Discovering Business Models for Software Process Management
An Approach for Integrating Time and Resource Perspectives from Legacy Information Systems

C. Arevalo, I. Ramos and M. J. Escalona
Department of Computer Languages and Systems, University of Seville, Avda. Reina Mercedes s/n, 41012 Seville, Spain

Keywords: Software Process Management, BPMN, Interoperability, Legacy Systems, Reverse Engineering, Model-Driven Engineering.

Abstract: Business Process Management (BPM) is becoming the modern core to support business in all type of organizations and software business is not an exception. Software companies are often involved in important and complex collaborative projects carried out by many stakeholders. Each actor (customers, suppliers or government instances, among others) works with individual and shared processes. Everyone needs dynamic and evolving approaches for managing their software projects lifecycle. Nevertheless, many companies still use systems that are out of the scope of BPM for planning and control projects and managing enterprise content (Enterprise Content Management, ECM) as well as all kinds of resources (ERP). Somehow systems include scattered artifacts that are related to BPM perspectives: control and data flow, time, resource and case, for example. It is aimed to get interoperable BPM models from these classical Legacy Information Systems (LIS). Model-Driven Engineering (MDE) allows going from application code to higher-level of abstraction models. Particularly, there are standards and proposals for reverse engineering LIS. This paper illustrates LIS cases for software project planning and ECM, looking at time and resource perspectives. To conclude, we will propose a MDE-based approach for taking out business models in the context of software process management.

1 INTRODUCTION

Software companies, as all companies operating in the current world, are involved in complex changes due to new globalization rules, such as joint ventures, mergers and acquisitions, as well as in new collaboration models among stakeholders such as outsourcing, offshoring or near shoring, among others. Today, most of these organizations take into consideration the Business Process Management paradigm (BPM) (Van der Aalst, 2004) (Dumas et al., 2013) (Weske, 2012) to support their software business processes that is, managing all kind of software project lifecycles. Such processes are more unpredictable (Ruiz-González et al., 2004) than other production processes, since they are continuously changing and tightly tied to communication among single people, working teams protocols and companies contracts. These organizations also use models, systems, standards and best practices in the market. In consequence, they must adapt their knowledge to the requirements of a global and changing environment, by using multiple kinds of sophisticated and automated tools at once, such as planning and control projects, documents and content management, together with all kinds of resources (human, technological, economical or material, for instance) to gain competitive advantage in the market. These systems constantly evolve for supporting new paradigms, architectures, methodologies and languages, as well as different databases. They should be tailor-made systems or customizable market products (like ECMs and ERPs) with different languages, data structures and details of each business process. Managing all these complex organizational factors simultaneously is seemed as a big challenge. Even though most of these Legacy Information Systems (LIS) (Bisbal et al., 1999) are not BPM-oriented, we need to use this new approach for viewing them as new business models composed of a set of integrated processes. A BPM model will offer *interoperability* among different systems. Geraci (Geraci, 1991) defines this term as *"Ability of a system or a product*

Arevalo C., Ramos I. and Escalona M..
Discovering Business Models for Software Process Management - An Approach for Integrating Time and Resource Perspectives from Legacy Information Systems.
DOI: 10.5220/0005454903530359
In *Proceedings of the 17th International Conference on Enterprise Information Systems* (ICEIS-2015), pages 353-359
ISBN: 978-989-758-096-3

to work with other systems or products without special effort on the part of the customer. Interoperability is possible by the implementation of standards". Each organization will take advantage of this ability to get higher flexibility in its internal business process and will share it with other actors in big projects, keeping strong independence from the architecture of information systems.

