APPLICATION OF SERVICE-ORIENTED COMPUTING AND MODEL-DRIVEN DEVELOPMENT PARADIGMS TO BUSINESS PROCESSES

A Systematic Review

Andrea Delgado
Computer Science Institute, Faculty of Engineering, University of the Republica, Montevideo, Uruguay

Francisco Ruiz, Ignacio García-Rodríguez de Guzmán, Mario Piattini
Alarcos Research Group, Technologies and IS Department, University of Castilla-La Mancha, Ciudad Real, Spain

Keywords: Business Process Management (BPM), Service-oriented Computing (SOC), Model-driven Development (MDD), Systematic Review.

Abstract: To achieve the defined value for their businesses, current organizations need to manage their business processes in an integrated manner, interconnecting the software systems that support these processes. Over the last few years, new paradigms have appeared to respond to this and other organizational and software needs: Business Process Management (BPM) and Service-Oriented Computing (SOC) which are closely interconnected. Additionally, the Model-Driven Development (MDD) paradigm has been called upon to play an important role in supporting business process implementation by software services. BPM handles the management of business processes, including their modelling, deployment, execution, analysis and improvement. Service-Oriented Computing bases software development on services, which correspond to business concepts and are created in order to perform business processes. Model-Driven Development promotes software development based on models which enable, among other things, transformations and the automatic generation of code for different platforms. With the aim of establishing the bases for research into the integration of these paradigms to support business process management in organizations, a systematic review was carried out, focusing on the current state of the literature concerning the application of service-oriented and model-driven paradigms to business processes.

1 INTRODUCTION

The software integration effort towards realizing horizontal business processes that give value to business has been an important issue over the last few years. Due to the verticality and complexity of software systems generally implemented in different technologies, integrating them has been a difficult task. One challenge is to unite the vision from business and software areas, to design and implement business processes in a way that allows us to react agilely to changes, mainly in two aspects: business and technologies. New paradigms have appeared to support this: Business Process Management (BPM), Service-Oriented Computing (SOC) and Model-Driven Development (MDD). BPM refers to the set of activities that organizations perform to optimize or adapt their business processes to new organizational needs; BPM Systems (BPMS) are tools supporting their integral management, from modelling to execution in a process engine (BPMI) (Smith et al, 2003). SOC bases the design of applications on services, which are software reusable elements through which providers and consumers interact in a decoupled way to perform business processes in defined sequences of invocations to services (orchestration, choreography). SOA is a realization of SOC (Erl, 2005) (Krafzig et al, 2005) (Papazoglou et al, 2007). Model-Driven Development (MDD) bases software development on models, whose elements are described using metamodels, models and languages which allow the automatic transformation between them along with the generation of code for different

Delgado A., Ruiz F., García-Rodríguez de Guzmán I. and Piattini M. (2010).
APPLICATION OF SERVICE-ORIENTED COMPUTING AND MODEL-DRIVEN DEVELOPMENT PARADIGMS TO BUSINESS PROCESSES - A Systematic Review.
In *Proceedings of the 5th International Conference on Software and Data Technologies*, pages 51-62
DOI: 10.5220/0003002700510062

platforms. MDA is a specific realization of MDD (MDA, 2003) (Mellor et.al, 2003) (Stahl et al., 2006)

In this article, a systematic review of the application of service-oriented and model-driven paradigms to business processes is presented, with the main objective of providing the basis for the research into business process improvement. The research path we have defined is that of the methodological application of service-oriented and model-driven paradigms to business processes. The focus is on disciplined, conceptually based and standardized associated practices, not on a specific tool or interpretation. This will allow us to maximize the value of the understanding and application of these paradigms jointly, to obtain an organization's defined business value. The rest of the article is organized as follows: section 2 briefly describes the procedure followed in carrying out the systematic review and its main elements; in section 3, relevant information extracted from the selected primary studies is presented, and in section 4, the main principles in paradigm integration identified in the analysis of the studies are discussed. Finally, section 5 presents some conclusions.

2 REVIEW PROCEDURE

Following the method proposed by Kitchenham (Kitchenham, 2004), a systematic review consists of three stages: planning the review, development of the review and publication of the results. These stages have several elements, starting from the definition of the research question, key words and research chains, the execution of sources for the defined chains, and the inclusion and exclusion criteria to select relevant and primary studies from which to extract the associated data. The defined objective is to identify the joint application of SO and MD paradigms to business processes, which allows us to assess aspects related to their use.

2.1 Research Question

The research question that guides this work includes these specific terms in its initial formulation: SOC, MDD, MDE, BP and BPM. In the update done in June 2009, it was extended to include new terms like Service-Oriented Development (SOD). The research question, then, is:

Which methodological/conceptual applications of Service-Oriented (Computing or Development) and Model-Driven (Development or Engineering) paradigms to business processes and the Business Process Management (BPM) paradigm have been carried out?

The main interest is in the methodological and conceptual levels of application, not in technological aspects or specific tools for implementation. The main identified key words for carrying out the searches are: Service-Oriented Computing and Development (SOC, SOD), Model-Driven Development and Engineering (MDD, MDE), Business Process and Business Process Management (BP, BPM). The research question, keywords and search strings were validated by experts.

2.2 Search String and Sources

The criteria for selecting the sources included the importance of the source as a repository of scientific articles, possibility of access and web site, resulting in the following selections: ACM Digital Library (http://portal.acm.org), IEEE Digital Library (http://www.computer.org), SCOPUS document db (http://www.scopus.com), Science@Direct in CS area (http://www.sciencedirect.com), WileyInterScience in CS area (http://www.interscience.wiley.com).

The search strings were obtained from the combination of the defined keywords, mainly with the Boolean operands "AND" and "OR". The main search string was constructed by combining the defined key words, obtaining the following general search string, which was adapted for each search engine of the selected sources:

("service-oriented computing" OR "service-oriented development") AND ("model-driven development" OR "model-driven engineering") AND ("business process" OR "business process management")

We decided not to include words like "conceptual" and "methodology" since generally, the approach is not explicitly stated in those terms, although it clearly appears in a reading of the study.

