2011 30th International Conference of the Chilean Computer Science Society

Process-Service Interactions using a SOA-BPM-based Methodology

Bazán Patricia
Linti Facultad de Informática UNLP
La Plata, Buenos Aires, Argentina
pbaz@info.unlp.edu.ar

Gabriela Perez
LIFIA - Facultad de Informática UNLP
La Plata, Buenos Aires, Argentina
gperez@lifia.info.unlp.edu.ar

Roxana Giandini
Lifia Facultad de Informática UNLP
La Plata, Buenos Aires, Argentina
giandini@lifia.info.unlp.edu.ar

Javier Diaz
Linti. Facultad de Informática UNLP
La Plata, Buenos Aires, Argentina
jdiaz@info.unlp.edu.ar

Abstract— **Reducing the gap between the definition, modeling and management of business definition, modeling and management of business processes and the realization of the main processes functions through software services requires a methodological approach. Addressing such gap, this paper focuses on the interactions between the activities of a business process and the functionality provided by software services. It presents a meta-model to formally specify such interactions. The proposed meta-model is instantiated through a case study. Lessons learnt through the development of an example were used to define a graphic editor prototype, as an automated tool for managing interactions between business processes and software services. The proposed approach enables the alternative application of top-down and bottom-up analysis techniques for modeling processes and services, and assists developers with an integrated tool for modeling business processes and software services in a seamlessly combined way.**

Keywords – Software Engineering, Business Process Management, Service Oriented Architecture, Business Process Management Notation, Meta-Modeling

I. INTRODUCTION

Business Process Management (BPM) is a strategy to manage and improve business performance by optimizing their processes through modeling, execution and performance measurement within a continuous improvement cycle [5]. BMP has gained considerable attention recently both for the communities of business administration and for computer science.

Service-Oriented Architecture (SOA) is a new approach to designing and building flexible and adaptable system that assists the development of systems in a dynamic business environment [4]. In SOA based systems, services can be shared and reused in various business processes. The result is a highly adaptable environment, with lower costs for the development of the applications, better integration and rapid deployment.

The integrated view of service orientation and process orientation leads to the definition of a methodological framework that sorts the concepts and establishes the business process lifecycle to give executable processes as a result. Previous works [6] [8] propose a general and comprehensive methodology for developing information systems based on the two approaches - SOA and BPM. The objective of such methodology is to provide a model for application integration, in order to align the business processes of an organization with software services that provide the functionality required by these processes. The methodology includes the following 8 stages:

1. *Organization and Strategic Planning* - outlining the profile of the solution and set clear goals and strategies to be used throughout the project.
2. *Process-Oriented Requirement Identification and Specification* - analyzing requirements from the point of view of the processes [6].
3. *Business Modeling* - identifying business processes and their main constraints. Business processes are described as a set of tasks in which the actors participate according to a given workflow.
4. *Process Modeling*- modeling each of the processes identified in the business use cases by using BPMN [2] [3], obtaining a business process diagram.
5. *Service Modeling* - identifying and defining the degree of granularity of the services. The aim is to ensure that the services designed comply with requirements of autonomy and atomicity, guaranteeing that each service has a clearly defined functionality and that it is uniformly and entirely accessible [4].
6. *Component Definition* - defining software components in terms of the services identified and their interaction mode (orchestration).
7. *Component Deployment* - deploying the results of previous stages on the chosen platform. It includes prototype development and integration with existing systems.
8. *Management and Monitoring* - assessing performance and seeking continuous improvements.

1522-4902/12 $26.00 © 2012 IEEE
DOI 10.1109/SCCC.2011.14

This paper proposes an extension to the described methodology. Specifically, it focuses on reviewing and improving the interaction between processes and services. The former ones obtained from a refinement of the business model and the latter identified and modeled to execute the process activities as new features or reusing existing services. To this end, this paper defines a service conceptualization meta-model based on business processes and their activities. The meta-model is underpinned by adapting a meta-model of SOAF (Service Oriented Architecture Framework) [9], combined with the BPMN meta-model [2]. The proposed meta-model is instantiated with a concrete example. In addition, a prototype of a graphical editor that provides the syntax of the meta-model is presented. The prototype enables to editing services from process activities.

The paper is organized as follows: Section 2 discusses the concept of the interaction between processes and services based on methodologies and frameworks available in literature. A comparison of existing related work and the proposal presented in this paper in included. Section 3 presents the methodological framework for modeling processes, services and their interaction. Section 4 details the proposed way of interaction between processes and services with a cyclical top-down, bottom-up approach. Section 5 presents the proposed meta-model definition for service conceptualization. Section 6 shows the application of the proposal through a case study. Section 7 describes the implementation of the software prototype. Finally, Section 8 draws some conclusions and outlines future work.

