An Outlook on Semantic Business Process Mining and Monitoring

A.K. Alves de Medeiros[1], C. Pedrinaci[2], W.M.P. van der Aalst[1], J. Domingue[2], M. Song[1], A. Rozinat[1], B. Norton[2], and L. Cabral[2]

[1] Eindhoven University of Technology, P.O. Box 513, 5600MB, Eindhoven, The Netherlands
{a.k.medeiros,w.m.p.v.d.aalst,m.s.song,a.rozinat}@tue.nl
[2] Knowledge Media Institute, The Open University, Milton Keynes, UK
{c.pedrinaci,j.b.domingue,b.j.norton,l.s.cabral}@open.ac.uk

Abstract. Semantic Business Process Management (SBPM) has been proposed as an extension of BPM with Semantic Web and Semantic Web Services (SWS) technologies in order to increase and enhance the level of automation that can be achieved within the BPM life-cycle. In a nutshell, SBPM is based on the extensive and exhaustive conceptualization of the BPM domain so as to support reasoning during business processes modelling, composition, execution, and analysis, leading to important enhancements throughout the life-cycle of business processes. An important step of the BPM life-cycle is the analysis of the processes deployed in companies. This analysis provides feedback about how these processes are actually being executed (like common control-flow paths, performance measures, detection of bottlenecks, alert to approaching deadlines, auditing, etc). The use of semantic information can lead to dramatic enhancements in the state-of-the-art in analysis techniques. In this paper we present an outlook on the opportunities and challenges on semantic business process mining and monitoring, thus paving the way for the implementation of the next generation of BPM analysis tools.

1 Introduction

Nowadays many companies use information systems to support the execution of their business processes. Examples of such information systems are ERP, CRM or Workflow Management Systems. These information systems usually generate events while executing business processes [9] and these events can be recorded in logs (cf. Figure 1). The competitive world we live in requires companies to adapt their processes in a faster pace. Therefore, continuous and insightful feedback on how business processes are actually being executed becomes essential. Additionally, laws like the Sarbanes-Oxley Act force companies to show their compliance to standards. In short, there is a need for good analysis tools that can provide feedback information about how business process are actually being executed based on the observed (or registered) behavior in event logs.

Business Process Management (BPM) systems aim at supporting the whole life-cycle (design, configuration, execution and analysis) necessary to deploy and

R. Meersman, Z. Tari, P. Herrero et al. (Eds.): OTM 2007 Ws, Part II, LNCS 4806, pp. 1244–1255, 2007.

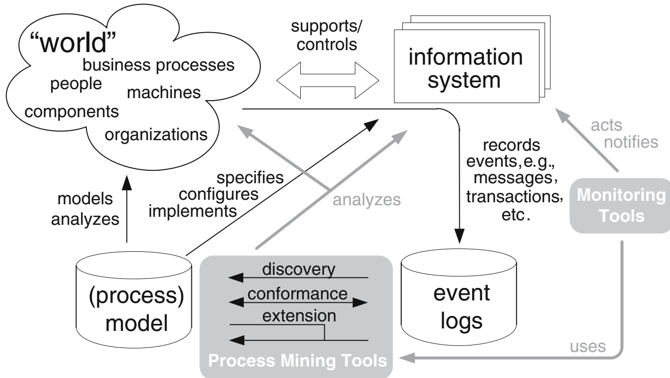

Fig. 1. Overview of process mining and monitoring

maintain business process in organizations. However, current approaches to BPM suffer from a lack of automation that would support a smooth transition between the business world and the IT world [14]. The difficulties for automating the transition between both worlds is due to a lack of machine processable semantics. Therefore, [14] proposes the creation of SBPM systems. Such systems combine Semantic Web and SWS technologies with BPM as a solution for overcoming these difficulties. In a nutshell, SBPM targets accessing the process space (as registered in event logs) of an enterprize at the *knowledge level* so as to support reasoning about business processes, process composition, process execution, etc. The driving force behind SBPM is the use of ontologies [12].

