagrammatic BP modeling tool and an Ontology Management System can occur.

**Scenario 1 – Independent development**

The diagrammatic model and the ontology of a process are developed independently by two different groups of experts with different methodological and technical backgrounds. No semantic constraints coming from the ontology are imposed during for modeling the BP.
For an interaction between the two artefacts a bridge between them need to be conceived and developed. Due to descriptive nature of the ontology and its characterization as a common reference the Semantic Annotation of the BP against the Ontology is indicated as the way to connect the two tools.
The annotation can be done at least in two different ways:

- a *manual* annotation done by domain experts who is in charge to identify and trace the conceptual correspondences
- with an *automatic* support through Automatic Mapping Discovery techniques. Such an approach is ideal but with the current technologies its efficiency is limited even under good conditions (i.e.,

[3] A complete specification of the BPAL ontology framework falls outside the scope of this paper, can be found in [0].

syntactically aligned structures, limited application domain). On these premises, a human being validation of the automatic matching should be finally needed.

Once the annotation has been performed, manually or with an automatic support, the semantic alignment of the BP respect to the Reference Ontology can be performed producing a *Semantic Diagnosis* that can be generated, yielding a feedback for improving the BP modeling.

**Scenario 2 – Development of a Semantic BP**
This scenario requires that the modeling of the BP is done taking the Ontology as a main reference. When building the BP diagram, only entities that have been previously defined in the ontology can be used.
Also in this scenario, we have two alternatives that depend on the technological integration of the BPOnto system and the BP modeling tool:

- A system integration between the BPOnto system and the BP modeling tool exists (e.g., a BPOnto window can be opened in the BPD tool). In this case, the alignment between the two artefacts is guaranteed during the construction of the BP (*interpreted approach*);
- The two environments are independent and they are not able to share their contents. The user models the BP taking the Ontology as reference, but no check can be performed during the modeling phase. In this case, the alignment have to be performed after the construction of the BP (*compiled approach*)

In this scenario, the quality of the BP strongly depends on the quality of the ontology.

**Scenario 3 – Full BP Ontology**
This scenario assumes that the ontology contains all the knowledge of a BP diagram. It is also assumed that the BP modeling tool is able to import the content of the BPOnto management system. In this case, the BP modeling tool has just the role of a graphical viewer, since the content is fully defined within the ontology tool. Of course, any modifications in the BP model should be first reported in the Ontology and then viewed in the BPD tool.

**Scenario 4 – Diagrammatic Ontology**
This scenario is the opposite approach with respect to the previous one. It is here assumed that the user interacts only with the diagrammatic tool for BP modeling and the export of this tool is imported by the BPOnto system. In this case, the primary interface is that of the BPD tool and the ontology is simply a back-end system. The ontology cannot be managed directly, therefore misalignment between the two cannot arise. If the ontology reasoner reveals an inconsistency, it can be corrected only through the BPD tool.

**Scenario 5 – Full Integration**
In this scenario the approaches of the two previous scenarios are joined in order to have a complete symmetry between the two modeling environments, with the possibility to model any process from two different perspectives at the same time. The two environment are tightly integrated and any update on one side is checked and reported on the other side.

The scenarios above described represent a first group of architectural configurations that one may encounter in building a semantically enriched BP modeling environment. In a future research we intend to explore a few of the reported architecture to better understand pros and cons.

## 6. Bibliography


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

**[2]** Object Management Group. Business Process Modeling Notation Specification. Version 1.0. February 2006 [www.bpmn.org/Documents/OMG%20Final%20Adopted %20BPMN%2010%20Spec%2006-02-01.pdf].

**[3]** A. De Nicola, M. Lezoche, and M. Missikoff. An Ontological Approach to Business Process Modeling. In: Proceedings of the 3rd Indian Int. Conference on Artificial Intelligence (IICAI), pp. 1794--1813. (2007).

**[4]** Intalio Business Process Management Suite [http://bpms.intalio.com/].

**[5]** F. D'Antonio, M. Missikoff, and F. Taglino. Formalizing the OPAL e-Business ontology design patterns with OWL. I-ESA Conference 2007.