The Process Mining Manifesto (Van der Aalst et al., 2012) identifies different perspectives in a BPM system: control-flow, data-flow, organization and resource, case and time perspectives. If we had a set of LIS and we aimed to represent an organization's these unwritten business processes, we could elaborate a conventional modeling business through interviews, document, information flows or business rules, among other tools. Now, if possible, we would like to consider business models derived from structures and rules already existing in each LIS. In software projects context, experts often use systems for planning and control (Hansen and Hansen, 2014), (Lang, 2010) and collaborative tools for ECM (Shariff, 2013), since both contain data structures and rules that are closely linked to business processes time (Pmbok, 2013) and resource perspectives. Firstly, we will work with these systems, but we could extend the study scope to many systems with other perspectives. These LIS include events, tasks, resources or resource groups, documents, code, and business rules, such as rules of time for planning and controlling tasks and rules of resource allocation, approval tasks or workflows. Such elements enable getting, through reverse engineering, core business processes that subsequently the expert can examine and polish.

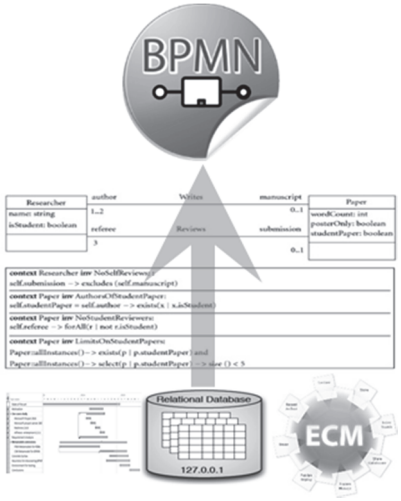

Figure 1: Extracting Business Models.

Figure 1 shows the scenario for taking out a target BPMN model from these LIS source. LIS persistence layer usually involves relational database structures and constitutes the most stable view of business along processes lifecycle. We are interested in database artifacts linked to a process time and resource perspectives. For this aim, this article suggests using the Model-Driven Engineering Paradigm (MDE) (Schmidt, 2006).

The paper is structured in different sections as follows: a) related work, b) taxonomy for time and resource allocation rules, c) our MDE-based approach for reverse engineering LIS and d) conclusions and future work.

2 RELATED WORK

We have explored standards and research work regarding Model-Driven Architecture (MDA), software process management, time and resource perspectives in BPMN and reverse engineering LIS databases as follows:

Firstly, MDA (MDA, OMG 2011) is the major representative of the MDE paradigm that proposes different level of abstraction to gain independency and interoperability (Elveszeter and Berre, 2010) among different views of a universe of discourse. It defines Platform Specific Models (PSM), Platform Independent Models (PIM) and Computer Independent Models (CIM), where PSM is close to the application code and CIM is found at business expert level. Meta Object Facility (MOF, OMG 2011) introduces the metamodel concept and Query and View Transformation (QVT, OMG 2011) is a language for mapping artifacts of different abstractions.

Regarding Model-Driven Development (MDD) (Mellor et al., 2003) and other methodologies, some studies, such us (Valderas and Pelechano, 2011) present a relevant current global view about MDE methodologies. It is not the aim of this paper to depict all of them, but we illustrate our work with Navigational Development Technique (NDT) (Escalona et al., 2008), which is supported by a set of tool-case grouped in NDT-Suite (García-García et al., 2012) and NDTQ-Framework. This methodology has being successfully used in several software development projects such as (Cutilla et al., 2011), (Escalona et al., 2013) and (Salido et al., 2014), which enable us to have a good set of data and examples.

Secondly, (García-Borgoñón et al., 2014) carries out a Systematic Literature Review (SLR) for

Software Process Management that covers software process modeling languages, methods, standards and best practices. They briefly conclude that Business Process Model and Notation (BPMN OMG, 2011) is the selected technology, among other approaches, to model processes, because it offers simplicity, standardization and support for execution processes. Many UML-based approaches have been developed, but there are actually two main languages: Software & Systems Process Engineering Metamodel Specification 2.0 (SPEM, OMG 2008), which provides a language for software methodologies and (ISO/IEC 24744, 2007), which offers guidelines to identify process models as well as improve consistency and uniformity when defining these standards.