2.3 Study Selection

To select the studies, inclusion and exclusion criteria had to be defined, according to the research question and the search strings defined. The inclusion criteria were first applied to select a reduced set of relevant studies to which to apply the exclusion criteria to select the primary studies, which responded to the research question. To be selected, the proposals had to deal with the methodological and/or conceptual application of SOC and MDD paradigms to business processes, presenting a methodology, unified proposal relating the paradigms, transformations and relations between business processes and service models and notations. Once the articles were filtered by reading the title, abstract and key words of all of the articles obtained, the relevant ones were selected according to the inclusion criteria. The exclusion criteria were then applied, by reading the articles thoroughly to select the primary studies with relevant information about the research issue. Studies were excluded if they focused on technology issues, or composition and/or generation of Web Services (WS) and languages for execution as WS-BPEL (WS-BPEL) or implementation with workflows (WfMc), integrated other paradigms such as agents or grid computing, dealt solely with one paradigm or did not include business processes as their main focus.

Table 1: Results obtained for the selected sources.

Source	Found with rep.	Relevant not rep.	Primary
ACM DL	259	3	3
IEEE DL	779	10	8
SCOPUS	264	8	5
Science Direct	503	5	2
Wiley InterScience	8	0	0
Subtotal (2000-2007)	1813	26	18
ACM DL	426	3	0
IEEE DL	300	8	4
SCOPUS	327	11	7
Science Direct	499	3	1
Wiley InterScience	4	0	0
Subtotal (2008-2009)	1556	25	12
Total (2000-2009)	3369	51	30

After eliminating the repeated articles and reading the title, abstract and keywords of the resulting articles, 26 were selected as relevant in the first search, which corresponds to the period 2000 to 2007, and 18 of those were selected as primary studies. In the review update which corresponds to the period 2008 to June 2009, 25 studies were selected as relevant, of which 12 were primary studies, creating a total of 51 relevant non-repeated studies from which 30 were selected as primary studies. It is clear that although the total quantity of articles found is similar for both periods, the first result covers a period of eight years, while the second covers a period of only one and a half year, demonstrating the increasing interest and work being done in the subject in recent years. The high number of total articles found decreased to 1.313 after the elimination of repeated articles between searches and sources. The selected primary studies are shown in the Appendix, Table 2.

2.4 Information Extraction

Once the primary studies were selected, the relevant information was extracted. To perform this task, a form consisting of many sections was defined, including general data such as title, publication and authors with affiliations and associated country, year of the study, general study description, and an indication of which paradigms it investigated. Several key aspects of paradigm integration were found in the process of information extraction, for which the studies were classified; the approaches are presented and discussed in section 4.

3 ANALYSIS OF THE RESULTS

This section analyzes and discusses the contents of the selected primary studies in order to extract, organize and present the relevant information.

3.1 Paradigm Integration

Figure 1 (a) shows the publications by paradigm integration. It can be seen that of the total number of studies, nearly half (47%) correspond to the application of service-oriented and model-driven development paradigms to business processes, and the other half is divided between the application of service orientation to business processes (30%) and the application of model-driven paradigms to business processes (23%). These results are consistent with last year’s tendency to unify the application of business processes and service-oriented paradigms to support organizational needs, and the increase in research in the last few years to provide automated support for this integration.

3.2 Trends in Publication

Figure 1 (b) shows the trends in publications for paradigm integration. As it can be seen, the greater number corresponds to the years 2006 and 2008, bearing in mind that the data shown for 2009 only reflect the studies found before July of that year, and the low number for 2007 could be related to the growth in technical publications that we were not interested in. It can be concluded that after a first period of incipient attention in business process support by services and model-driven development, this has gained increasingly more attention not only

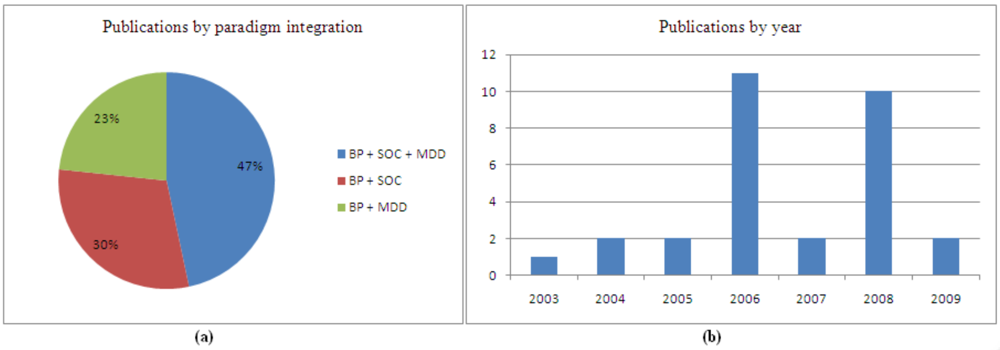

Figure 1: Publications by paradigm integration (a) and by year (b).