II. STATE OF THE ART IN PROCESS – SERVICES INTERACTION

Existing methods and frameworks underlying the work presented in this paper are classified into three main categories: services, processes and the interaction between them. For each category, related work is presented in the following three sections, respectively. Finally, a discussion comparing related work and this work is presented in the last section.

A. *Services*

SOMA (Service-oriented modeling and architecture) is a method for software development created by IBM. SOMA allows the user to design and build solutions based on SOA. SOMA defines techniques and provides detailed tasks to guide the analysis, design, implementation, testing and component development services [24].

SoaML (Service Oriented Architecture Modeling Language) [13] describes a UML profile and a meta-model for service design within a service-oriented architecture. Its aim is to support the activities of service modeling and design, within the model-driven development approach.

e3-value [29] is a methodology used to model and interconnect services. The e3-value method is supported by graphical modeling tools with certain reasoning ability and it is suitable to represent interorganizational processes and services. Its ontology allows to describe organizations that behave in accordance with certain procedures and regulations [27].

The main goal of the Eclipse STP project (SOA Tools Platform) [21] is to build frameworks to support the maintenance of the lifecycle of services in SOA. One of the STP active sub-projects provides an editor for BMPN [22] that has been used in this work.

B. *Processes*

IBM WebSphere Business Modeler [11] and Rational Modeler [12] are two examples of tools that assist in the transformation of business processes in a model that is closed to SOA and expressed in UML.

The process approach that was applied to build web applications is a perspective addressed in [30] [31] through WebML and its application to the construction of WebRatio BPM, an Eclipse-based tool that implements a methodology for web application development from a top-down design that is process-based and model driven.

C. *Integration between Processes and Services*

MINERVA (Model drIveN and sErvice oRiented framework for the continuous business processes improVement & relAted tools) is a tool that supports BPSOM (Business Process Service Oriented Methodology,) [25] defined as a plugin for Eclipse. The framework defines QVT transformations (Query / View / Transformation) between BPMN models and SoaML models to generate services automatically [26].

[8] features the development of a methodology to generate a model for integrating applications within an organization, so as to align the processes that define its operation with the services that support the functionality. This 8-stage methodology puts a strong emphasis on requirement identification with a process approach. There is also a business modeling stage, prior to the modeling of processes and services. Following, service identification and modeling, there are two more stages: component design and implementation. Finally, there are two global stages: Organization and Strategic Planning at the beginning and Management and Monitoring at the end, involving and crossing internal phases.

D. *Discussion*

SOMA is a very comprehensive method yet highly focused on service architecture. In our opinion, it does not consider in depth the task of requirement specification and the integration of process and software lifecycles.

SoaML focuses only on the modeling and design stages of software services, not considering process lifecycle. Therefore, when compared to our proposal, it is only a partial tool. Our proposal supports a concept similar to e3-value. However, our proposal emphasizes the internal processes of an organization, uses notational standards and adheres to a top-down modeling of business processes combined with a bottom-up analysis of the services.

The Eclipse STP Project does not propose the interaction between services and process activities.

The proposals in [11] and [12], in addition to being IBM proprietary, do not include the BPMN notation for process modeling, but transform them directly into executable components in the BPEL language.

WebRatio BPM addresses exclusively the domain of web applications that, while popular, are not the only type of software applications required by most organizations. Moreover, its top-down technique does not contemplate the possibility that preexisting features can be discovered as part of the activities of business processes.

MINERVA is a very complete solution and has a strong formal support to derive services but is immersed in a specific methodological framework whose characteristics and goals are different from those proposed in [8], which inspired this work.

In conclusion, most of existing related work emphasizes services. Our proposal also places value on business processes as elements that allow a cyclic interaction with services through interaction steps. These interaction steps, combine alternatively top-down and bottom-up approaches for modeling. Tables 1 and 2 synthesize the above discussion.

TABLE I. COMPARISON OF TOOLS AND METHODOLOGIES (PART 1)

Tool / Methodology	*It includes requirements*	*Modeling Objects*
SOMA	No	Only services
SoaML	No	MDD approach services
e3-value	No	Modeling and interconnection of services
Eclipse STP	No	Frameworks for maintenance of the life cycle of services in a SOA
Rational Modeler/ IBM WebSphere	No	Transformation tools business processes in a model close to SOA and expressed in UML
WebRatio BPM	No	Eclipse-based tool that implements a methodology for Web application development from a top-down design, process-based and model-driven
MINERVA	No	Services and processes within its own methodology
This proposal	Yes	It covers both the life cycle of business processes as software in a unified way.

