USING BPMN AND TRACING FOR RAPID BUSINESS PROCESS PROTOTYPING ENVIRONMENTS

Alessandro Ciaramella[1], Mario G. C. A. Cimino[2], Beatrice Lazzerini[2] and Francesco Marcelloni[2]
[1]*IMT Lucca Institute for Advanced Studies, Piazza San Ponziano 6, 55100 Lucca, Italy*

[2] *Dipartimento di Ingegneria dell'Informazione: Elettronica, Informatica, Telecomunicazioni*
University of Pisa, Via Diotisalvi 2, 56122 Pisa, Italy

Keywords: Business Process Management, BPMN, Business Simulation, Traceability.

Abstract: Business Process (BP) analysis aims to investigate properties of BPs by performing simulation, diagnosis and verification with the goal of supporting BP Management (BPM). In this paper, we propose a framework for BPM that relies on the BP Modeling Notation (BPMN). More specifically, we first introduce a method to deal with the BPM life cycle. Then, we discuss a platform to support this life cycle. The platform comprises three basic modules: a visual BPMN-based designer, a process tracing service, and a BP Manager for, respectively, the design, configuration and execution phases of the BPM life cycle. The proposed framework is particularly useful to perform business simulations such as what-if analysis, and to provide an efficient integration support within the supply-chain. In this study, we also show some practical application of this framework through a real-world experience on a leather firm, offering an environment for process communication as well as for time and cost analysis.

1 INTRODUCTION

Business Process (BP) analysis aims to investigate properties of BPs that are neither obvious nor trivial. To this end, the term "analysis" is used with a rather broad meaning, including simulation and diagnosis, verification and performance analysis (Van der Aalst *et al.*, 2003). Process simulation facilitates process diagnosis since, by simulating real-world cases, domain experts can verify whether process models work properly and possibly propose modifications of the original process models. Further, if BP models are expressed in process languages with clear semantics, their structural properties can be analyzed. For example, we could verify whether certain activities of the processes can never be activated and fix the modeling mistake. While basic structural properties of process models have been studied for some time, very few software products actually support them. Only recently, some tools for BP management have been proposed with the aim of managing business processes using methods, techniques, and software to design, control, and analyze operational processes (Van der Aalst *et al.*, 2003).

As Smith and Fingar (2003) pointed out, the enabler of a good BPM is an open process-modeling language standard, which allows interoperability and ease of use. Unfortunately, at present, there does not exist a widely accepted and adopted standard (Wang *et al.*, 2006), but there are several language proposals to get an abstract representation of business processes. For instance, UML 2.0 Activity Diagram (Object Management Group, 2008, UML) allows a high-level representation of dynamic system behavior, which is well suitable for model business processes under a control-flow perspective. This kind of diagrams, however, is extremely limited in capturing organizational aspects, or in managing resource allocation and distribution (Russel *et al.*, 2006).

Petri net-based languages have interesting properties, such as the use of formal semantics, the presence of analysis techniques, and a state-based control flow. Some problems, however, arise in terms of expressiveness, because these languages are unable to represent patterns involving multiple partners, coordination and cancellation (Van der Aalst and Hofstede, 2002).

Event-Driven Process Chains (EPCs) (Scheer, 1999) are a method for representation of BP models

Ciaramella A., G. C. A. Cimino M., Lazzerini B. and Marcelloni F. (2009).
USING BPMN AND TRACING FOR RAPID BUSINESS PROCESS PROTOTYPING ENVIRONMENTS.
In *Proceedings of the 11th International Conference on Enterprise Information Systems - Information Systems Analysis and Specification*, pages 206-212
DOI: 10.5220/0002005002060212

focused on supporting data and model interchange. They have been proposed for allowing business managers to easily understand and use BPs. The main drawback of the EPCs is the lack of a support for information management and organizational aspects (List and Korheer, 2006).

Integrated DEFinition (IDEF) modeling languages have been conceived for use in system engineering as a standard method to document and analyze processes. IDEF adopts different methods which have been introduced in subsequent phases in the standard. Each method refers to specific modeling aspects. In order to provide a complete description of the behavior of a system, these different methods should be integrated. The use and integration of these methods require consistency between different levels of modeling which could be difficult to maintain (Noran, 2005).