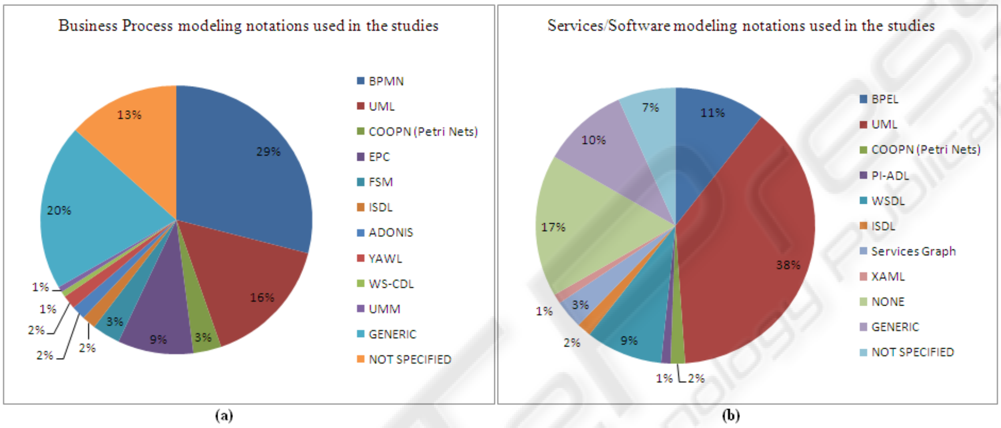

Figure 2: Business Process (a), and Service/Software (b) modelling notations used in the studies.

from academia but also from organizations, as real project applications of the proposals have been conducted, as presented in some studies.

3.3 Notations Used

Regarding business process and service modelling, one of the most important issues is the notations used to specify them. Many efforts have been made to define notations to support the different views of software development and business process integration. Different notations can be used for each identified need ranging from business process modelling to execution, simulation, the assessment of desired and undesired properties for the models, specification of services, service interaction and composition. As can be seen in Figure 2 (a), the most widely used notation for business process modelling is BPMN (BPMN) (29%) and the most used notation for service/software modelling is UML (UML) (38%) shown in Figure 2 (b), the former being the main standard for business process modelling, since its adoption by the OMG in 2006, and the latter the accepted standard for software development. Other notations also used for business process modelling are UML (16%) and EPC (EPC) (9%) (Figure 2 (a)), and for service modelling for business process execution, WS-BPEL (11%) and WSDL (9%) (Figure 2 (b)) when service implementation is via WS.

In the majority of the studies, at least one notation is used or recommended for business process and service modelling, although generic notations are sometimes used or no specific notation is used at all, usually when the approach is methodological.

3.4 Type of Case Study

Of the total number of studies, 60% correspond to examples prepared to show different aspects of the proposal, generally based on standard business processes used in organizations, and 13% correspond to real projects in organizations in which the proposal has been used in joint projects with industry. On the other hand, 27% of the studies do not present a case study, generally when the case study is mentioned as future work. It can be concluded that although some work has been done on the real application of the proposals, more is needed to effectively show the benefits for organizations.

4 MAIN PRINCIPLES IN PARADIGM INTEGRATION

In this section the main principles regarding SOC, MDD and BPM paradigm integration as found in the systematic review are presented and discussed along with an illustration from the studies. The presence of each principle in the total of the selected studies is shown in Figure 3.

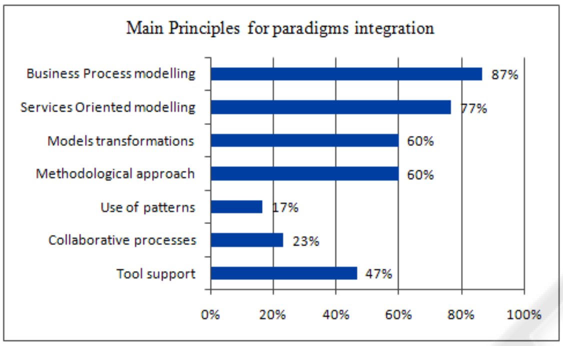

Figure 3: Main principles in paradigm integration.

4.1 Business Process Modelling

One of the most important issues for the support of business processes (BP) by service orientation (SO) in a model-driven way is how the organizations handle their business processes, especially regarding the explicit modelling in some notations, including the flow of the activities to be performed, the data exchanged and the roles involved. There are a variety of notations that can be used to specify business processes; each has advantages and disadvantages depending on the needs of the organization, the type of BP and the use intended for the model, as can be seen in Figure 2(a).

Studies modelling BP using BPMN are many, and include (Liew et. al, 2004) where the BPMN BP model is annotated with extra information to be used in transformations. (Tao Tao et. al, 2006) model BP after the analysis of needed services from system functionalities. (Henkel et al, 2005) model BP that are then realized by technical processes using existing services that match business functionality. (Gacitua-Decar et al., 2008) uses an enhanced BPMN by domain model elements and an own UML profile. (Rychly et al., 2008) models BP with BPMN then transformed into services diagram. (Thomas et al., 2008) define an initial conceptual model in EPC for process design transformed into a conceptual-technical application model in BPMN for process configuration. (Oquendo, 2008) use BPMN to model BP which are then transformed into a PI-ADL formal language. (Touzi et al., 2009) use BPMN to model collaborative BP adding a special pool called Collaborative Information System (CIS), which mediates between partnered information systems.

UML is also used as in (Mili et al, 2006) to represent and classify generic BP to be used in several domains which are then instantiated using a catalogue of software components. (Zdun et al, 2007) use AD to model BP as a long-running interruptible process called macroflow and IT-oriented processes as a short-running transactional process called microflow, where both defined layers are above a business application service layer. (Quartel et. al, 2005) uses AD to model BP mapping them into an ISDL conceptual model related with application design and implementation models via the ISDL models. (Herold et al., 2008) uses AD to model BP, and a use case diagram and business structure diagram to model other aspects of the business view. (Bruckmann et al., 2008) uses AD to model BP representing functions with actions and process with activities.

EPC is found in studies such as (Roser et al., 2006) to specify the BP from which to obtain services in UML, (Murzek et. al, 2006) and (Mendling et. al, 2006) who present horizontal transformations between different notations, the former between EPC and ADONIS and the latter between yEPC and YAWL. Other less frequently used notations are Concurrent Object Oriented Petri Nets (COOPN) in (Chen et. al, 2006) to model BP in the design phase at a PIM level and FSM (Finite state machine) in (Tao Tao et al, 2007) to model core BP, specifying generic activities and separating others applicable to a particular usage context. Some generic notations are used in studies to show activities, data and the flow of the process, but with no specific notations.