**[6]** V. X. Tran and H. Tsuji. OWL-T: A Task Ontology Language for Automatic Service Composition, ICWS 07, pages 1164-1167, IEEE Computer Society.

**[7]** N. Desai, A.U. Mallya, A.K. Chopra, and M.P. Singh. OWL-P: A methodology for Business Process development. In proceeding of AOIS 2005. pp 79-94.

**[8]** I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. SWRL: A semantic web rule language combining OWL and RuleML, May, 2004 (W3C Submission).

**[9]** A. Haller, M. Marmolowski, E. Oren, and W. Gaaloul. oXPDL: a Process Model Exchange Ontology. Technical report at DERI-TR-2007-12-12, 2007.

**[10]** The Workflow Management Coalition. Workflow Standard Process Definition Interface – XML Process Definition Language. 2005.

**[11]** I. Niles and A. Pease. Towards a standard upper ontology. In Proceedings of the International Conference on Formal Ontology in Information Systems, pp. 2–9. 2001.

**[12]** M. Hepp. Products and services ontologies: A methodology for deriving owl ontologies from industrial categorization standards. International Journal on Semantic Web & Information Systems, 2(1):72–99, 2006.

**[13]** A. Haller, J. Gontarczyk, and P. Kotinurmi. Towards a complete SCM Ontology - The Case of ontologising RosettaNet. In Proceedings of the 23rd ACM Symposium on Applied Computing. ACM, Fortaleza, Brazil, 2008.

**[14]** M. Gruninger. Ontology of the process specification language. In S. Staab and R. Studer, (eds.) Handbook on Ontologies, pp. 575–592. Springer, 2004.

**[15]** OWL. Web ontology language, Feb 2004. http://www.w3.org/TR/owl-features/.

**[16]** OWL-S (Web Ontology Language for Web Service) - http://www.daml.org/services/owl-s/1.0/.

**[17]** D. Roman, H. Lausen, and U. Keller. D2v1.3. Web Service Modeling Ontology (WSMO). Deliverable, http://www.wsmo.org/TR/d2/v1.3/, October 2006.

**[18]** DSTC, IBM, Sandpiper Software. Ontology Definition Meta-model, Revised submission to OMG, 10 January 2005; http://www.omg.org/docs/ad/05-01-01.pdf.

**[19]** Soley, R., et al. Model Driven Architecture, OMG, 2000.

**[20]** Van der Aalst, W.M.P. Formalization and Verification of Event-driven Process Chains, Information & Software Technology 41, 639--650 (1999).

**[21]** White, S.A.. Introduction to BPMN, IBM, 2004.

**[22]** IBM Inc., Microsoft Corp., BEA Inc., SAP AG, and Siebel Systems Inc. The Business Process Execution Language for Web Services. http://www128.ibm.com/developerworks/library/specification/ws-bpel.

**[23]** Y. Qian and Y. Xu.Tool Support for BPEL Verification in ActiveBPEL Engine. In: Proceedings of the 18th Australian Software Engineering Conference (ASWEC), pp. 90--100 (2007)

**[24]** Van der Aalst, W.M.P., Pesic, M.: DecSerFlow: Towards a Truly Declarative Service Flow Language. In: Bravetti, M., Nunez, M., Zavattaro, G. (Eds.) WS-FM 2006. LNCS, Vol. 4184, pp. 1--23, Springer (2006)

**[25]** J.F. Allen. Maintaining knowledge about temporal intervals. Communications of the ACM, 26, 832.843, (1983).

**[26]** D.M. Gabbay. Temporal Logic: Mathematical Foundations and Computational Aspects. Oxford University Press (1991).

**[27]** R. Kowalski and M. Sergot. A logic-based calculus of events. New Generation Computing. 4(1):67–96 (1986).

**[28]** F. Pirri and R. Reiter. Some contributions to the meta-theory of the Situation Calculus. Journal of the ACM, 46(3):325–361 (1999)

**[30]** OMG Group: Semantics of Business Vocabulary and Business Rules Specification. OMG (2006)

**[31]** N.E. Fuchs, K. Kaljurand, and T. Kuhn. Discourse Representation Structures for ACE 6.0. Technical Report ifi-2008.02, Department of Informatics, University of Zurich (2008)