BP Modeling Notation (BPMN) (Object Management Group, 2008, BPMN) has been introduced in 2004 by the BP Management Initiative (BPMI.org) and has been adopted as a standard by the Object Management Group (OMG) in 2005. The primary goal of BPMN is to provide a notation that is readily understandable by all business stakeholders (White, 2004). BPMN is also supported with an internal model that enables the generation of executable programs in Web Service Business Process Execution Language (WSBPEL) (OASIS, 2007), i.e., a language for specifying BP behavior based on web services. Thus, BPMN represents a standardized bridge for the gap between the business process design and implementation. Furthermore, BPMN provides support to represent the most common control-flow modeling requirements that occur when defining process models (Wohed *et al.*, 2006). In this study, BPMN is considered as a reference standard in BPM.

In this paper, we introduce a framework based on BPMN for managing business processes. The framework proposes a method to deal with the BPM life cycle and a platform to support the method along the different phases of the life cycle. Finally, we describe the application of the framework to a real-case in the leather supply chain.

The paper is structured as follows. Section 2 illustrates the most important phases of the proposed method. Section 3 is devoted to an introduction of the BPMN standard as a visual language. Section 4 concerns the correspondence between BPMN and WSBPEL. The proposed BPM platform is presented in Section 5, whereas Section 6 describes the application of the framework to two real example processes in the leather supply chain.

2 BPM LIFE CYCLE

A BPM life cycle describes an approach that comprises various phases in support of operational business processes (Van der Aalst *et al.*, 2003). There is a variety of BPM approaches in the literature (Weske *et al.*, 2004). In this context, BPMN is a core enabler of an emerging approach aimed at unifying previously distinct disciplines, such as process modeling, simulation, workflow and enterprise integration, so as to propose a single standard paradigm. In this paper, we consider the BPM life cycle proposed by Van der Aalst *et al.* (2003) and Weske *et al.* (2004). The cycle consists of four phases. In the first phase, i.e., the *design phase*, processes are (re-)designed. In the *configuration phase*, designs are implemented by configuring a process-aware information system (e.g., a workflow management system). After configuration, the *enactment phase* starts when the operational business processes are executed using the information system. In the *diagnosis phase*, the operational processes are analyzed to identify problems and to find possible improvements. In the following, the union of the enactment and the diagnosis phases is called *execution* phase.

In this paper, we focus on a specific implementation of this life cycle, which relies on the BPMN standard. Figure 1 describes this implementation. First, we model the business processes by using a BPMN designer tool which allows defining the processes in a visual and understandable manner. In this phase, called *modeling*, a set of parameters (e.g., duration time and/or quality features) is also associated by the designer with each activity in the model. In the second phase, i.e., *coding*, the models conceived by means of BPMN are translated into a business execution language, i.e., WSBPEL 2.0, in order to obtain an executable set of components representing the modeled business processes. Third, in the *tuning*

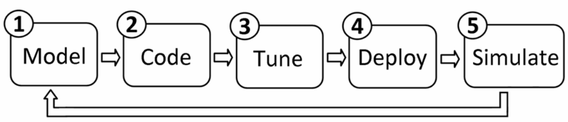


Figure 1: A BPM life cycle implementation.

phase, the business activities are configured by extracting a set of values for the previously selected parameters from a process tracing service. Fourth, in the *deployment* phase, the resulting components are deployed in a BP server, which is based on web services technology. Finally, in the last phase, i.e., *simulation*, simulations are performed (e.g., what-if analysis for detailed cost and time estimations). In this phase, thanks to the use of web services-based components, a communication infrastructure between supply-chain actors can be also automatically derived. As shown in Figure 1, the overall method can be executed iteratively, in order to achieve a continuous improvement of the BP analysis.

3 BPMN AS A VISUAL LANGUAGE FOR PROCESSES

The importance of conceptual modeling is largely recognized in the literature (Wand and Weber, 2002). Modeling is a learning process, where business analysts can make clear requirements, boost the domain knowledge and express rough solutions. Visual models act as communication channels between business managers and technicians, and provide the necessary documentation to manage post-project activities. BPMN is an OMG effort to provide businesses with an operational standard to visually model business processes and their relationships.