4.2 Service Oriented Modelling

Another important issue regarding service support for business processes is the service-oriented modelling approach. Once the BP are known and modelled, each defined task from the process and even the process itself have to be realized by one or a group of services. Although the flow of the execution of services can be automatically obtained from the BP model, the definition, design, model and implementation of services is of great importance for tracking their correspondence to BP and to existing or new systems providing them.

Services and software modelling is mainly done using UML, which is shown in Figure 2(b), as in (de Castro et al., 2006), where the first step is to define the needed services and from them obtain use cases, service processes to support them and then generate the required service composition. (Roser et al., 2006) model services showing the definition of services obtained for each proposed architectural approach (centralized and decentralized broker and brokerless) from the CIM description of business, (Zdun et. al, 2007) use AD at a microflow level, showing the services involved in the IT technical processes, (Gacitua-Decar et. al, 2008) categorize services into: business services abstracting activities on business entities, and technical services abstracting functionality and data provided by applications and to manage issues such as security and messaging. Although the use of the UPMS profile for services is mentioned it is not shown in the study. (Rychly et al., 2008) define a service diagram using an UML profile, specifying interfaces providing only one functionality, port, service consumer and provider, along with sequence and composition service diagrams. (Herold et al., 2008) use stereotypes such as ServiceAction, which represents the services provided by the application layer with core functionality. (de Castro et al., 2008) models the IS view at PIM level, with business services to be offered by the system as uses cases, and functionalities and process needed, (Bruckmann et al.,2008) defines software modelling with class diagrams and state diagrams. (Touzi et al.,2009) defines three views for SOA model: services for business functionalities, information for data and messages exchanged between services, and process for services interaction.

Other used notation are WS-BPEL and WSDL, as in (Thomas et al., 2008) where WS-BPEL is derived for execution from BPMN models, (Hu et al., 2003) uses WSDL to describe services, (Oquendo, 2008) uses a formal architectural language PI-ADL for SOA, including orchestration and choreography and generate WS-BPEL for process execution. Other less used notations are as in (Quartel et. al, 2005) where services are modelled using an ISDL dialect defining components that provides application services obtained from the business model, (Cauvet et al., 2008) where business services are modeled with three parts: profile (goal), structure (process) and process part (BP) and service composition resulting in a BP or BP fragment using a service composite graph, or generic ones such as in (Tao Tao et. al, 2006) where services are identified from essential functionalities of the system.

4.3 Model Transformations

Transformations between models used for the specification of BP and services are one key aspect of paradigm integration. Many approaches have been proposed to transform and generate software models from BP models, where existing languages can be used to define mappings and transformations, although new ones or different approaches are also defined. The OMG Query/Views/Transformations (QVT) standard and ATL (Jouault et al., 2006) are the most relevant examples. Transformations not only make it possible to automatically obtain elements of a target model from an origin model, but also to explicitly specify the correspondences between elements and the semantics involved.

Vertical transformations from one level of abstraction to another generally applied in a top-down way can be found in (de Castro et. al, 2006), where four PIMs are defined to model system behaviour: user services, extended use case, service process and service composition, defining mapping rules that can be completely or partially automated. In (Chen et al., 2006), embedded process controllers are generated from BP to be integrated into existing IS such as ERP via service interfaces, generating java components. (Quartel et. al, 2005) define transformations to travel from one ISDL conceptual model to another, from business to service implementation. (Mili et. al, 2006) uses a question approach defining variation points and BP variants in generic BP which are then mapped to a software components library of generic business components, to automatically assemble software systems. In (Roser et al., 2006), three different architectural approaches for software systems (centralized and decentralized broker and brokerless) are derived from a CIM description of business, establishing how services for each approach correspond to BP. (Zdun et. al, 2007) apply transformations successively based on defined patterns, starting with the macroflow-microflow pattern which establishes the conceptual basis and the process-based integration architecture pattern that guides the design of an architecture based on sub-layers for the service composition layer. (Henkel et al., 2005) propose going from BP models to technical processes matching existing services by applying transformation patterns classified with respect to the quality of the transformation. In (Gacitua-Decar et al., 2008) conceptual transformations are defined based on the successively application of patterns from the top to the bottom layer, using graphs for pattern matching. (Rychly et al., 2008) define two steps for BP to services transformation: identifying tasks in BP representing service invocations, then using a proposed technique that integrates BP and object modelling into a Business Service Model (BSM), mediating between business requirements and implementation. (Herold et al., 2008) go from a business model (CIM) to an analysis model (PIM), identifying serviceAction in tasks, then to a Design model (Architecture specific model, ASM), mapping services to the target architecture where each component provides a set of services. (de Castro et al., 2008) define transformations from a value model to use case models, defining mapping rules from the CIM to PIM level between model elements, which are automated using ATL and tools. (Bruckmann et al., 2008) use stereotypes actions that map to user and system functions in a defined metamodel described in an XML schema as interchange format and input for transformation engines. (Oquendo et al., 2008) define mapping between BPM constructs and PI-ADL for SOA expressions, showing mappings for a subset of process patterns and BPMN core elements. (Touzi et al., 2009) define two types of transformation rules: basic generation to create elements of the target model, and binding rules to generate links between them, using ATL.

Horizontal transformations on the same level of abstraction can be found in (Murzek et al., 2006) based on control flow patterns (van der Aalst, 2003), identifies patterns in the original BP, transforming each one into the target notation to obtain the target BP model, and in (Mendling et al. 2006) where transformations are based on elements of each notation and the algorithm traverses the yEPC process graph node by node to transform the BP. In (Sadiq et al., 2006) an automatic distribution of collaborative BP from an integrated one is proposed, defining the correspondences and algorithms to extract the distribute models from the integrated one.