TABLE II. COMPARISON OF TOOLS AND METHODOLOGIES (PART 2)

Tool / Methodology	*It is a methodology*	*Adherence to standards*
SOMA	Yes	No. It is an IBM methodology.
SoaML	Yes	Yes. It defines a UML profile.
e3-value	Yes at large-scale	No
Eclipse STP	No	Yes
Rational/ Transformation SOA	No	No. It is IBM proprietary.
WebRatio BPM	No	Yes, as an extension of BPMN.
MINERVA	Yes	Yes
Our proposal	Yes.	Yes

The next section describes the interaction steps of the proposed methodology [8].

III. METHODOLOGICAL FRAMEWORK

As envisioned in [8], the business modeling stage produces a process map which, together with the identified and specified requirements and business use cases, forms the basis for modeling business processes.

From the business model, modeling business processes is a refinement step, where the processes are enriched with defining functional aspects of system use cases and description of information objects. This refinement step allows dividing the process into tasks and defines roles and textual documentation that contribute to BPD. Obtaining the process model gives the proper ground for identifying the elements (services) which will undertake the business processes: the latter becoming consumers of these services.

Figure 1 on the left shows the eight steps of the proposed methodology [8] showing the overlaps between the methodology phases and the phases of process and service lifecycles. On the right, the figure shows the way in which the methodology phases interact, both from the point of view of process life cycle (depicted as rows) and software life cycle (depicted as columns).

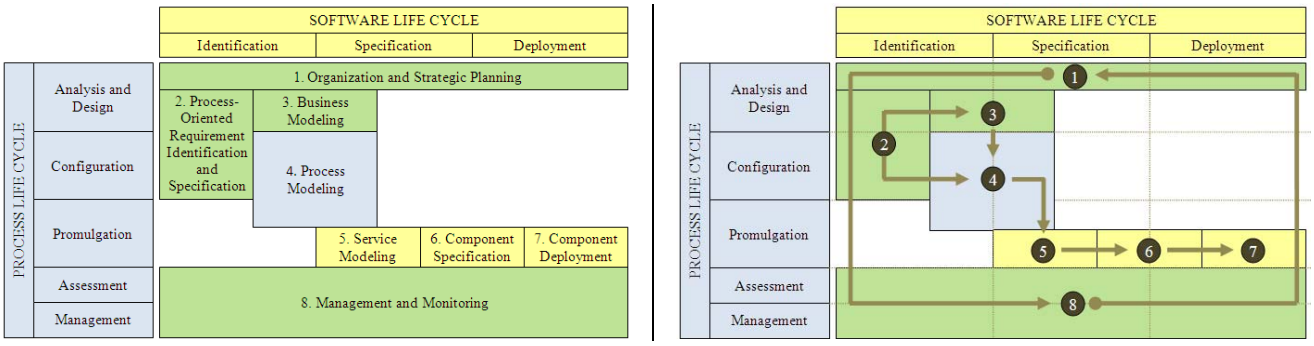

FIGURE 1. METHODOLOGY PHASES OVERLAPPED WITH PROCESS AND SERVICE LIFE CYCLES AND INTERACTIONS BETWEEN PHASES

A. *Process Modeling and Classification*

Business process modeling focuses on the identification and modeling of processes. Applying different levels of abstraction, there are three related concepts: 1) *processes*, 2) *sub-processes* and 3) *tasks*. A *process* is a network of "things to do" or activities. A *sub-process* is a process in itself, whose functionality is part of a larger process. A *task* is the lowest level of a process, a unit that cannot be decomposed.

BPM meta-model presented in Figure 3 describes process activities as generalizations of sub-processes and tasks.

B. *Service Modeling and Classification*

Services can be seen, within the object oriented paradigm, as objects that provide more information and less coupling due services are generally not associated; but whose behavior is triggered by its meta-information, basically in terms of service contracts and execution contexts. Nevertheless, their similarity to objects makes it possible to model them in terms of interfaces.

Inspired by UML classes, Erl defines the symbol of services as a circle comprised of two areas: one for the service name and another for its capabilities [10].

Service contracts indicate the interfaces they provide, their operations and parameters, favoring the cataloging of their services and the resulting composition to extend the functionality.

The granularity of a service communicates the level of detail associated with their functional scope.

There is no single criterion that indicates the level of granularity that is appropriate for a service, but the design principles and service classifications used have a direct impact on the level of granularity.