To describe business processes, BPMN offers the Business Process Diagram (BPD), which is very similar to the UML Activity Diagram. A BPD consists of four basic categories of elements (Object Management Group, 2008, BPMN): *Flow Objects*, *Connecting Objects*, *Swimlanes*, and *Artifacts* as shown in Figure 2. Key concepts are briefly defined in the following. *Events* are representations of something that can happen during the BP; *business flow* is activated by a *start event* and terminated by an *end event*, while *intermediate events* can occur anywhere within the flow. Business activities can be atomic (*tasks*) or compound (*processes*, as connection of tasks); *gateways* represent decision points to control the business flow. *Data* objects model any information required or provided by activities, whereas *groups* allow logical clustering of activities. *Annotations* are exploited to add comments or text information to diagrams. *Connecting objects* connect flow objects together: *sequence flows* show the order of execution of activities in the BP, *message flows* represent messages exchanged between business entities, and *associations* highlight inputs and outputs of activities. A *pool* represents a participant in a BP and *lanes* allow detailed categorization of activities within a pool. A process can be depicted by means of another BPD. This allows to represent more exhaustive business descriptions, maintaining the overall view of the process flow.

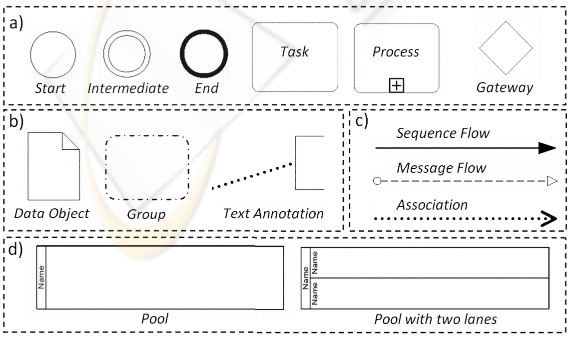


Figure 2: Basic elements in BPMN: (a) flow objects, (b) artifacts, (c) connecting objects, and (d) swimlanes.

Figure 3 shows the process for manufacturing a bag in a real leather firm. Figure 3.a outlines the macro processes, from the bag design to its shipment. Figure 3.b represents a drill down through the “check and ship out” sub-process, where semi-finished products originated from the “assemble” sub-process are checked against quality. The arrow within a circle (Off-Page connector) denotes a link between pieces of the diagram. If the products are good, they are packed and shipped out; otherwise proper corrective actions are triggered to handle the error (the *error* event is represented by a flash of lighting). Finally, Figure 3.c shows the expansion of the “check product” sub-process: in the first gateway, features of the product are compared with specifications and quality plan. If the product is compliant, it can be packed and shipped out. Otherwise, the product is analyzed so as to determine the causes of the fail and examine the possibility to remove the error. If the error is judged removable, then the “reworking” sub-process is fired, otherwise the product is dispatched to the “reject” sub-process.

4 FROM BPMN TO BPEL

Similar to the abstraction and automation goals of UML 2.0, the strength of BPMN resides in two important aspects. First, its simplicity, which is due to the abstraction level provided by the standard. Second, the possibility of being translated (in an automatic manner) into a business execution language, and then to generate a machine-readable prototype of business processes. BPMN was developed with a solid mathematical foundation provided by exploiting process calculus theory (White, 2004). This theory is an essential requirement for a good business process modeling language, in order to automate execution and to easily provide proofs of general consistency properties, as it is widely recog-

