International Journal of Cooperative Information Systems
Vol. 18, No. 2 (2009) 225–260

TOWARDS A SERVICE-ORIENTED MDA-BASED APPROACH TO THE ALIGNMENT OF BUSINESS PROCESSES WITH IT SYSTEMS: FROM THE BUSINESS MODEL TO A WEB SERVICE COMPOSITION MODEL

VALERIA DE CASTRO* and ESPERANZA MARCOS†

Kybele Research Group, Rey Juan Carlos University
Tulipán S/N, P.C. 28933, Móstoles, Madrid, Spain
**valeria.decastro@urjc.es*
†esperanza.marcos@urjc.es

ROEL WIERINGA

Centre for Telematics and Information Technology
University of Twente, P. O. Box 217
7500 AE Enschede, The Netherlands
roelw@cs.utwente.nl

In recent years, the automation of business processes has become one of the most prominent and promising uses of Web service technology. Consequently several languages have been created for the execution of business processes, making it possible to define new and more complex services or business processes which are implemented for example by means of Web service composition. Nevertheless, these kinds of languages are not suitable for use in the early stages of the development process of information systems. Special methodologies or techniques are therefore necessary to allow systems analysts to understand services from a business point of view, while facilitating the design and development of Web service composition. In this paper, we present a service-oriented approach to information system development that starts by identifying, through business modeling, the services required by the customers of a business, to make it possible to create a Web service composition model. This model will facilitate the transformation to specific languages for business process execution, thereby reducing the development efforts made in service-oriented applications. The method proposed is illustrated by means of a Web application for the management of medical images, which we have taken as a case study.

Keywords: Service-oriented computing; service-oriented engineering; web services composition model; business model; model driven development.

1. Introduction

New technological proposals for the Web such as XML, Web Services, business process automation, B2B, etc., have led to the emergence of a new paradigm for the development of applications known as *Service-Oriented Computing* (SOC),[33] which uses services as the basic constructs to support rapid, low-cost and easy composition of distributed applications. Service-oriented computing is currently one of the major research topics in the field of software development and has brought about an evolution in Information Systems (ISs) themselves, as well as in how they can be developed. Fully integrated companies are now being replaced by business networks in which participants provide each other with specialized services.[11] Many ISs, mainly those for the Web, are therefore being created as a means of offering services over the Internet, usually involving simple or complex business processes in which several participants are collaborating.

As a consequence of this change of paradigm and, as occurred with object-oriented computing in its day and with the later emergence of object-oriented methodologies, in the field of software engineering a new "service-oriented engineering area" is arising. The roadmap for research in service-oriented computing outlined by Papazoglou, Traverso, Dustdar and Leymann[34] stresses the importance of defining methodologies for service-oriented analysis and design. They claim methodologies or development techniques of this type are crucial elements for developing meaningful services and business process specifications as well as a basic requirement for service-oriented applications that leverage Web services.

According to the roadmap,[34] one of the main challenges facing service-oriented computing is the provision of methodologies that support the specification and design of service composition, allowing software engineers to go from the earlier stages of business analysis to the final step of implementation. While the design and development of simpler services is a relatively easy task, the development of business process comprising several independent services is not. The main reason for this is that the transformation from high-level business modeling, generally carried out by business analysts, to an executable business process language, such as BPEL4WS,[3] is a far from trivial issue.[39] MDA (*Model Driven Architecture*)[31] is an important tool for the alignment of high-level business processes and information technologies.[20] It provides a conceptual structure that brings together the diagrams used by business managers and analysts and those used by software developers, and is capable of organizing them in such a way that the requirements specified in one diagram can be traced through other more detailed ones derived from them.

The proposal we present in this work is a *service-oriented approach to IS development*, called SOD-M (Service-Oriented Development Method), which is integrated in an MDA framework.[26,27] SOD-M defines a model-driven process that starts from high-level business environment modeling and enables us to obtain a Web service composition design. We should point out that following a *service-oriented approach* means that ISs will be built using *services* as first-class objects for the whole development process, but it does not imply that the systems will necessarily be implemented by Web services. In fact, this is just one of several options (perhaps one of the most recommendable), but others could be chosen, such as components, objects, etc.

One relevant characteristic of the method is the use of *services* identified in high-level business modeling as starting elements for software construction. The concept of service exists in both management science and computer science. A unifying concept of service is that it is a client-provider interaction that creates value.[21] The value is created because this interaction satisfies a specific need of the client.

SOD-M uses the e^{3}value method[19] as a business modeling approach, which allows us to understand the business environment in which the IS will be used and to identify the services that will be offered by the systems to satisfy customers' needs (from now one business services). In e^{3}value, a business model is called a value model. Moreover, SOD-M defines models for the behavioral modeling of the IS, including new modeling elements to make it possible to represent the business services and how they will be provided through the Web service composition.