A key aspect of maintaining systems and the processes they support is the capability to analyze them. This analysis can be real-time and may eventually lead to some action or can just be used to inform the involved systems/people. When going SBPM, the main opportunity is that this analysis can be enhanced because it is based on concepts rather than syntax. This semantic perspective is captured by annotating the elements in the systems. So, two challenges arise in this aspect: (i) *how to make use of this semantic data*, and (ii) *how to mine this semantic information* and, consequently, help in the migration of current systems to SBPM environments. In this paper we show how process mining and monitoring techniques successfully utilize semantic data in SBPM systems.

Process mining techniques are especially suitable to analyze event logs. The analysis provided by current process mining techniques [2,4] can be seen as from three types: *discovery*, *conformance* and *extension* (cf. Figure 1). The techniques that focus on *discovery* mine information based on data in an event log only. This means that these techniques do not assume the existence of pre-defined models to describe aspect of processes in the organization. Examples are *control-flow mining* algorithms that extract a process model based on the dependency relations that can be inferred among the tasks in the log. The algorithms for *conformance* checking verify if logs follow *prescribed* behaviors and/or rules. Therefore, besides a log, such algorithms also receive as input a model (e.g., a Petri net or a set of rules) that captures the desired property or behavior to check. Examples are the mining algorithms that assess how much the behavior expressed in a log matches the behavior defined in a model and points out the differences, or algorithms used for auditing of logs (in this case, the model is the property to be verified). The *extension* algorithms enhance existing models based on information discovered in event logs, e.g., algorithms that automatically discover business rules for the choices in a given model.

Process monitoring deals with the analysis of process instances at *runtime* by processing events propagated by the information systems supporting business processes. The goal of process monitoring is to track the enactment of processes as they are performed, in order to have timely information about the evolution of business activities, supporting business practitioners in the identification of deviations and the eventual application of corrective measures. In fact, experience shows that many factors can alter the ideal evolution of business processes (e.g., human intervention, mechanical problems, meteorological adversities, etc) and the quick adoption of special measures can mitigate to an important extent the eventual consequences, thus reducing or even avoiding derived economical losses. The importance of process monitoring in BPM is widely acknowledged and in fact all the main vendors in this sector provide their own solution. Two kinds of monitoring are usually distinguished: (i) *active monitoring* which is concerned with "real time" propagation of relevant data concerning the enactment of business processes, such as the status or the execution time; and (ii) *passive monitoring* which delivers information about process instances upon request.

The ideas presented in this paper are currently being implemented in the context of the European project SUPER [1]. As stated in [1], SUPER "aims at providing a semantic-based and context-aware framework, based on Semantic Web Services technology that acquires, organizes, shares and uses the knowledge embedded in business processes within existing IT systems and software, and within employees' heads, in order to make companies more adaptive". This semantic framework will support the four phases of the BPM life-cycle.

The remainder of this paper provides an outlook about semantic business process mining (Section 2) and monitoring (Section 3), discusses related work in the area of semantic analysis (Section 4), and presents the conclusion and future steps (Section 5).

2 Semantic Business Process Mining

The use of ontologies in SBPM yields two opportunities for process mining techniques. The first opportunity is to make use of the ontological annotations in logs/ models to develop more robust process mining techniques that analyze the logs/ models at the concept level. In this case, it is assumed that event logs and models indeed link to ontologies. The second opportunity is to use process mining techniques to discover or enhance ontologies based on the data in event logs.

Developing Semantic Process Mining Techniques

As explained in Section 1, current process mining techniques focus on the *discovery* of models, the *conformance* between models and logs, and *extension* of

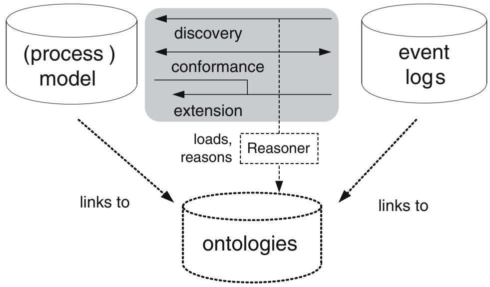

Fig. 2. Semantic process mining: basic elements

models based on information derived from event logs (cf. Figure 1). However, the analysis they support is purely syntactic. In other words, these mining techniques *are unable to reason over the concepts behind the labels in the log*, thus the actual semantics behind these labels remain in the head of the business analyst which has to interpret them. Leveraging process mining to the conceptual layer can enhance state-of-the-art techniques towards more advanced, adaptable and reusable solutions.