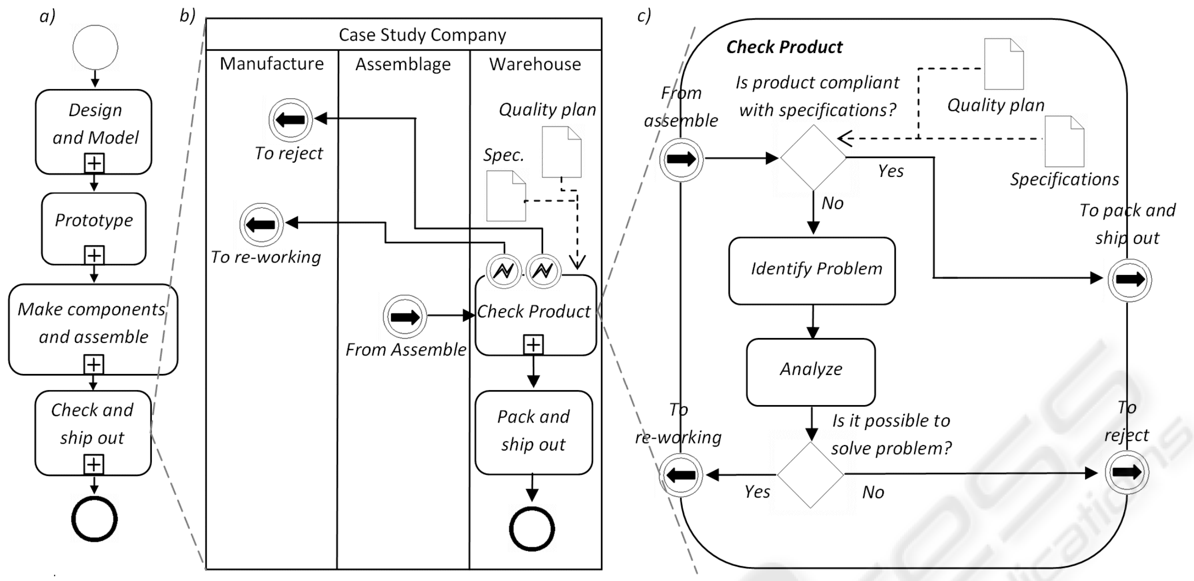


Figure 3: Excerpt of the business process flow representation in BPMN, for the production of a bag. (a) Overall process flow; (b) Drill down through the "check and ship out" process; (c) Drill down through the "check product" process.

nized in the literature (Puhlumann, 2007, and Förster, 2003). Smith and Fingar (2003) highlighted that BPMN is the equivalent graphical visual notation of BPML, the BP Modeling Language developed by BPMI. BPML provides "the reliability, coherence and simplicity needed for users to be able to manipulate processes with great confidence" (Smith and Fingar, 2003). Hence, BPMN was conceived with the specific intention of creating a bridge from the business perspective to the technical perspective about processes (White, 2004).
The use of BPML has been deprecated in favor of WSBPEL, the OASIS standard to specify business process behavior orchestrating web services. However, regardless of BPMN developers declarations (White, 2004), the BPMN-to-WSBPEL mapping is not seamless, as some researchers pointed out (e.g. Oyuang *et al.*, 2006), basically due to the different nature of these languages: graph-oriented the former, block-based the latter.

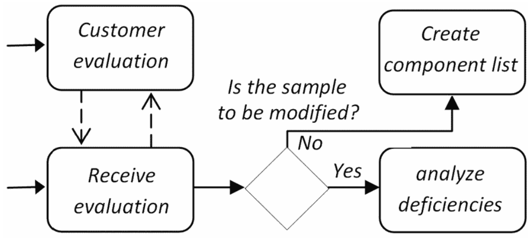
An example of BPMN-to-BPEL coding is shown in Figure 4, where the process flow goes through a decision point (a gateway). First, the "receive evaluation" process exchanges messages with the "customer evaluation" one, and hence the gateway is fired. If the sample has to be modified, then the "analyze deficiencies" process is executed. Otherwise, the "create component list" process is executed.

5 THE BUSINESS PROCESS MANAGEMENT PLATFORM

Weske *et al.* (2004) define a BPM System as "a generic software system that is driven by explicit process designs to enact and manage operational business processes". There are several BPM systems that help business analysts improve their work: BPM market is not well established today, and there is not a predominant vendor.
A recent survey from BPTrends (Harmon and Wolf, 2008) lists 22 BPMS Suites used in organizations, like IBM WebSphere BPM, Oracle BPEL Process Manager, SAP NetWeaver, and so forth. Moreover, two-thirds of interviewed people have declared to adopt graphics modeling tools (e.g. Microsoft Visio, a pure drawing tool) to manage and document business processes in their own company. In particular, we experienced, by using different tools, that the robustness of the BPMN-to-BPEL coding module is crucial for the effectiveness of the overall BPM framework. For instance, in our experimental studies, we tested the Visual Paradigm solution for BPM, i.e., Business Process Visual Architect (BP-VA). Although commercial declarations (Visual Paradigm, 2007) boast to provide extensive support for BPMN, we realized that in BP-VA the following constructs are not immediately translated: (i) an "end" event immediately after a task or a gateway; (ii) tasks immediately after a macro-process; (iii) a task that receives and sends messages at the same

time; (iv) *until loops*, where condition is tested after loop has executed.
The current version of the proposed framework uses the BPM solution proposed by Intalio (www.intalio.com). It is fully compliant with the BPMN standard: it is an open source project and is a very user-friendly tool to support BPM, making it suitable also for non-technical users.