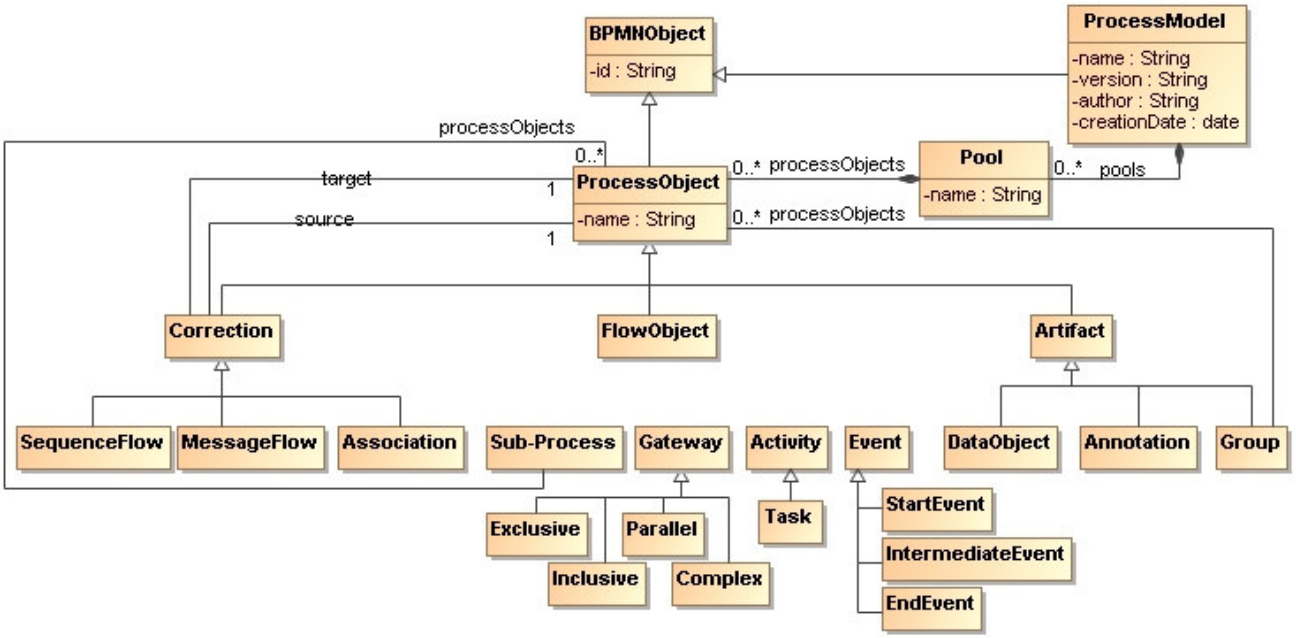

FIGURE 2. SIMPLIFIED BPMN META-MODEL

The classification proposed in our meta-model takes into account *business*, *infrastructure* and *entity* services.

Business services have the functional scope of business entities; they are highly reusable and agnostic to many business processes; for example, a service solving the entire logic of a purchase order, including the restrictions imposed by the domain where it is issued.

Entity services have functional limitations associated with classes within a class model. They have less reuse potential and are composed of other services; for example, a service providing operations on a class, enabling the creation, update, and access to the class attributes. *Infrastructure services* are cross-cutting services that solve aspects of technology or those specific to the software application; for example a service managing exception reports or a service establishing a generic connection to a relational database.

Finally, a service provides a collection of capabilities that are grouped based on the functional context for which the service was defined. The ability of a service represents the set of specific functions that the service provides. Such functions are specified in the service contract and constitute the mechanisms through which the service is invoked.

IV. INTERACTION STEPS

As mentioned in Section 3, once the processes are modeled as a set of ordered activities, services are modeled to provide the functionality required by such processes. The modeling of services begins with their identification.

The identification of services can be addressed with a bottom-up approach. The idea behind this approach is to consider the service as a well-defined scope of functionality and make it accessible uniformly and entirely, regardless where it is used [3]. The bottom-up approach allows us to consider individual functional components and then their composition in a global structure.

Meanwhile, business processes were modeled at earlier stages following a top-down approach. This approach comes from more general elements (processes) and decomposes them into components with a higher degree of refinement (sub-processes and tasks). This approach allows a global and comprehensive perspective.

For these reasons, we conclude that process modeling adheres to a top-down approach and service modeling has bottom-up characteristics. Thus, the interaction steps for the modeling of processes and services show the need for a mechanism to help maintain this iterative cycle of top-down / bottom-up refinements. To provide this mechanism, this paper proposes the definition of a meta-model to conceptualize services. This service meta-model will facilitate the interaction between services and processes modeled in BPMN. This premise is based on the existence of a process meta-model provided by BPMN.

V. SERVICE CONCEPTUALIZATION META-MODEL

The proposed service meta-model is based on adapting and integrating the SOAF service conceptualization meta-model [9] (Figure 3) and the BPMN meta-model [23] (Figure 2). The underlying idea is considering each process as a set of activities, where each activity is seen as an abstraction of certain functionality that will be performed by a service.