A combination of vertical and horizontal transformations can be found in (Liew et al, 2004), where the BPMN BP model is annotated with information processed by the defined algorithms, and then transformed into several UML software artifacts: horizontal from BP to AD, vertical from BP to use cases, collaboration and deployment diagrams. (Orriens et al., 2006) define mappings between defined models in three levels that can be horizontal or vertical; using five elements capturing particular facets: what, how, where, who and when. (Thomas et al., 2008) define horizontal transformations between EPC conceptual model to BPMN conceptual-technical model, and vertical ones from BPMN to BPEL for process execution.

4.4 Methodological Approach

When modelling business processes, services and other software artifacts needed to support software development, a systematic approach to guide the development is essential. Even if some artifacts can be obtained automatically from others, a guide for the activities to be done and the flow between them, among other aspects, are a key factor for success. Software development processes have been successfully used in recent years, such as Unified Process (Jacobson et al., 1999), and approaches to include service views, activities and artifacts to guide service development have also been defined.

(Papazoglou et. al, 2006) define a methodology for SO design and development from business models, defining SO design and development principles such as service coupling, cohesion and granularity. It defines six phases: planning, analysis (process identification, process scoping, business gap analysis and process realisation) and design (service design concerns, specification, business processes), construction and testing, provisioning (service governance, certification, metering and rating, billing strategies), deployment, execution and monitoring which are traversed iteratively. (Kohlborn et al., 2009), after reviewing thirty existing service development approaches, propose a consolidated approach that combines examined methodologies and adds new items. They define two main parts for the process: the derivation of business services with four phases of preparation, identification, detailing and prioritization, and the derivation of software services to support them with phases: preparation, identification and detailing.

(Tao Tao et. al, 2006) based on the methodology (Papazoglou et al. 2006) adopts three primary phases of BP analysis, design and implementation, defining for BP analysis the steps: services identification, component identification, process scoping and process realization analysis. (Zhao et al., 2006) define the following phases: contracting, collaboration and design to define services provided and required by each organisation involved and to design and coordinate collaborations. (Herold et al., 2008) propose a model-driven approach with four phases: business development, requirement analysis, architectural design and implementation modelling, with guidelines and transformations to move from one model to the other. (de Castro et. al, 2006) defines a method for service composition with a process comprising several steps related with model generation, defining metamodels, models and artifacts to be obtained from each step, specifying activities with tasks and inputs and outputs. The business model is the general input for the process and its output the services composition model. (Gacitua-Decar et al., 2008) define steps to apply pattern techniques to successively refine BP into services that realize them, identifying business patterns in BP, and technical services. (Thomas et al., 2008) define three phases: process design, configuration and execution including in each the defined models and transformations. (de Castro et al., 2008) define an SOD-M service-oriented development method with an MDA-based approach with a process and transformations between models as in (de Castro et al., 2006). (Touzi et al., 2009) define models, metamodels and transformations to go from collaborative BP (CIM) to the SOA model (PIM) from which to generate code (BPEL), based on the PIM4SOA paradigm.

For B2B development, in (Baghdadi, 2004) a design process is proposed, consisting of six steps: BP specification, decomposition and distribution specification, mapping and validation, supporting services and components specification, logical B2B application architecture, implementation and integration technology, defining objective, input and output artifacts and models and tools.(Hu et al., 2003) define steps for going from BP definitions, implementing activities that can be internal applications or remote services provided by others, with a service mediating layer to bridge activity specifications with its implementation. (Huemer et al., 2008) define a top-down methodology based on existing approaches, starting from business and BP models to services deployment artifacts, outlining a description, notations and tools to use for each step, backed with a software factory to generate code.

Other approaches include different visions for service development, as in (Cauvet et al., 2008), which defines an iterative service composition process in which services matching BP requirements are selected and alternative services can be generated, using ontologies to match BP requirements to service goals. (Chen et al., 2008) define the BITAM-SOA framework for business-IT alignment extending ATAM, via architecture, governance and communication, defining three layers comprising specific modules, using them to guide a process model for service design, development and management, which can be top-down or bottom-up. In (Tao Tao et al., 2007), four steps in service development are defined based on core BP determination, adding a usage context creation for Configurable Context BP (CCBP) generation and service interface derivation.

4.5 Use of Patterns

Design patterns for software development are well known and have been used by the community for a long time, with the reference point being that of GoF (Gamma et. al, 1995). For business process modelling, the most relevant work is (van der Aalst et al., 2003), in which several BP constructions are defined and analyzed. The importance of reusing the best existing solutions for known problems is well established in the Software Engineering literature, so another key aspect for paradigm integration is the use of patterns at various stages of development.

Different pattern approaches are presented, as in (Zdun et al., 2007), where a pattern language with patterns and primitive patterns is defined for the integration of BP and technical processes based on services, which are applied successively in a top-down way from a macroflow defined process. In (Henkel et al., 2005), transformation patterns are defined to be used when going from BP to technical processes, applying levels of realization and realization types of BP by using existing services to match BP in lossfull, constrained, lossless and exceeded realizable transformations. (Gacitua-Decar et al., 2008) define business patterns in two types: process and domain patterns and SOA patterns; a pattern catalogue organizes them into templates.

(Murzek et al., 2006) use workflow patterns for the control flow aspect of BP models, as a basis for the horizontal transformations between different BP notations. (Oquendo, 2008) also uses process patterns to map BPMN constructs to PI-ADL expressions which are iterative and applied to the original BP for transformations to services in PI-ADL. In (Roser et al., 2006), the Broker architectural pattern is used and patterns for service interaction are modelled in UML collaborations. Other architectural patterns are also mentioned in various studies, being the most used the Layers pattern to define and organize architectural levels.

4.6 Collaborative Processes

The modelling of collaborative processes adds complexity and coordination requirements to business process models. However, collaborative process modelling is one of the most needed activities in organizations, in order to best define the way to perform their collaborative business with their partners in a coordinate and beneficial way.