BPMN and MDD are becoming the key elements of the aforementioned standards for software management. Business Process Definition Metamodel (BPDM, OMG 2008) provides interoperability by means of different tools, so we will work with a metamodel for business processes based on BPMN, UML activity diagrams (UML, OMG 2011) and BPDM.

Thirdly, with respect to time perspective in BPMN, researchers agree on its weakness to show this dimension. (Flores and Sepúlveda, 2011) propose patterns to express time rules included in a Gantt chart. They work with BPMN 1.2 supported artifacts. Besides, (Gagné and Trudel, 2008) focus on Allen's Interval Algebra, extending BPMN diagrams with decorators in Time-BPMN (Gagné and Trudel, 2009), where the start and end task events and associations among them are semantically redefined to display these constraint types. (Awad et al., 2009) recommend a set of Workflow Resource Patterns (WRP) over processes, working with an extension of BPMN metamodel for the resource perspective and OCL rules as expression of allocation rules.

Finally, LIS maintenance can be too expensive because these systems are not always well documented. Additionally, if they are highly complex, they are not easy to replace for a new system (Bisbal et al., 1999), (Heuvel, 2006). OMG Architecture-Driven Modernization (ADM) (Ulrich and Newcomb, 2010) is an approach based on MDA that identifies metamodels and processes for LIS modernization. It is also important within MDA to review the Information Management Model (IMM, OMG 2006) as a proposal of interoperability that, among others, establishes metamodel views for tables, keys, routines and triggers (Eisenberg, 1996) of a SQL relational database (Melton and Simon, 2002). We have also revised, among other sources, the work by (Boronat et al. 2005) and (Pérez-Castillo et al., 2011 and 2012) for relational database schema reverse engineering (the latter uses the concept of process archaeology and code mining) and by (Izquierdo et al., 2008 and 2012) where its main example is a DSL tool named Grammar to Model Language (Gra2Mol), which acts as a bridge going from grammar-ware to model-ware. These studies are based on an ADM-approach and use basic relational SQL database metamodels for extracting tables and keys, such as UML classes and associations. Other studies, for example by (Demuth et al., 1999 and 2001) and (Arevalo et al., 2014), also take out ECA rules hardcoded as triggers (Eisenberg, 1996) for enriching UML class diagrams with Object Constraint Language (OCL, OMG 2014) assertions.

3 TIME AND RESOURCE RULES

There are several interesting characteristics derived from databases to define our BPM, as it is mentioned in subsequent work. Basically, this paper mainly focus on time and resource perspectives, although it enables the taxonomy for business rules we are searching in each LIS. However, other kind of rules may appear when considering other perspectives, but they are out of the scope of this analysis. We just aim to find Time Constraints and Resource Allocation Constraints. Figure 2 shows a rule taxonomy.

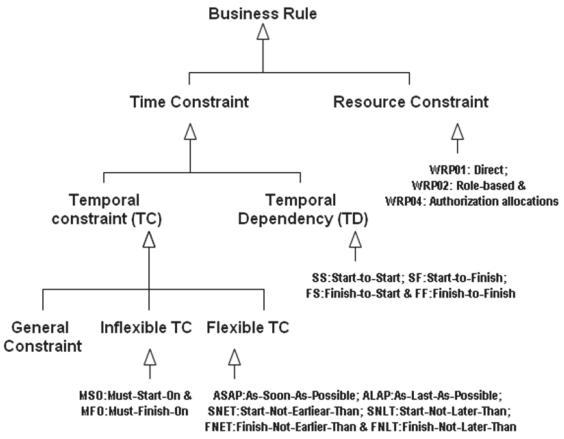

Figure 2: Business rules taxonomy.

As Time Constraints concerns, it defines rules over a task, a process or a sub-process or precedence dependencies between pairs of tasks, processes and sub-processes. They are shown as Temporal Constraints (TC) and Temporal Dependencies (TD), and in turn, (TC) is subdivided into (a) General Constraints, (b) Inflexible TC and (c) Flexible TC.