Furthermore, Intalio offers a BPM platform to execute and simulate prototypes of business processes modeled with BPMN and translated into a BPEL specification.




```
<bpel:if>
  <bpel:condition>1 =$evaluation...
  </bpel:condition>
  <bpel:sequence>
    <bpel:empty
      bpmn:label="Analyze deficiencies"
      bpmn:id="..." />
  </bpel:sequence>
  <bpel:else>
   <bpel:sequence>
   <bpel:empty
     bpmn:label="Create Component list"
      bpmn:id="..." />
   </bpel:sequence>
  </bpel:else>
</bpel:if>
```

Figure 4: Example of BPMN-to-BPEL coding.

5.1 The Auto-configuration Phase

One of the purposes of our approach is to perform analytical BP simulations. To this aim, a set of parameters can be associated with each activity. Two examples of these parameters are the duration time and types of defects. Quality requirements often play a crucial role in modern BPM, and thus they deserve particular attention in the corresponding traceability systems as well. The ISO 9000 standard defines quality as the totality of features and characteristics of a product or service that bear its ability to satisfy stated or implied needs. To meet quality requirements, we used a traceability framework developed in Bechini *et al.* (2008). This framework models an abstract class *QualityFeature*, which includes a description of the feature itself and a collection of methods to set and retrieve feature values. Values can be either categorical or numerical. This class organization allows dealing uniformly with different quality features.
Once modeled, each quality feature can be associated with a corresponding parameter of an activity, and this parameter can be fulfilled with a concrete value derived from the traceability system. For instance, the duration time can be derived by calculating the average time of the corresponding traced activity.

5.2 Deployment Process

After being modeled, business processes can be automatically deployed in the Intalio BPM platform. Several artifacts are generated during this step: the BPEL code representing the overall process, some WSDL files to model interactions with other business actors or to describe external services, and other files specific to Intalio.

In particular, the web interfaces related to business processes are managed in the deployment process by suitable Web forms named XForms (W3C, 2004) inside the BPMS Web User Interface. An XForm is an evolution of XHTML Web form, based on the Model-View-Controller paradigm (W3C, 2004). Users exploit XForms to perform message exchange with the system or other business actors, or to choose alternatives in decision points. In our case study, we created several forms to allow business participants to have the control in making decisions at the correct time, in order to lead processes. For instance, an XForm can show the prices of the ordered item and ask for its acceptance, thus letting the user decide on the process flow.

5.3 Simulation and Execution Phase

Simulation is a precious tool to understand how a process behaves, to measure its performance, and to carry out cost-effective business analysis (Tumay, 1996). Kellner *et al.* (1999) list six reasons to perform simulations: strategic management, planning, control and operational management, process improvement and technology adoption, understanding, and training and learning.

Simulation provides a safe environment to test the efficiency and effectiveness of the modeled process, before enacting it in real business. Changes and enhancements are easy to implement directly on the process model, restarting our BPM framework.

Further, in our platform, once the deployment step is completed, information business process can be directly executed on the Intalio BPMS Web User

Interface, which represents an interaction context for business actors communication. In this way, a robust integration within the supply chain is achieved. For instance, a customer can order the accurate quantity of required goods, and a supplier can assure a punctual shipment. This information exchange is automatically connected to the business status.

6 CASE STUDY

We applied our proposed framework to a real leather firm to have concrete insights about its processes and to enhance decision making activities. In this section, we will show two examples concerning the improvement of outsourcer selection, and the analysis of the value stream in the supply chain. Typically, the outsourcers are used to manufacture the bag after cutting the leather.