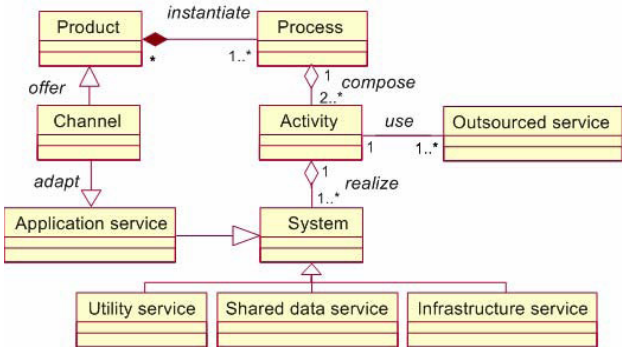

FIGURE 3. SOAF SERVICES CONCEPTUALIZATION META-MODEL

The proposed meta-model is called P2S - ProcessToService and is presented in Figure 5. It is built as an instance of the MOF standard language (Meta Object Facility) [18], which constitutes the most abstract layer in the 4-layer OMG modeling architecture [17] [20].

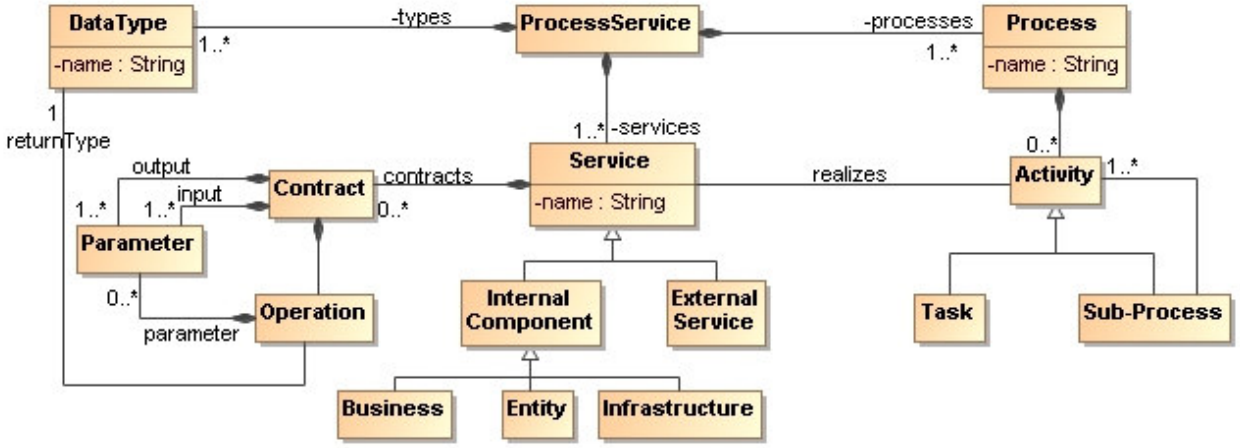

FIGURE 4. P2S SERVICE CONCEPTUALIZATION META-MODEL

The P2S meta-model specifies that a *Process* is realized by an aggregation of *ProcessService*(s). Each *ProcessService* involves various *DataType* and is performed by a set of *Service(s)*. A *Service* is conceptualized as a generalization of *Internal Component* – new component defined by the development process, or *External Service* – existing service. Each *Internal Component* defined is classified according to the generalization of meta-classes in the meta-model – *Business*, *Entity* or *Infrastructure.*

In addition, the meta-model specifies that each *Service* "realizes" an *Activity* – that could be a *Sub-Process* or a *Task*. Thus, the top-down and bottom-up combination described in Section 4 is applied. That is, after applying a top-down analysis for identifying process activities, a bottom-up analysis is applied to obtain the functional components that will realize such business activities. The premise for determining a functional component is that the component should contain: 1) the information to be

managed, 2) the operations required to manipulate such information, and 3) the business rules that govern such manipulation.

As a result of executing a bottom-up analysis for specifying the required services, each process activity will be associated with the service that will realize it.

Finally, each *Service* has a *name* and offers a collection of *Contract*(s). In turn, each *Contract* has a set of *Operation*(s) that implement the functionality provided by the service. Each *Operation* can have zero to many *Parameter*(s) and produces a result of certain *DataType*.

VI. APPLYING THE META-MODEL – CASE STUDY

This section presents the application of P2S to a case study. The case models a process that takes place in a repair center for a transport company. The process refers to a request for vehicle repair issued by drivers to the center.

The application of the proposed model is as follows. First, the process is specified in BPMN. Second, activities are identified from the BPMN process model. Third, from the activities, a set of services is identified for realizing them.