All kinds of rules, except General Constraints, are founded in (Pmbok, 2013), (Flores and Sepúlveda, 2011) and (Gagné and Trudel, 2008 and 2009).

(a) Firstly, General Constraints provide chronological rules between the start and finish events of an activity. Furthermore, time-interval must be in the scope of parent activity (process or sub-process).

(b) Secondly, Inflexible TC fix the start and finish task events: *'MSO: Must Start On'* and *'MFO: Must Finish On'*.

(c) Last, Flexible TC enable a floating start and finish task events; thus *'ASAP: As Soon As Possible', 'ALAP: As Last As Possible', 'SNET: Start Not Earlier Than', 'SNLT: Start Not Later Than', 'FNET: Finish Not Earlier Than' and 'FNLT: Finish Not Later Than'* allow advancing or delaying a task, whenever the critical path of the total process is not affected.

TD constraints govern previous relations between events of a preceding task and events of a current task. Thus, the four possible combinations are: *'SS: Start to Start', 'SF: Start to Finish', 'FS: Finish to Start'* and *'FF: Finish to Finish'*.

With respect to Resource Allocation Constraints (Awad et al., 2009) propose a metamodel to capture resource and case perspectives of a business process and also the OCL specification of nearly thirty Workflow Resource Patterns (WRP). Nevertheless, we will find simple patterns like *(WRP01) Direct Allocation, (WRP02) Role Allocation* and *(WRP03) Authorization Allocation* in software systems such as project planning tools and ECMs. More complex allocation patterns are associated with case perspective in a process that needs references to individual task instances.

4 EXTRACTING BUSINESS MODELS

Our approach stands as a model-driven solution. If we consider time and resource perspectives to enrich our heuristic process, we will enhance the representation of business processes in comparison with the models obtained by means of typical ADM approaches (see Section 2). However, in order to illustrate the grade of abstraction and make it understandable, we present our approach in the context of MDA working at different MDA levels of abstraction: (i) Target PIM BPMN models with time and resource perspectives and (ii) Source PSM Relational Models including table structures, constraints and rules at lower level, but related to the same perspectives. A specific heuristics with QVT transformations will allow applying reverse engineering to turn PSM models into PIM models. Nonetheless, we will need specific metamodels at both levels.

4.1 BPMN Metamodel Extensions

BPMN metamodel extensions are necessary to capture time and resource semantics, and later on, other perspectives may be included with new extensions. We only work with BPMN 2.0 and OCL 2.4 standard specifications, therefore transforming models into executable code will be easier.

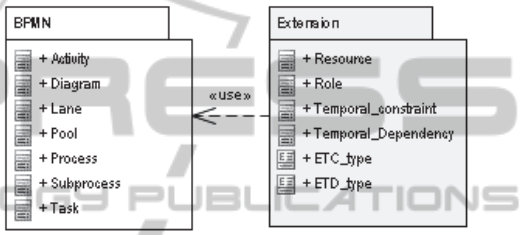

Figure 3: PIM Packages.

Figure 3 shows two packages: *BPMN* for standard selected artifacts and *Extension* for new time and resource-related classes.

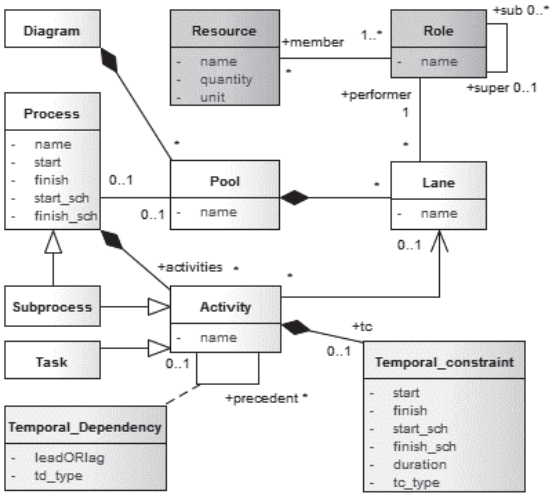

Figure 4: BPMN metamodel extensions.