(Orriens et al., 2006) present a Business Collaboration Context Framework (BCCF), capturing models in a business collaboration information model (BCIM) with three levels: strategic, operational and service level, with mappings which are then developed and managed, driven by rules which make it possible to validate and verify the alignment between model elements. In (Roser et al., 2006), SO systems realizing collaborative BP are derived from a business level in a model and architecture-driven development perspective (architecture approaches centralized and decentralized broker and brokerless) as part of the ATHENA project, where the PIM4SOA comprising a set of metamodels and tools allows the description of services and their collaborations at PIM level. (Zhao et al., 2006) provide support for BP collaborations of dynamic virtual organizations on the basis of a service-oriented relative workflow, adding definitions for IT and BP, where private information is hidden by wrapping local workflows into perceivable workflows according to visibility constraints for defined perceptions. (Touzi et al., 2009) propose a methodology for developing collaborative architectures following the MDA approach from collaborative BP to the SOA model, adding specific elements for collaborative modelling as an intermediate collaborative pool in BPMN models, and collaborative services from it.

For B2B application development, (Baghdadi, 2004) defines a four-layered architecture with four interrelated abstraction levels: business models and process, BP decomposition and distribution, supporting services and integration technology to guide the design process. (Hu et al., 2003) define a three-level conceptual framework and architecture to flexible service enactment in B2B collaborative processes, with a service-mediating layer to bridge the BP definition with its implementation with services. (Huemer et al., 2008) define three layers based on the Open-edi reference model for inter-organizational systems: business operational view (BOV) comprising business models and BP model layers, and the functional service view (FSV) comprising a deployment artifacts layer.

4.7 Tool Support

Another key aspect for paradigm integration is tool support for each development stage in the business and software development effort. To effectively help closing the business and systems gap, the use of tools that enable a smooth integration between both areas, models and artifacts is needed.

Studies that provide their own tool support include (Orriens et al., 2006) with the tool ICARUS developed to support the framework proposal, (Sadiq et al., 2006) with a BP editor that implements the algorithms for BP distribution and (Zdun et al., 2007) with a model-driven tool chain that supports modelling and model validation, using UML transformed into DSL syntax, validating models and transforming them into EMF for generating Java and BPEL code. (Oquendo, 2008) uses an own core toolset and customizable tools for PI-ADL developed previously for the Archware project.

Other studies use existing tools, adding their own support when needed, as in (Chen et al., 2006) where an IDE COOPN Builder is used to edit, visualize and verify COOPN modules, including a java and WS generator. (de Castro et al., 2006) use the Oracle BPEL development tool and own tools developed for the MIDAS framework, while in (Quartel et al., 2005) an ISDL editor and simulator is used adding a prototype of a BPEL profile for ISDL. (Mili et al., 2006) use EMF to define metamodels and the Eclipse BPEL plug-in to model BP. (Tao Tao et al., 2006) use the Oracle BPEL process manager for implementation and EJBs, and apacheAxis and MySQL as infrastructure. (Roser et al., 2006) use ARIS for BP modelling and a prototype implemented in ATHENA for the generation of services and BPEL processes, (Hu et al., 2003) uses the IBM MQSeries workflow as a basis for a prototype implementation of the proposal. (Huemer et al, 2008) evaluate the MS DSL Tools for Visual Studio and ADONIS as candidate tools for the software factory. (de Castro et al., 2008) use Eclipse with EMF for metamodel definition, GMF for model visualization and ATL for transformations, working on code generation into different WS platforms. In the same direction, (Touzi et al., 2009) use a set of Eclipse tools: Intalio designer for collaborative BP modeling, EMF for metamodel definition, ATL for transformations and TOPCASEDO for visualizing UML models. References for the tools can be found in the studies.

5 CONCLUSIONS

This article has presented a systematic review of the literature concerning the application of service-oriented and model-driven paradigms to business processes. The results obtained constitute the starting point for research into the methodological and conceptual approach of paradigm integration, which beyond a specific implementation or technology enables the systematic application of the concepts defined in the paradigms.

The selected studies present different lines of work and several important aspects of paradigm integration, which allowed us to extract the main principles that have to be taken into account when carrying out paradigm integration. Business process and service modelling have proven to be two of the most important principles regarding the relation between business and software areas. To move from one area to another, two other principles were found to provide the basis for bringing the areas closer: the methodological and the model-driven approach. The former provides guidelines to develop services systematically from business processes, while the latter, by means of mappings, rules and transformations, makes the automation of development possible, with models and metamodels being the key elements. Two important principles to improve the quality of the solutions are: the reuse of patterns for BP and service modelling which reduce errors in early stages of development, and the tool support for development, including facilities for the verification of model properties, simulation of process and execution of transformations.

Finally, it must be noticed that the search was reduced to a limited number of search engines and excluded studies which address paradigm integration but mainly in a purely technological way that did not provide any contributions in the context of this work. However, since this is such a broad area and one of growing interest, future work will include updates and reviews of other sources to include new studies that contribute to the ongoing research work.

ACKNOWLEDGEMENTS

This work was partially funded by the Agencia Nacional de Investigación e Innovación (ANII) from Uruguay, ALTAMIRA project (Junta de Comunidades de Castilla-La Mancha, Spain, Fondo Social Europeo, PII2I09-0106-2463) and PEGASO/MAGO project (Ministerio de Ciencia e Innovación MICINN, Spain, and Fondo Europeo de Desarrollo Regional FEDER, TIN2009-13718-C02-01).

REFERENCES

Baghdadi Y., 2004.ABBA: an architecture for deploying business-to-business electronic commerce applications, *In Electronic Commerce Research and Applications*, Vol. 3, Is. 2.

(BPMI), Business Process Management Initiative <http:// www. bpmi.org/>

Bruckmann T., Grunh V., 2008. AMABULO- A Model Architecture for Business Logic, In ECBS'08, 15th *IEEE International Conference and Workshop on the Engineering of Computer Based Systems*.