According to the proposed methodology, the services are identified by chorded circles. As explained in Section 3, the relation between activities and services is based on the relationship "realize". In the following sub-sections, the development of the case study is presented in detail.

A. Description of the Process and BPMN Modeling

The process description is as follows. When a driver comes with a vehicle to repair, a work order (WO) is generated (*Issue work order*). For each required repair, an item is included in the WO, the required materials are identified and checked if there is available stock. (*repairs*). If all materials are available for all the items in the WO, a mechanic and a pit are assigned (*Assigned mechanic and pit*). Following, each planned repair is done (*Do repair*). When a repair is finished, the actual cost is calculated considering the used materials and the working hours (*Register repair cost*). When all repairs are finished, the WO is closed (*Close WO*) and the driver is notified (*Notify driver*). Finally, the driver receives the repaired vehicle (*Receive vehicle*). The BPMN process model is presented in Figure 5 which shows the main activities of the process.

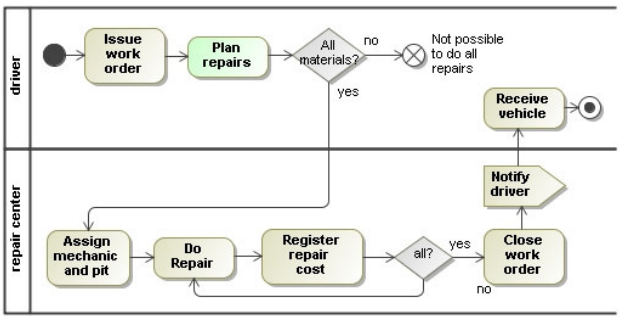

FIGURE 5. WORK ORDER PROCESS

Two roles involved in the process are identified: the driver and the repair center. The *Plan repairs* activity is modeled as a sub-process in order to better detail the tasks

The *Plan repairs* sub-process comprises: 1) for each repair, it creates an item associated with the WO which details the required inputs or materials; 2) checks the inventory stock of these inputs, 3) if stock is available for all inputs, it calculates the cost. The remaining activities are modeled as simple tasks.

B. Service Models Represented by Chorded Circles

According to the SOA-BPM integrated approach, once the business process is modeled, services are to be modeled using the chorded circle notation and following the premise of services as the building blocks of business processes.

In Figure 6, services are identified by grouping behavior following the criteria specified in Section 5. This conceptual construction follows a bottom-up approach because this grouping behavior arises from the classes identified through an object-oriented analysis. Thus, the *WorkOrder Service* manages work orders and their items in a unified way. A similar situation occurs with *Driver Service.* Both are entity services. In addition, *PlanningService* will be responsible for planning the repairs – a business service.

Following an object-oriented analysis, entity services, like *WorkOrder* and *Driver,* arise from classes modeling business entities. This approach contributes to isolate functional aspects that remain linked to the activities of the business process.

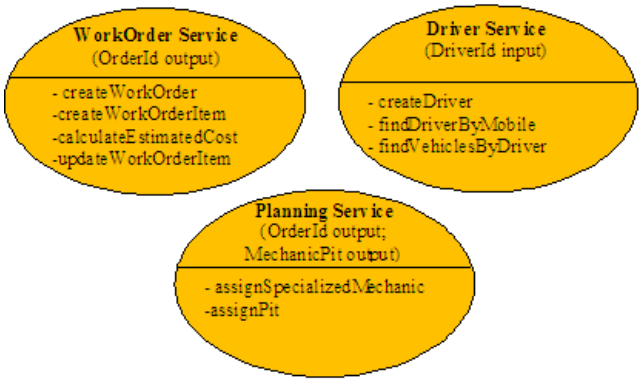

FIGURE 6. WORK ORDER PROCESS SERVICE MODEL

VII. PROPOSAL IMPLEMENTATION PROTOTYPE

In order to provide tools for the implementation of the P2S meta-model, we implemented a service graphical editor prototype. To do this, we used Eclipse plugins that are included in the Eclipse Modeling project [14, 15, 19].

Following the integrated SOA-BPM methodology [8], once the business model is finished, the next activity is to model services. To model the business process, we used an Eclipse project open source graphical editor called BPMN2 [28]. The editor splits the information into two files: one

containing the instances of the process elements (with a BPMN extension), and the other having graphic information.

From the files created by BPMN2, Figure 7 shows how the essential information for modeling services is exported from the bpmn file to P2S. The information is exported in two files, following similar criteria as already explained for BPMN. Later, the prototype uses such files to graphically edit the services.