As regards the outsourcer selection, our framework is able to trace the most important parameters to support this decision. In particular, for a set of crucial activities, we collected information about lead time, variance of lead time, maximum observed delay, price, percentage of faults (or defects), and average delay in signaling defects. Comparing the same parameters for each outsourcer, executive managers can decide the outsourcer which best fits specific requirements of a job order. Furthermore, the process tracing service has allowed managers to know the exact time requirements for certain phases of the manufacturing process. Table 1 shows gathered information concerning the production of a high quality bag. The considered activities sweep from the processing of the leather, through the cut of the bag elements, to the assembly and packing of the end product. Depending on the chosen outsourcer, duration times and costs can change accordingly. Furthermore, other quality parameters have been modeled to better cope with the outsourcer selection problem. In particular percentage of defects and average delay in signaling defects have been traced in order to give executive managers the chance to select the suitable outsourcer.

As regards the second application, the analysis of the value streams concerns the characterization of activities into two classes: value-added and non-value-added activities. This allows determining possible causes of waste and reducing total lead time within the supply chain. Using the proposed framework, each activity has been classified and an execution time has been associated with it. We grouped all activities of our case study into *manufacturing*, *warehousing*, *waiting*, *transport* and *inspection* processes. For each process, we traced lead times as shown in Table 2. This information has allowed management to investigate causes of waste (e.g., waste of time for waiting, waste of space for warehousing, waste of workforce for transport) and to identify improvement areas.

Table 1: Gathered information from process tracing service concerning the production of a high quality bag.

Activity	Lead time *(min.)*	Δ lead time *(min.)*	Price per minute *(Euros)*	Δ price *(Euros)*
Cutting	25	± 3	0.32	± 0.05
Predisposition	28	± 3	0.25	± 0.05
Assembly	135	± 10	0.29	± 0.03
Packing	11	± 2	0.28	± 0.02

Table 2: Lead time for each process.

Activity	Time *(min.)*	Percentage
Manufacturing	2880	53.51 %
Warehousing	1920	35.67 %
Waiting	480	8.92 %
Transport	95	1.77 %
Inspection	7	0.13 %
Total	5382	100.00 %

7 CONCLUSIONS

In this paper, we have proposed a framework to perform business process analysis and to provide an effective integration between supply-chain actors. The framework relies on the BPMN to model processes in a manner that is both human and machine readable, on the basis of visual BP diagrams and BPEL code, respectively.

We have also developed a supporting platform which is mainly composed of a visual designer, a process tracing service, and a business process server. The framework has been experienced and documented on a real leather firm, considering two business cases: the improvement of outsourcer selection, and the analysis of the value stream in the supply chain.

A critical evaluation of the proposed system, inclusive of future improvements, will be based on the results and issues raised by the partners that are experiencing the system. As first result, the system should allow a significant reduction of the time and effort needed to evaluate the real production cost. Then, the system should permit to significantly reduce the problems caused by delays in signaling defects. The most common consequence of these delays is a late shipment, which is very negative in fashion sector. Finally, the system should guarantee an increase in supply chain integration by controlling possible defects and limiting their effects.

On the other hand, the lack of general metrics to assess the quality of the modeled BPs and of the whole BPM workflow (Vanderfeesten *et. al.* 2007) does not allow comparing our system with similar approaches.

ACKNOWLEDGEMENTS

This work was supported by the Tuscany Region under the DOCUP 2000-2006, Measure 2.8, Action 2.8.4, project "Innovazione, progettazione, qualità, tracciabilità per il sistema moda" and the DOCUP 2000-2006, Measure 1.7, Action 1.7.1, project "Tracce sulla pelle". The authors would like to thank the Consorzio Centopercento Italiano and A.S.S.A. Società Cooperativa Consortile for the valuable help in modeling the processes used as example in the paper.

REFERENCES

Bechini, A., Cimino, M.G.C.A., Marcelloni, F., Tomasi, A., 2008. Patterns and technologies for enabling supply chain traceability through collaborative e-business. In *Information and Software Technology*, Elsevier Science, Vol. 50, Issue 4, pp. 342-359.

Förster, M., 2003. Theory of Business Process Modelling: The Pi-Calculus. In *Process-oriented Information Systems*, M. Weske, J. Hündling, H. Schuschel (Eds.), Potsdam, Germany, pp.1-17.