The basic elements to build semantic process mining tools are: *ontologies*, *references from elements in logs/models to concepts in ontologies*, and *ontology reasoners* (cf. Figure 2). *Ontologies* [12] define the set of shared concepts necessary for the analysis, and formalize their relationships and properties. The *references* associate meanings to labels (i.e., strings) in event logs and/or models by pointing to concepts defined in ontologies. The *reasoner* supports reasoning over the ontologies in order to derive new knowledge, e.g., subsumption, equivalence, etc. The use of ontologies and reasoners causes an immediate benefit to process mining techniques: the level of abstraction is raised from the syntactical level to the semantical level. The following paragraphs sketch some of the ways in which semantics can aid process mining (some of which have been implemented in ProM [3]).

The *discovery* techniques mine models based on event logs. Control-flow mining techniques are prominent in this perspective. These techniques focus on the discovery of a business model that capture the control-flow structure of the tasks in the log. Currently, these techniques mainly discover a *flat* model showing all the tasks encountered in the log, i.e., a single large model is shown without any hierarchy or structuring. However, if the tasks in these instances would link to concepts in ontologies, subsumption relations over these ontologies could be used to aggregate tasks and, therefore, mine *hierarchical* process models supporting different levels of abstraction. Other discovery techniques focus on organizational mining, which target the discovery of organizational related aspects in event logs. These algorithms are based on the tasks in the logs and the performers of these tasks. The main driving force here is the concept of task similarity. In a nutshell, tasks are considered to be similar based on their names, performers and context (neighboring tasks in the process instances). When these concepts are linked to tasks/performers in logs, more robust similarity criteria can be inferred that make use of the *conceptual* relationships in the ontologies. Consequently, better models can be mined.

The *conformance* checking techniques verify how compliant a model and a log are. This model captures properties/requirements that should be fulfilled by the execution of processes. An example of such technique is the LTL Conformance Checker [2] which allows for the auditing of logs. The problem here is that these techniques require an *exact* match between the elements (or strings) in the log and the corresponding elements in the models. As a consequence, many defined models cannot be reused over different logs because these logs do not contain the same strings as the elements in the models. When ontologies are used, these models can be defined over *concepts* and, as far as the elements in different logs link to the same concepts (or super/sub concepts of these concepts), the conformance can be assessed without requiring any modification of the models or the logs.

The *extension* techniques enhance models based on information mined from event logs. Like the conformance checking techniques, the enhancements are only possible with an exact match between elements in models and logs. Thus, the use of ontologies would bring this match to the concept level and, therefore, models could also be extended based on different logs.

As mentioned before, several of these ideas are currently being implemented as semantic plug-ins in the ProM tool. Actually, the *Semantic LTL Checker* analysis plug-in is already publicly available [1]. This plug-in extends the original LTL Checker [2] by adding the option to provide concepts as input to the parameters of LTL formulae. All the semantic plug-ins developed in ProM are based on the following concrete formats for the basic building blocks (cf. Figure 2): (i) *event logs* are in the SA-MXML file format, which is a semantically annotated version of the MXML format already used by ProM [2]; (ii) *ontologies* are defined in WSML [10]; and (iii) the *WSML 2 Reasoner Framework* [3] is used to perform all the necessary reasoning over the ontologies.

Using Process Mining to Discover/Enhance Ontologies

So far we have focussed on using semantics to enhance process mining techniques. However, there are opportunities in the other direction too because process mining techniques can be used to (i) discover or enhance ontologies and (ii) automatically infer concepts to elements that are not semantically annotated but that belong to partially annotated logs/models. When deploying SBPM systems, a core requirement is that (some of) the elements in the configured models should link to concepts in ontologies because that is how the semantic perspective is embedded in such systems. Therefore, if companies want to go in this direction, they need to add these semantic annotations to their systems. Here,

---

[1] This plug-in can be downloaded together with the nightly build for the ProM tool at http://ga1717.tm.tue.nl/dev/prom/nightly/. It can be started by clicking the menu option "Analysis → Semantic LTL Checker".