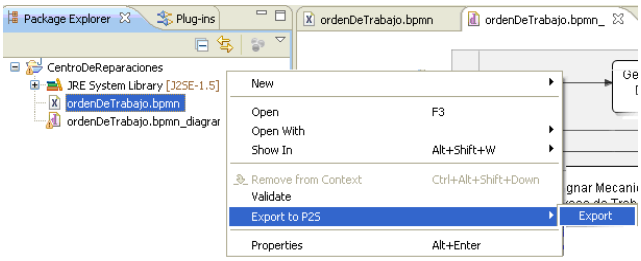

FIGURE 7. EXPORTING WO BUSINESS PROCESS DESCRIPTION TO P2S

Figure 8 shows the example as an instance of the meta-model proposed in Figure 4. This instance contains some of the activities identified in the process model in Figure 5 and the services modeled in Figure 6. The same figure shows the service graphical editor with some instances of the repair center example (in the Spanish version of the case study). It also highlights the instance of the Entity class Service; called OrdenDeTrabajo (WorkOrder). As prescribed by the methodology, the prototype also enables to associate services and process activities through the "realize" property.

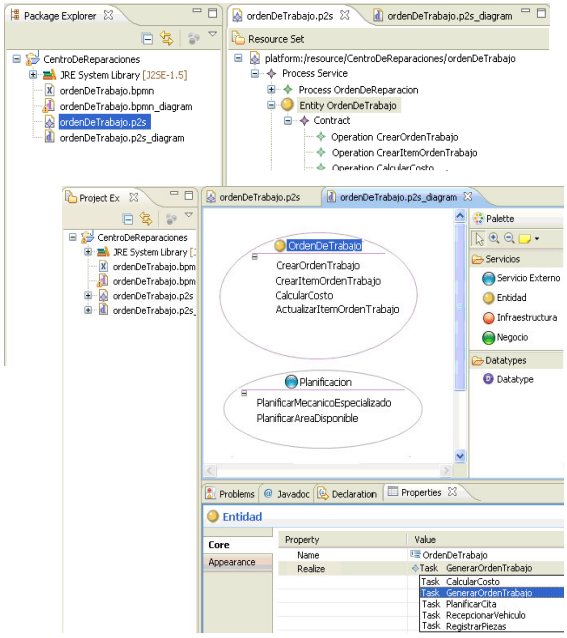

FIGURE 8. PROTOTYPE SERVICE GRAPHICAL EDITOR

VIII. CONCLUSIONS

The integrated approach between BPM and SOA [8] defines the different development stages, outputs of each stage, and interactions between stages. In particular, when defining the methodology, the interaction between process modeling and service modeling was not treated in depth.

This paper provides an approach for the interaction steps between process and service modeling, by using the concept of meta-model written in the MOF language - a graphic OMG meta-language intended to specify notations that support ambiguity-free interoperable tools.

While the SOAF methodology provides a framework to insert the notion of SOA in traditional technological models, our proposal is to take a simplified SOAF meta-model and apply it as top-down/bottom-up interaction steps between processes and services.

From the processes and activities described with process models, and the services identified and modeled in the service model, the proposed meta-model relates such two concepts and provides a notation for specifying the interaction between process activities and services, while defining a service taxonomy. The interaction step enables to apply a top-down process analysis and a bottom-up service analysis alternatively. Such approach facilitates the discovery of existing functionalities that are capable of performing the process activities.

Additionally, we have implemented a software prototype (supported by the proposed meta-model) that will assist the developer in defining the interaction between processes and services in business process modeling, following the standard BPMN language. This work thus constitutes an extension to the aforementioned BPM-SOA methodology, which provides an informal approach to define this combination of process modeling and service modeling.

The graphical editor of the prototype allows instantiating services in combination with BPMN graphical elements. Meanwhile, as mentioned in Section 2, SOA and the BPMN language already have open source editors within the Eclipse project [21, 22]. We have used these editors to export process models to facilitate the modeling of interactions between processes and services.

Future work will follow two research lines. On the one hand, we will focus on the evolution of these editors to improve their ability to express and incorporate features, such as the generation of WSDL (Web Service Description Language) [32] for the publication of services and even their orchestration within a BPEL code engine [32]. On the other hand, we plan to continue the refinement of the BPM-SOA methodology.

REFERENCES

[1] Open SOA, http://www.osoa.org/display/Main/Service+Component+Architecture+Home

[2] Business Process Modeling Notation (BPMN) Version 1.2 OMG, http://www.omg.org/spec/BPMN/1.2

[3] Imaz M.: Curso Introducción a SOA/BPM. Organizado por SADIO (2008).