In Figure 4 *Process* and *Activity* have dates (real and scheduled) associated to start and finish task events; an activity may have *TCs* and *TDs* with preceding activities; a TD may have a delay interval with respect to its predecessor: *lead (-)* or *lag (+)*. *Resource* and *Role* support the basic allocations rules

(WRP01, WRP02 and WRP03) with OCL rules (Awad et al., 2009). The latter is associated with the *Lane* class to determine the *performer* of an *Activity*.

4.2 Heuristics from PSM to PIM

We will use the PSM metamodels capabilities relational database proposed, among others by (Arevalo et al., 2014) and (IMM, OMG 2006), to take out table schemas and constraints. The metamodel uses two packages, Generic Abstract Syntax Metamodel (GASTM) and Specific Abstract Syntax Metamodel (SASTM), to capture enough behavior from each concrete LIS relational model, i.e.: Oracle, MS-SQL*Server, PostgreSQL and MySQL, for instance, that have their own procedural languages to code triggers with PL/SQL, Transact-SQL, pgSQL and MySQL stored procedures as variations of ISO SQL/PSM (Eisenberg, 1996).

We suggest a model-to-model (M2M) (MDA OMG, 2011) method to transform structures from LIS databases at PSM level into target PIM BPMN metamodel (see Figure 5). It will:

(i) Select a table from each LIS database that is linked to time and resource perspectives. Map tables and foreign keys to metamodel classes and associations. Map table columns to classes attributes.

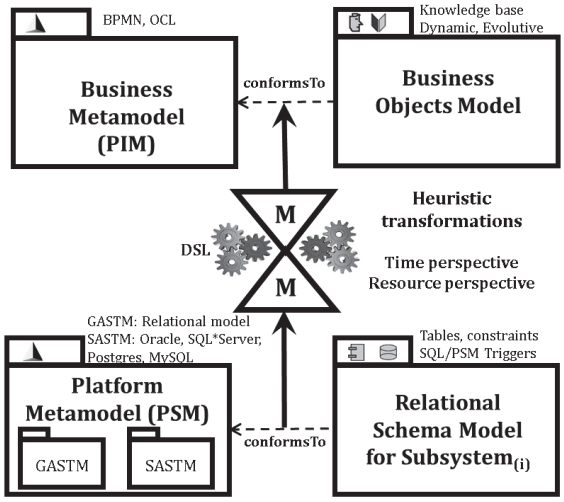

Figure 5: Reverse engineering from PSM to PIM.

(ii) Map database constraints to OCL business rules given by the classes taken out, considering studies like those by (Demuth et al., 1999 and 2001) and (Arevalo et al., 2014) for this purpose.

(iii) Infer new classes and OCL constraints to capture time and resource perspectives given on LIS systems, taking into account the taxonomy rules defined in Section 3.

This proposal includes models, metamodels, a Domain Specific Language (DSL) and heuristic transformations where: (a) Every model conforms to its metamodel; (b) Metamodels, both at PSM and PIM level are defined with a DSL based on UML2 profiles; (c) The procedure transforms artifacts among different levels of abstraction and it also merges different source models to get a unique and global business data object view according to the referenced PIM metamodel.

5 CONCLUSIONS AND FUTURE WORK

BPMN is essential for software organizations. Consequently, this paper proposes a MDE-approach to find out business models from LIS, such as project planning and ECM systems. For this aim, we propose the following points:

(i) Work with time constraints and resource allocation rules gathered from project planning and ECM systems. Derive a BPMN PIM view with more expressive structures and rules that ADM approaches could apply to generic LIS systems.