(BPMN) Business Process Modeling Notation, Object Management Group (OMG), 2006 - 2008.

Cauvet C., Guzelian G., 2008. Business Process Modeling: a Service-Oriented Approach, In *HICSS'08, 41st Hawaii International Conference on System Sciences*.

Chen, A., Buchs, D., A., 2006. Generative Business Process Prototyping Framework, In RSP'06, 16th International Workshop on Rapid System Prototyping.

Chen H., 2008. Towards Service Engineering: Service Orientation and Business-IT Alignment, In HICSS'08, *41st Annual Hawaii International Conference on System Sciences*.

De Castro, V., Marcos, E., López Sanz, M., 2006. A model driven method for service composition modelling: a case study. In *International Journal Web Engineering and Technology*, Vol. 2, No. 4.

De Castro V., Vara Mesa J. M., Herrmann E., Marcos E., 2008. A Model Driven Approach for the Alignment of Business and Information Systems Models, In ENC '08, *9th Mexican International Conference on Computer Science*.

(EPC) van der Aalst, W, 1999, Formalization and Verification of Event-driven Process Chains. Information and Software Technology 41, 639–650.

Erl, T., 2005. Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall.

Gamma, E., Helm, R., Johnson, R., Vlissides, J., 1995, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley

Gacitua-Decar V., Pahl C., 2008. Pattern-based business-driven analysis and design of service architectures, In *ICSOFT'08, 3rd International Conference on Software and Data Technologies*.

Henkel, M., Zdravkovic, J., 2005. Supporting Development and Evolution of Service-based Processes, In *ICEBE'05, 2nd. International Conference on e-Business Engineering*.

Herold S., Rausch A., Bosl A., Ebell J., Linsmeier C., Peters D., 2008. A Seamless Modeling Approach for Service-Oriented Information Systems, In *ITNG'08, 5th International Conference on Information Technology:New Generations*.

Hu J., Grefen P., 2003. Conceptual framework and architecture for service mediating workflow management, In *Information and Software Technology*, Vol.45, Is. 13.

Huemer C., Lieg P., Schuster R., Werthner H., Zapletal M., 2008. Inter-organizational systems: from business values over business processes to deployment, In IEEE-DEST'08, *2nd IEEE International Conference on Digital Ecosystems and Technologies*.

Jacobson, I., Booch, G., Rumbaugh, J., 1999, The Unified Software Development Process, Addison Wesley.

Jouault, F., Kurtev, I., 2006, Transforming Models with ATL (ATLAS Transformation Language), Satellite *Events at the MoDELS 2005 Conference*.

Kitchenham, B., 2004. Procedures for Performing Systematic Reviews. Software Engineering Group, Department of Computer Science, Keele University and *Empirical SE National ICT*, Australia.

Krafzig, D. Banke, K. Slama, D., 2005. Enterprise SOA, Service Oriented Architecture Best Practices, *Prentice Hall*.

Kohlborn T., Korthaus A., Chan T., Rosemann M., 2009. Identification and Analysis of Business and Software Services- A Consolidated Approach, In *IEEE Transactions on Services Computing*, Vol. 2, No. 1.

Liew, P., Kontogiannis, K. Tong, T., 2004. A Framework for Business Model Driven Development. In STEP'04, *12th International Workshop on Software Technology and Engineering Practice*.

(MDA), Model Driven Architecture, 2003, Guide version 1.0.1, Object Management Group (OMG)

Mellor, S., Clark, A., Futagami, T., 2003. Model Driven Development - guest editors' introduction, In *IEEE Computer Society*, September/October edition.

Mendling, J., Moser, M., Neumann, G., 2006. Transformation of yEPC Business Process Models to YAWL. In *SAC'06, 21st. Symposium on Applied computing*.

Mili, H., Frendi, M., et. al, 2006. Classifying Business Processes for Domain Engineering, In *ICTAI'06, 18th International Conference on Tools with Artificial Intelligence*.

Murzek, M., Kramler, G., Michlmayr, E., 2006. Structural Patterns for the Transformation of Business Process Models, In *EDOCW'06, 10th International Enterprise Distributed Object Computing Conference Workshops*.

Oquendo F., 2008. Formal Approach for the Development of Business Processes in terms of Service-Oriented Architectures using PI-ADL, In SOSE'08, 4th. *IEEE International Symposium on Service-Oriented System Engineering*.

Orriens, B., Yang, J., Papazoglou, M., 2006. A Rule Driven Approach for Developing Adaptive Service Oriented Business Collaboration, In *SCC'06, 3rd. International Conference on Services Computing*.

Papazoglou, M., van den Heuvel, W., 2006. Service-oriented design and development methodology. *International Journal Web Engineering and Technology*, Vol. 2, No. 4.

Papazoglou, M., Traverso, P., Dustdar, S., Leymann, F., 2007. SOC: State of the Art and Research Challenge, In *IEEE Computer Society*, November edition.

Quartel, D., Dijkman, R., van Sinderen, M., 2005. An approach to relate business and application services using ISDL, In *EDOC'05, 9th International Enterprise Computing Conference*.

(QVT) Query/Views/Transformations, 2008, v. 1.0, Object Management Group (OMG)

Rychly M., Weiss P., 2008. Modeling of Service Oriented Architecture: from business process to service realization, In *ENASE'08, 3rd International Conference Evaluation of Novel Approaches to SE*.

Roser, S., Bauer, B., Muller, J., 2006. Model- and Architecture-Driven Development in the Context of Cross-Enterprise Business Process Engineering, In *SCC'06, 3rd. Int. Conference on Services Computing*.

Sadiq, W., Sadiq, S., Schulz, K., 2006. Model Driven Distribution of Collaborative Business Processes, In *SCC'06, 3rd. Int. Conference on Services Computing*.