[2] The schema for the SA-MXML format is available at http://is.tm.tue.nl/research/processmining/SAMXML.xsd

[3] This framework is publicly available at http://tools.deri.org/

three options are possible. The first one is to *manually* (i) create all the necessary ontologies and (ii) annotate the necessary elements in the SBPM systems. The second option is to use tools to (semi-)*automatically* discover ontologies based on the elements in event logs. Note that, if necessary, these mined ontologies can be manually improved. The third option is a combination of the previous two in which models/logs are partially annotated by a person and mining tools are used to discover the other missing annotations for the remaining elements in logs/models. Discovery and extension process mining techniques can play a role in the last two options.

Basically, three opportunities exist to extract semantics from logs. First, process mining techniques can be created to derive relationships between concepts for activities and performers. This scenario assumes that the subsumption relationships for the concepts in an ontology have not been defined. A task is usually only executed by a group of performers who have certain properties (e.g. organizational units, skills) for a given process, and these properties can be expressed by the concepts linked to these performers. This way subsumption relationships can be discovered from event logs that contain semantic information. Second, if the log is partially annotated then mining techniques can be developed to automatically annotate the tasks and/or performers that do not link to any concepts. Third, if there are no semantic annotations, concepts that describe tasks or performers can be discovered from process logs by applying the existing mining techniques to discover these concepts/ontologies. The mined organizational structures such as roles and teams can be good candidates for concepts. Note that a group of performers executing a same task might belong to the same role and have the same role concept. Performers involved in the same instances might have the same team concept.

3 Semantic Business Process Monitoring

Reaching the level of automation demanded by current businesses requires reasoning over the knowledge gained by applying mining techniques combined with pre-existing contextual domain knowledge about some specific business process. We refer as *Semantic Business Process Monitoring* to the enhancement of Business Process Monitoring with formal semantic descriptions to achieve this. We propose a 5-phases approach, *Observe - Evaluate - Detect - Diagnose - Resolve*, structured around an extensive use of ontologies as the core means for defining formal conceptualizations, and Problem-Solving Methods (PSM) as composable SWS encapsulating the expertise of the monitoring tool [6,20].

Figure 3 depicts our approach to Semantic Business Process Monitoring. The process starts with the *Observe* phase, which is in charge of gathering information populated by the IT infrastructure. The *Evaluate* phase uses this information for computing process metrics such as the execution time, the number of failures, etc. The *Detect* phase follows and uses previously computed metrics and monitoring data in order to detect or predict process deviations and special situations one might want to track. Finally, once a process deviation has been identified, the

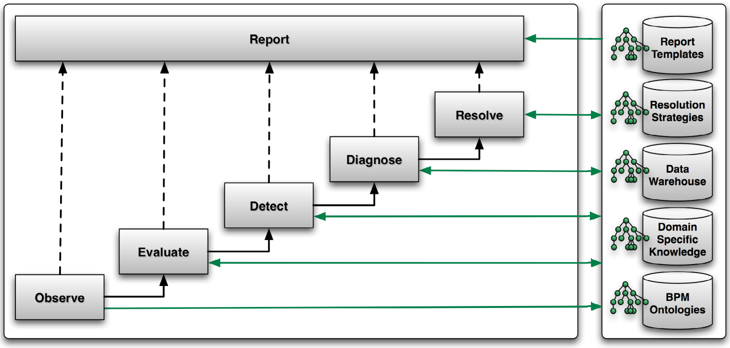

Fig. 3. Phases of Semantic Business Process Monitoring

Diagnose phase is in charge of determining the cause which can eventually be used during the *Resolve* step for defining and applying corrective actions. In parallel, at any time, we have to present information to the user about the overall monitoring process. Each of these monitoring phases, but in particular Detection, Diagnosis, and Resolution, present interesting challenges that have to be addressed and where knowledge-based techniques can help to improve the current state-of-the-art. We shall next identify the main opportunities that arise and depict the approach we envision for Semantic Business Process Monitoring.