[4] IBM Global Technology Services: How service-oriented architecture (SOA) impacts your IT infrastructure (2008)

[5] Oracle Whitepaper: SOA Governance: Framework and Best Practices (2007)

[6] Bazán P., Giandini R, Diaz J. "Modelos de procesos y modelos de servicios: una visión unificada". Jornadas Chilenas de Computación 2009. XXI Encuentro Chileno de Computación (ECC) (2009)

[7] Weske Mathias, "Business Process Management: Concepts, Languages, Architectures". Springer, Pag 3-67. ISBN 978-3-540-73521-2. 2008

[8] Bazán P. "Un modelo de integrabilidad con SOA y BPM". Tesis de Maestría en Redes de Datos. Facultad de Informática. Universidad Nacional de La Plata. Abril 2010.

[9] Erradi, A., Anand, S., Kulkarni, N. N.: SOAF: An Architectural Framework for Service Definition and Realization, IEEE Int. Conf. on Services Computing, 2006, pp. 151-158.

[10] Erl, Thomas. "SOA Principles of Service Design". Prentice Hall. ISBN-13: 9780132344821. 2007. Pag.25-119

[11] Rational Software Modeler http://publib.boulder.ibm.com/infocenter/rsmhelp/v7r5m0/index.jsp?topic=/com.ibm.xtools.transform.cfm.wbm.doc/topics/ccwbmtransf.html. (al 16/10/2009)

[12] Transformation to SOA: Part 1. From business process to service model architecture using IBM WebSphere Business Modeler and IBM Rational Software Architect http://www.ibm.com/developerworks/rational/library/07/1225_bennet-balaba/. 2007 (al 16/10/2009)

[13] Service oriented architecture Modeling Language (SoaML) - Specification for the UML Profile and Metamodel for Services (UPMS). OMG Document. 2008

[14] Eclipse Modeling Framework EMF. http://www.eclipse.org/modeling/emf/

[15] The Eclipse Project. Home Page. Copyright IBM Corp, 2000.

[16] Tutorial BPMN. http://wiki.eclipse.org/GMF_Tutorial_BPMN

[17] Object Management Group (OMG), http://www.omg.org

[18] OMG/MOF Meta Object Facility (MOF) 2.0. OMG Adopted Specification. October 2003. http://www.omg.org

[19] Eclipse EuGENia. www.eclipse.org/gmt/epsilon/doc/eugenia/

[20] C. Pons, R. Giandini, G. Pérez. "Desarrollo de Software Dirigido por Modelos. Conceptos teóricos y su aplicación práctica". EDULP & McGraw-Hill Educación.(2010). ISBN 978-950-34-0630-4.Cap 4

[21] SOA Tools Platform Project http://www.eclipse.org/stp/

[22] SOA Tools BMPN Modeler http://www.eclipse.org/bpmn/

[23] Roxana Giandini, Gabriela Pérez, Claudia Pons. Un lenguaje de Transformación específico para Modelos de Proceso del Negocio. XXXVI Conferencia Latinoamericana de Informática (CLEI 2010). 18 al 22 de Octubre de 2010, Asunción, Paraguay.

[24] A. Arsanjani, S. Ghosh, A. Allam, T. Abdollah, S. Ganapathy, K. Holley. "SOMA: A method for developing service-oriented solutions". IBM SYSTEMS JOURNAL, VOL 47, N 3. 2008

[25] Delgado, A., Ruiz, F., García - Rodríguez de Guzmán, I., Piattini, M.: Towards a Service- Oriented and Model-Driven framework with business processes as first-class citizens, In: 2nd Int. Conf. on Business Process and Services Computing (BPSC'09), Leipzig, (2009)

[26] Delgado, A., García - Rodríguez de Guzmán, I., Ruiz, F., Piattini, M.: Tool support for Service Oriented development from Business Processes, 2nd International Workshop on Model-Driven Service Engineering (MOSE'10), Málaga, (2010)

[27] Joris Hulstijn and Jaap Gordijn. Risk analysis for inter-organizational controls. In Joaquim Filipe and Jose Cordeiro editors, Proceedings of the 12th International Conference on Enterprise Information Systems (ICEIS 2010), Vol. 3:314-320, 2010

[28] http://www.eclipse.org/projects/project_summary.php?projectid=modeling.mdt.bpmn2

[29] http://www.e3value.com/

[30] Marco Brambilla, Stefano Butti, Piero Fraternali: WebRatio BPM: A Tool for Designing and Deploying Business Processes on the Web. ICWE 2010: 415-429.

[31] Marco Brambilla, Stefano Ceri, Piero Fraternali, Ioana Manolescu: Process modeling in Web applications. ACM Trans. Softw. Eng. Methodol. 15(4): 360-409 (2006)

[32] BPEL Tutorial http://go.techtarget.com/r/8638480/2744320. 2009 (al 16/10/2009)

[33] WSDL 1.1 http://www.w3.org/TR/wsdl.