(ii) Work with a target metamodel based on actual versions of BPMN and OCL, allowing extensions with more resource allocation rules, new perspectives and a process polishing in liaison with the business expert.

Nowadays, we are working in OCL formalization of rules taxonomy. More LIS systems may constitute sources for extracting and merging process models. Therefore, we will work with study cases with the aim of evaluating this approach and going further with the analysis of more perspectives rather than time and resource.

ACKNOWLEDGEMENTS

This research has been supported by the MeGUS project (TIN2013-46928-C3-3-R) of the Ministry of Economy and Competitiveness, Spain.

REFERENCES

Arevalo, C., Gómez-López, M. T., Reina Quintero, A.M., Ramos, I., 2014. *An Architecture to Infer Business Rules from Event Condition Action Rules Implemented in the Persistence Layer*. In R. Pérez-Castillo, & M. Piattini (Eds.) Uncovering Essential Software Artifacts

through Business Process Archeology (pp. 201-221). Hershey, PA: BSR.

Awad, A., Grosskopf, A., Meyer, A., & Weske, M., 2009. *Enabling resource assignment constraints in BPMN*. Hasso Plattner Institute, Potsdam.

Bisbal, J., Lawless, D., Wu, B., Grimson, J., 1999. *Legacy Information Systems: Issues and Directions.* 0740-7459/99, ©IEEE.

Boronat, A., Carsí, J.A., Ramos, I. 2005. *Automatic Reengineering in MDA Using Rewriting Logic as Transformation Engine*. CSMR 2005: 228-231.

BPDM, OMG 2008. *Business Process definition Metamodel (BPDM)*, http://www.omg.org/spec/BPDM/

BPMN, OMG 2011. *BPMN V2.0*. http://www.omg.org/spec/BPMN/2.0/

Cutilla, C. R., García-García, J. A., Alba, M.; Escalona, M.J., Ponce, J., Rodríguez, L., 2011. *Aplicación del paradigma MDE para la generación de pruebas funcionales*; *Exp. AQUA-WS*. 6ª Conf. Iberic. S.T.I., Demuth, B., Hußmann, H., 1999. *Using UML/OCL constraints for relational database design*. UML. doi:10.1007/3-540-46852-8_42.

Demuth, B, Hußmann, H., Loecher,S., 2001. *OCL as a specification language for business rules in database applications*. UML. doi:10.1007/3-540-45441-1_9.

Dumas, M., La Rosa, M., Mendling, J., Reijers, H. A., 2013. *Fundamentals of business process management*. Heidelberg: Springer.

Elveszeter, B., Berre, A.J., 2010. *OMG Specifications for Interoperability*. IESA I.C. (p. 31). John Wiley & Sons.

Eisenberg, A., 1996. *New standard for stored procedures in SQL*. ACM SIGMOD Record 25 (4): 81–88.

Escalona, M.J., Aragón, G., 2008. *NDT. A Model-Driven Approach for Web Requirements*. IEEE TSE, vol. 34, no. 3, pp. 377-390.

Escalona, M.J., García-García, J. A., Más, F., Oliva, M., Del Valle, C., 2013. *Applying model-driven paradigm: CALIPSOneo experience*. CAiSE'13, vol. 1017, pp. 25-32.

Flores, C., Sepúlveda, M., 2011: *Temporal Specification of Business Processes through Project Planning Tools*. In BPMW (pp. 85-96). Springer Berlin Heidelberg.

Gagné, D., Trudel, A., 2008: *The Temporal Perspective: Expressing Temporal Constraints and Dependencies in Process Models*. Process Models in BPM and Workflow Handbook.

Gagné, D., Trudel, A., 2009. *Time-bpmn*. In CEC'09 IEEE Conference on (pp. 361-367).

García-Borgoñón, L, Barcelona, M.A., García-García, J.A., Alba, M., Escalona, M.J., 2014. *Software Process Modeling Languages: a Systematic Literature Review*. ISS Vol. 56. 103-116.