Observe. The Observe phase is concerned with obtaining *monitoring* information and lifting it into a semantic form. This phase covers the so-called Extract-Transform-Load step which requires integrating a large amount of disparate information coming from several distributed and heterogeneous systems. Ontologies [12] are therefore a particularly well-suited candidate for supporting this task. An initial version of such an ontology has been defined in [18] based on the MXML format defined within the ProM framework [3]. Once in an ontological form, the monitoring information supports navigation, manipulation, and querying at the knowledge level, which is closer to human understanding and can potentially lead to important improvements in the user interface. In fact, in a recent report by Gartner [19] metadata management is presented as the most important capability that Business Intelligence tools should integrate. Ontologies are therefore a key enabling technology for achieving this. Additionally, semantic monitoring data is amenable to automated reasoning thus enabling the application of knowledge-based technologies as described next. Among the possibilities brought, consistency checking can be applied in this phase for detecting anomalies in the monitoring data itself thus reducing the noise for subsequent analysis and potentially enhancing quality of the analysis results.

Evaluate. This phase is in charge of the timely computation of process metrics, such as the execution time or the number of failures. We can distinguish between generic metrics that can be computed for every process, and domain-specific metrics [8]. To support business practitioners, we envision the definition of domain-specific metrics using a metric ontology, and the capability for users to define SWS that can be invoked by platforms like the IRS-III [6] to perform the metric computation. In a somewhat recursive way, we envisage *formalizing the analysis results themselves*. This provides independence with respect to the engines or algorithms utilized for performing the calculations, and supports a semantically enhanced view over the results. More importantly, an ontological definition of analysis results, enhances the overall body of knowledge for supporting further reasoning. In fact, it is quite usual that taking a decision requires performing and correlating diverse analysis, e.g., by combining the processes that did not perform well, with the resources involved in them, one could identify the bottlenecks. Formalizing the results enables reasoning over the computationally expensive analysis results within runtime monitoring tasks, as well as it allows for automatically combining them in order to perform more complex evaluations. In this sense we envision the use of SWS technologies for supporting the definition of analysis processes as the orchestration of different analysis techniques.

Detect. The Detect phase is in charge of identifying or predicting deviations with respect to the expected behavior of a process. The simplest approach is based on the definition of thresholds with respect to certain metrics. More complex solutions can be applied by approaching detection as a classification problem [8]. Our approach can support the seamless application of knowledge-based algorithms, e.g., classification PSMs [20], the enhancement of existing algorithms with semantic information, or even the runtime adaptation of the detection process. It is known that selecting the appropriate algorithm to apply given the task at hand is particularly important [5,8]. Having an extensive conceptualization of the BPM domain can indeed be particularly beneficial in order to select the presumably most suitable algorithm. This can be achieved by performing dynamic selection of SWS implementing some algorithm on the basis of the characteristics of the domain. For example, knowing the kind of process analyzed, e.g., shipping process, we can identify the typical or more relevant deviations, e.g., deadline exceeded, and select the algorithm accordingly. Additional advantages can be gained if relations between metrics and domain data, as well as mining results are modelled, allowing the system to overcome the lack of information earlier in the execution of the process. Finally, contextual knowledge can also strengthen existing algorithms like data mining approaches to symptoms detection [8] where this knowledge can play an important role supporting the enhancement of the algorithm with semantic feature selection.

Diagnose. Once any deviation has been detected or predicted, we have to *diagnose* the origin of the problem. In the BPM community, diagnosis often depends on the actual interpretation of the data by the user [8,16]. In order to do so the detection phase is often based on some structured approach that can be relatively easily understood by humans, e.g., decision trees. Diagnosis has been a popular topic in Artificial Intelligence, and has led to a quite exhaustive characterization of the task as well as to a wide range of implementations [5,20] which it would be desirable to benefit from. Knowledge-based methods have been applied to diagnosing automated systems (where some behavioral model typically exists), as well as to the diseases (where this kind of model is typically missing). It is therefore safe to assume that we can make use of the wealth of research on diagnosis for Semantic Business Process Monitoring. It is worth noting in this respect that a close integration between monitoring and mining can allow us to reuse mined process models for informing the diagnosis algorithm. This can be of great advantage when no prescribed process model exists or when the prescribed model differs to an important extent from the actual mined model.