Smith, H., Fingar, P., 2003. Business Process Management: The third wave, Meghan-Kieffer Press

Stahl, T., Volter, M. et. al., 2006, Model-Driven Software Development, Technology, Engineering, Management, John Wiley & Sons, Ltd.

Tao Tao, A., Yang, J., 2006. Develop Service Oriented Finance Business Processes: A Case Study in Capital Market, In *SCC'06, 3rd. International Conference on Services Computing*.

Tao Tao, A., Yang, J., 2007. Supporting Differentiated Services With Configurable Business Processes, In *ICWS'07, 5th. International Conference on WS*.

Thomas O., Leyking K., 2008. Using Process Models for the Design of Service-Oriented Architectures: Methodology and E-Commerce Case Study, In *HICSS'08, 41st Annual Hawaii International Conference on System Sciences*.

Touzi J., Benaben F., Pingaud H., Lorré J. P., 2009. A model-driven approach for collaborative service-oriented architecture design, *International Journal of Production Economics*, Vol. 121, Is. 1.

(UML) Unified Modeling Language, Object Management Group (OMG), 2005.

Van der Aalst, W., ter Hofstede, A., Kiepuszewski, B., Barros, A. 2003. Workflow Patterns, In Distributed and Parallel Databases, 14(3), pages 5-51.

(WS), Web Service Architecture, World Wide Web Consortium(W3C)<http://www.w3.org/TR/ws-arch/>

(WfMC), Workflow Management Coalition, <http:// www.wfmc.org/>

(WS-BPEL) WS- Business Process Execution Language, Advancing Open Standards for the Information Society (OASIS) <http://www.oasis-pen. org/ committees/tc_home.php? wg_abbrev= wsbpel>

Zdun, U., Hentrich, C., Dustdar, S., 2007. Modeling Process-Driven and Service-Oriented Architectures Using Patterns and Pattern Primitives, *ACM Transactions on the Web*, Vol. 1, No. 3, Article 14.

Zhao, X., Liu, C., Yang, Y., 2006. Supporting Virtual Organisation Alliances with RelativeWorkflows, In APCCM'06, *3rd. Asia-Pacific Conference on Conceptual Modelling*.

APPENDIX

The selected primary studies in the systematic review are presented in Table 2.

Table 2: Primary studies selected.

Id	Title	Authors/Year	Paradigms integration
1	A Framework for Business Model Driven Development	Liew et al, 2004	BP+MDD
2	Generative Business Process Prototyping Framework,	Chen et al, 2006	BP+SOC+MDD
3	A model driven method for service composition modelling: a case study	de Castro et al, 2006	BP+SOC+MDD
4	A Rule Driven Approach for Developing Adaptive Service Oriented Business Collaboration	Orriens et al, 2006	BP+SOC+MDD
5	An approach to relate business and application services using ISDL	Quartel et al, 2005	BP+SOC+MDD
6	Classifying Business Processes for Domain Engineering	Mili et al, 2006	BP+MDD
7	Develop Service Oriented Finance Business Processes: A Case Study in Capital Market	Tao Tao et al, 2006	BP+SOC
8	Model- and Architecture-Driven Development in the Context of Cross-Enterprise Business Process Engineering	Roser et al, 2006	BP+SOC+MDD
9	Model Driven Distribution of Collaborative Business Processes	Sadiq et al, 2006	BP+MDD
10	Modeling Process-Driven and Service-Oriented Architectures Using Patterns and Pattern Primitives	Zdun et al, 2007	BP+SOC+MDD
11	Service-oriented design and development methodology	Papazoglou et al, 2006	BP+SOC
12	Structural Patterns for the Transformation of Business Process Models	Murzek et al, 2006	BP+MDD
13	Supporting Development and Evolution of Service-based Processes	Henkel et al, 2005	BP+MDD
14	Supporting Differentiated Services With Configurable Business Processes	Tao Tao et al, 2007	BP+SOC+MDD
15	Supporting Virtual Organisation Alliances with RelativeWorkflows	Zhao et al, 2006	BP+SOC
16	Transformation of yEPC Business Process Models to YAWL	Mendling et al, 2006	BP+MDD
17	ABBA: an architecture for deploying business-to-business electronic commerce applications	Baghdadi, Y. 2004	BP+SOC
18	Conceptual framework and architecture for service mediating workflow management	Hu et. al, 2003	BP+SOC
19	Pattern-based business-driven analysis and design of service architectures	Gacitua-Decar el al, 2008	BP+SOC+MDD
20	Inter-organizational systems: from business values over business processes to deployment	Huemer et. al, 2008	BP+SOC
21	Modeling of Service Oriented Architecture: from business process to service realization	Rychly et. al, 2008	BP+SOC+MDD
22	Business Process Modeling: a Service-Oriented Approach	Cauvet et. al, 2008	BP+SOC
23	Using Process Models for the Design of Service-Oriented Architectures: Methodology and E-Commerce CaseStudy	Thomas et. al, 2008	BP+SOC+MDD
24	A Seamless Modeling Approach for Service-Oriented Information Systems	Herold et. al, 2008	BP+SOC+MDD
25	A Model Driven Approach for the Alignment of Business and Information Systems Models	de Castro et al, 2008	BP+SOC+MDD
26	Towards Service Engineering: Service Orientation and Business-IT Alignment	Chen, H., 2008	BP+SOC
27	Identification and Analysis of Business and Software Services- A Consolidated Approach,	Kohlborn et. al,2009	BP+SOC
28	AMABULO- A Model Architecture for Business Logic	Bruckmann et. al, 2008	BP+MDD
29	Formal Approach for the Development of Business Process in terms of Service-Oriented Architectures using PI-ADL	Oquendo, F., 2008	BP+SOC+MDD
30	A model-driven approach for collaborative service-oriented architecture design	Touzi et. al, 2009	BP+SOC+MDD