Harmon, P., Wolf, C., 2008. The State of Business Process Management. BPTrends, (*http://www.bptrends.com).*

Kellner, M.I., Madachy, R.J., Raffo, D.M., 1999. Software Process Simulation Modeling: Why? What? How?. In *Journal of Systems and Software*, Vol. 46, No. 2/3, pp- 91-105.

List, B; , Korherr, B., 2006. An Evaluation of Conceptual Business Process Modelling Languages. In *SAC'06, Proc. 221st ACM Symposium on Applied Computing,* pp. 1532-1539.

Noran, O.S., 2005. Business Modelling: UML vs. IDEF. Griffith University, White paper, *(http://www.cit.gu.edu.au/~noran).*

OASIS, 2007. WS-BPEL, Web Services Business Process Execution Language version 2.0. *http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html*, accessed 2008.

Object Management Group, 2008, BPMN, Business Process Modeling Notation, V1.1. Official specification *(http://www.omg.org/spec/BPMN/1.1/PDF).*

Object Management Group, 2008, UML Resource Page, *http://www.uml.org,* accessed 2008.

Oyuang, C., Dumas, M., ter Hofstede, A.H.M., Van der Aalst W.M.P., 2006. From BPMN Process Models to BPEL Web Services. *In ICWS'06, Proc. IEEE International Conference on Web Services*, pp. 285-292.

Puhlmann, F., 2007. On the suitability of the Pi-Calculus for Business Process Management. In *Technologies for Business Information Systems,* Springer-Verlag, Berlin, 2007, pp. 51-62.

Russel, N., Van der Aalst, W.M.P., ter Hofstede, A.H.M., Wohed, P., 2006. On the suitability of UML 2.0 Activity Diagrams for Business Process Modeling. In *APCCM'06, Proc. Third Asian-Pacific Conference on Conceptual Modelling*. pp. 95-104.

Scheer, A.W., 1999.: ARIS – Business Process Modeling. Springer Verlag.

Smith, H.., Fingar, P., 2003. Business Process Management: The Third Wave. Meghan-Kiffer press, Tampa, Florida USA.

Tumay, K., 1996. Business Process Simulation. In *Proc. 27th conference on Winter Simulation Conference*, Virginia, U.S.A, 1996, pp. 55-60.

Van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M., 2003. Business Process Management: A survey. In *LNCS Springer-Verlag*, Berlin Heidelberg, pp. 1-12.

Van der Aalst, W.M.P., ter Hofstede, H.M, 2002. Workflow Patterns: On the Expressive Power of (Petri-net-based) Workflow Languages. In *Proc. Fourth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools,* K. Jensen (Ed), vol. 560, pp. 1-20.

Vanderfeesten, I., Cardoso, J., Mendling, J., Reijers, H.A., and van der Aalst, W.M.P, 2007. Quality Metrics for Business Process Models. In: L. Fischer, ed., Workflow Handbook 2007, Workflow Management Coalition, Lighthouse Point, Florida, USA.

Visual Paradigm, 2007. Visual Paradigm: Introduction to Business Process Visual Architect. In *BP-VA user's guide*, Chapter 1.

W3C, 2004. The Forms Working Group, official website, *http://www.w3.org/ MarkUp/Forms/*, accessed 2008.

Wand, Y., Weber, R., 2002. Research Commentary: Information Systems and Conceptual Modeling - A Research Agenda. In *Information System Research*, Vol. 13, No. 4, pp. 363-376.

Wang, W., Ding, H., Dong J., Ren C., 2006. A comparison of Business Process Modeling Methods. In *SOLI'06, Proc. IEEE International Conference on Service Operations and Logistics, and Informatics*, pp. 1136-1141.

Weske, M, Van der Aalst, W.M.P., Verbeek, H.M.W., 2004. Advances in business process management. In *Data & Knowledge Engineering*, Elsevier, Nro. 50, pp. 1-8.

White, S.A., 2004. Introduction to BPMN. Business Process Trends.

Wohed, P., Van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M., Russell, N., 2006. On the suitability of BPMN for Business Process Modelling. In *LNCS Springer-Verlag*, Berlin Heidelberg, pp. 161-176.