Resolve. The final phase is concerned with the design and application of a resolution strategy for addressing some previously diagnosed process deviation. Resolution is by far the most complex task within our approach and in fact little work besides ad-hoc exception handling or *undo and retry* has been done within the BPM community [13,16]. These approaches cannot cope with the wide range of deviations that can arise during the enactment of a process and fully automated handling of any process deviation is simply not realistic due to unforeseen situations affecting user-defined and process-specific conditions [16]. Hence, in a similar vein to [16] we contemplate the application of Case-Based Reasoning for retrieving, adapting, and applying resolution strategies in an attempt to deal with previously diagnosed deviations. Like in the previous phases, the resolution strategies will be defined as orchestrations of SWS, allowing users to specify their own strategies by reusing and combining problem-solving expertise over their domain specific terms. This approach is inline with that proposed by [15] that can in fact serve as a basis for defining general resolution templates. We expect however that the capability for executing PSMs and our extensive conceptualization of the BPM domain will enable the creation of more complex strategies. For instance, Organizational knowledge can support the escalation of tasks [22], Rescheduling based on Configuration Problem-Solving can allow adapting resource allocation, or even Planning and Scheduling using reusable and equivalent process fragments can support the implementation of process escalations by degrading the Quality of Service [22].

4 Related Work

The idea of using semantics to perform process analysis is not new [7,11,14,17,21]. In 2002, Casati et al. [7] introduced the *HPPM intelligent Process Data Warehouse (PDD)*, in which taxonomies are used to add semantics to process execution data and, therefore, support more business-like analysis for the provided reports. The work in [11] is a follow-up of the work in [7]. It presents a complete architecture for the analysis, prediction, monitoring, control and optimization of process executions in Business Process Management Systems (BPMSs). This set of tools suite is called *Business Process Intelligence (BPI)*. The main difference of these two approaches to ours is that (i) taxonomies are used to capture the semantic aspects (in our case, ontologies are used), and (ii) these taxonomies are flat (i.e., no subsumption relations between concepts are supported). Hepp et al. [14] proposes merging Semantic Web, Semantic Web Services , and Business Process Management (BPM) techniques to build Semantic BPM systems. This visionary paper pinpoints the role of ontologies (and reasoners) while performing semantic analysis. However, the authors do not elaborate on the opportunities and challenges for semantic process mining and monitoring. The works by Sell et al. [21] and O'Riain et al. [17] are related to ours because the authors also use ontologies to provide for the semantic analysis of systems. The main difference is the kind of supported analysis, since their work can be seen as the extension of OLAP tools with semantics. The work in [17] shows how to use semantics to enhance the business analysis function of detecting the core business of companies. This analysis is based on the so-called Q10 forms. Our paper is the first one to provide an outlook on semantic process mining and monitoring techniques.

5 Conclusions and Future Work

This paper has presented several directions for the development of semantic process mining and monitoring tools. These tools can be used to analyze SBPM systems. The main opportunity provided by such systems is the *link between the generated events* (necessary for analysis) *and the actual concepts they represent*. This link is achieved by annotating the elements (models, events etc) in SBPM systems with concepts in ontologies. However, this same opportunity also raises two challenges. The first one is *how to make use of this semantic perspective* in process mining and monitoring tools. For the development of semantic process mining tools, we have proposed a framework composed of three building blocks (annotated event logs, ontologies and reasoners) and have discussed different ways in which techniques aiming at the discovery, conformance or extension perspectives could go semantic. For the monitoring tools, we have explained a five-phase approach (observe, evaluate, detect, diagnose and resolve) in which knowledge-based techniques play an essential role. The second challenge is *how to mine the semantic information* and, therefore, help in the migration of current information systems to SBPM environments. Here we have illustrate how process mining techniques could use events relating to tasks and performers to (i) automatically discover or enhance ontologies, and (ii) help in the semantic annotation of the elements in information systems.

As indicated throughout the paper, some of the presented ideas have already been implemented in the context of the SUPER European project. In fact, our future work will proceed in this direction (the development of further ideas in the SBPM environment defined in SUPER).

Acknowledgements. This research is supported by the European project SUPER [1].