Geraci, A., 1991. *IEEE Standard Computer Dictionary.* IEEE Press, Piscataway, NJ, USA.

Hansen, C., Hansen, P. K., 2014. *Microsoft Project 2013*. Target Gruppens Forlag.

Heuvel, W., 2006. *Aligning Modern Business Processes and Legacy Systems: A Component-Based Perspective.* CIS**,** MIT Press.

IMM, OMG 2006. *Information management metamodel (IMM),* http://www.omgwiki.org/imm/doku.php.

ISO/IEC 24744, 2007. *Software Engineering Metamodel for Development Methodologies.*

Izquierdo, J.L.C., Cuadrado, J. S., Molina, J. G., 2008. *Gra2MoL: A domain specific transformation language for bridging grammarware to modelware in software modernization.* In Workshop on MDSE.

Izquierdo, J.L.C., García Molina, J., 2012. *Extracting models from source code in software modernization*. Software & Systems Modeling.

García-García, J. A., Ortega, M. A., García-Borgoñón, L., & Escalona, M. J., 2012. *NDT-Suite: a model-based suite for the application of NDT*. Web Engineering (pp. 469-472). Springer Berlin Heidelberg.

Lang, J. P., 2010. Redmine-Overview.

Mellor, S. J., Clark, A. N., Futagami, T., 2003. *Model-driven development - Guest editor's introduction*. IEEE Softw. 20, 5, 14–18.

Melton, J., Simon, A.R., 2002. *SQL:1999*. Morgan Kaufmann. pp. 541–542. ISBN 978-1-55860-456-8.

MOF, OMG 2011. *Meta Object Facility (MOF)*, http://www.omg.org/spec/MOF/2.4.1.

OCL, OMG 2014. *Object Constraint Language (OCL), Version 2.4.* http://www.omg.org/spec/OCL/2.4/.

Pérez-Castillo, R.; Piattini, M. et al., 2011. *MARBLE. A business process archeology tool.* ICSM 2011:578-581.

Pérez-Castillo R., Piattini, M. et al., 2012. *Database schema elicitation to modernize relational data-bases*. ICEIS.

Pmbok, 2013. *A Guide to the Project Management Body of Knowledge (Pmbok Guide)*. 5th Edition. Project Management Institute.

QVT, OMG 2011, *Meta Object Facility (MOF) 2.0 QVT*. http://www.omg.org/spec/QVT/1.1/.

Ruiz-González, F., Canfora, G., 2004. *Software Process: Characteristics*. T&E., vol. 5, pp. 5-10.

Salido, A., García-García, J.A., Gutiérrez, J.; Ponce, J., 2014. *Tests Management in CALIPSO-neo: A MDE Solution*. Journal of SE&A.

Schmidt, D.C., 2006. *Model-Driven Engineering*, Computer, vol.39, no.2, pp.25-31.

Shariff, M., 2013. A*lfresco 4 Enterprise Content Management Implementation*. Packt Publishing Ltd.

SPEM, OMG 2008. *Software & Systems Process Engineering Metamodel Specification (SPEM 2.0).* http://www.omg.org/spec/SPEM/2.0.

Ulrich, W. M., & Newcomb, P., 2010. *Information Systems Transformation: Architecture-Driven Modernization Case Studies*. Morgan Kaufmann.

UML, OMG 2011, Unified Modeling Language (UML 2.0). http://www.omg.org/spec/UML/.

Valderas, P., Pelechano, V., 2011. *A Survey of Requirements Specification in Model-Driven Development of Web Applications*. TWEB 5(2):10.

Van der Aalst, W., 2004. *Business process management: a personal view*. BPM Journal, vol. 10, no. 2, p. 5.

Van Der Aalst, W. et al., 2012. *Process mining manifesto*.

In BPM workshops (pp. 169-194). Springer.
Weske, M., 2012. *Business Process Management. Concepts, Languages, Architectures*, 2nd Edition. Springer.