References

1. European Project SUPER - Semantics Utilised for Process Management withing and between Enterprises. `http://www.ip-super.org/`
2. van der Aalst, W.M.P., de Beer, H.T., van Dongen, B.F.: Process Mining and Verification of Properties: An Approach Based on Temporal Logic. In: Meersman, R., Tari, Z. (eds.) On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE. LNCS, vol. 3760, pp. 130–147. Springer, Heidelberg (2005)
3. van der Aalst, W.M.P., van Dongen, B.F., Günther, C.W., Mans, R.S., Alves de Medeiros, A.K., Rozinat, A., Rubin, V., Song, M., Verbeek, H.M.W., Weijters, A.J.M.M.: ProM 4.0: Comprehensive Support for Real Process Analysis. In: Kleijn, J., Yakovlev, A. (eds.) Application and Theory of Petri Nets and Other Models of Concurrency (ICATPN 2007). LNCS, vol. 4546, pp. 484–494. Springer, Heidelberg (2007)
4. van der Aalst, W.M.P., van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, A.J.M.M.: Workflow Mining: A Survey of Issues and Approaches. Data and Knowledge Engineering 47(2), 237–267 (2003)
5. Benjamins, R.: Problem-solving methods for diagnosis and their role in knowledge acquisition. Int. Journal of Expert Systems: Research & Applications 8(2), 93–120 (1995)
6. Cabral, L., Domingue, J., Galizia, S., Gugliotta, A., Tanasescu, V., Pedrinaci, C., Norton, B.: IRS-III: A Broker for Semantic Web Services based Applications. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 201–214. Springer, Heidelberg (2006)
7. Casati, F., Shan, M.-C.: Semantic Analysis of Business Process Executions. In: Jensen, C.S., Jeffery, K.G., Pokorný, J., Šaltenis, S., Bertino, E., Böhm, K., Jarke, M. (eds.) EDBT 2002. LNCS, vol. 2287, pp. 287–296. Springer, Heidelberg (2002)
8. Castellanos, M., Casati, F., Dayal, U., Shan, M.-C.: A Comprehensive and Automated Approach to Intelligent Business Processes Execution Analysis. Distributed and Parallel Databases 16(3), 239–273 (2004)
9. Dumas, M., van der Aalst, W.M.P., ter Hofstede, A.H.M.: Process-Aware Information Systems: Bridging People and Software through Process Technology. Wiley & Sons, Chichester (2005)
10. Fensel, D., Lausen, H., Polleres, A., de Bruijn, J., Stollberg, M., Roman, D., Domingue, J.: Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer, Heidelberg (2007)
11. Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M., Shan, M.-C.: Business Process Intelligence. Computers in Industry 53(3), 321–343 (2004)
12. Gruber, T.R.: A translation approach to portable ontology specifications. Knowledge Acquisition 5(2), 199–220 (1993)
13. Hagen, C., Alonso, G.: Exception handling in workflow management systems. IEEE Transactions on Software Engineering 26(10), 943–958 (2000)
14. Hepp, M., Leymann, F., Domingue, J., Wahler, A., Fensel, D.: Semantic business process management: A vision towards using semantic web services for business process management. In: ICEBE, pp. 535–540 (2005)
15. Klein, M., Dellarocas, C.: A Knowledge-based Approach to Handling Exceptions inWorkflow Systems. Comput. Supported Coop. Work 9(3-4), 399–412 (2000)
16. Luo, Z., Sheth, A., Kochut, K., Miller, J.: Exception Handling in Workflow Systems. Applied Intelligence 13(2), 125–147 (2000)

17. O'Riain, S., Spyns, P.: Enhancing the Business Analysis Function with Semantics. In: Meersman, R., Tari, Z. (eds.) On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE. LNCS, vol. 4275, pp. 818–835. Springer, Heidelberg (2006)
18. Pedrinaci, C., Domingue, J.: Towards an Ontology for Process Monitoring and Mining. In: Semantic Business Process and Product Lifecycle Management (SBPM 2007), Innsbruck, Austria (June 2007)
19. Schlegel, K., Hostmann, B., Bitterer, A.: Magic Quadrant for Business Intelligence Platforms, 1Q07. Gartner RAS Core Research Note G00145507, Gartner (January 2007)
20. Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., van de Velde, W., Wielinga, B.: Knowledge Engineering and Management: The CommonKADS Methodology. MIT Press, Cambridge (1999)
21. Sell, D., Cabral, L., Motta, E., Domingue, J., Pacheco, R.: Adding Semantics to Business Intelligence. In: DEXA Workshops, pp. 543–547. IEEE Computer Society Press, Los Alamitos (2005)
22. van der Aalst, W.M.P., Rosemann, M., Dumas, M.: Deadline-Based Escalation in Process-Aware Information Systems. Decision Support Systems 43(2), 492–511 (2007)