To validate SOD-M we tested its applicability with a set of different cases studies of two types: (a) *laboratory case studies*, specifically designed to investigate needs in the service-oriented development of ISs; and (b) *real case studies*, i.e. the development of actual ISs in real situations. In this paper, we illustrate the method with one of the real case studies developed, a Web Information System (WIS) for medical image management and processing, called GesIMED. It was developed in collaboration with the GTEBIM research group at the Rey Juan Carlos University[16,17] and is now being used successfully by neuroscientists and clinicians conducting research and clinical studies in several medical and research centres in Madrid, Spain.

In previous proposals we presented a first version of some of the behavioral models included in SOD-M.[13] In this paper we complete our previous work by presenting the high-level business models and the guidelines for obtaining behavioral models based on them. We also redefine the process and models previously defined, presenting a new service-oriented approach to IS development. For reasons of space, we do not offer a very formal presentation of the method and all the meta-models, but we do give an overview of the main concepts of SOD-M as well as the process and models. So, through a case study, we explain how Web service composition design may be derived from high-level business modeling.

The rest of this paper is structured as follows. In Sec. 2, we present some related work describing the main benefits and contributions of our proposal. Section 3 briefly describes the method and the process proposed by SOD-M. Section 4 illustrates SOD-M (the models and mappings) with a case study, GesIMED: a WIS for medical image management and processing. Finally, in Sec. 5, we conclude by summarizing the main contributions and outlining future work.

2. Related Work

Analysis of current literature related to service-oriented development shows that most authors treat the topic from a technological point of view.[6,11,33] However, despite the impact of the service-oriented computing paradigm on software development and on the way in which systems can be constructed, there are currently no methodologies to facilitate software development based on the new paradigm.

Nevertheless, two types of proposal related to this work may be identified: on the one hand, *those originating in the field of Web engineering*, i.e. web methodologies that have been adapted to give support to new technologies for the Web, such as Web services and business process; and on the other hand, *those defining modeling* *methods or techniques for the development of certain aspects of service-oriented systems,* e.g. modeling techniques for Web Services or business processes composing Web Services.

2.1. *Methodologies for WIS development*

Many methodological proposals have arisen for the building of Web systems or applications in a systematic and rigorous way, giving rise to the new profession of Web Engineering.[18] The first methodological proposals in this field appeared, in general, as adaptations of classical methodologies for specializing in a specific type of development, such as studies from the fields of hypermedia and multimedia, and databases. However, the new technological proposals for the Web (i.e. XML, Web services, business process automation, Web semantics, etc.) have brought about changes in the techniques and methodologies used for WIS development. Thus, over the last few years, new extensions have appeared, proposals existing, for example, for representing business processes in WIS, such as the extensions to WebML,[8,10] to UWE and OO-H,[22] to OOHDM[37] and also to ADM,[15] among others; and also for the representing WISs that interact with Web services.[9,23]

Table 1 shows a comparison of the main WIS development methodologies, along with a column for SOD-M, to show its main differences with regard to the current state of the art. Analysing these proposals implies studying the following characteristics: (1) the *Development Paradigm*, which can be object-oriented or service-oriented, defined in this paper as one in which *services* are considered the basic elements for software development; (2) *Models proposed*, identifying those proposed for specifying the business environment and behavior of the system; (3) *Modeling Notation*, determining whether the proposals use UML as a modeling language and, if so, whether they define UML profiles for a specific development domain; and, (4) *MDA-Based Approach*, indicating whether the proposals are defined on the basis of MDA characteristics, i.e. whether they define models for the different levels of abstraction proposed for MDA (CIMs — Computation Independent Models, PIMs — Platform Independent Models, and PSMs — Platform Specific Models), and also whether mapping between them is specified.

One of the most important differences between SOD-M and existing approaches to WIS development is the development paradigm. As shown in Table 1, the most widely used approach to WIS construction is object-oriented (OOHDM, UWE, WebML, OOWS, Autoweb and W2000). Its origin was in the beginnings of the Web, when applications were basically generated as a means of publishing information. It is nevertheless evident that Web users nowadays expect not only information from the WIS of an organization, but also Internet access to the different services available. As a result, such methodologies as UWE, WebML and OO-H, among others, have extended their development processes in such a way as to contemplate the representation of business processes in the development of hypertext. SOD-M also defines guidelines for incorporating business processes in IS construction but, unlike

Table 1. Summary of the main proposals for the Development of WIS.

	Proposals for the Development of WIS								
	OOHDM *Object* Oriented	UWE *Object* Oriented	WEBML *Object* Oriented	OOWS *Object* Oriented	OO-H *Object* Oriented	WSDM *Object* Oriented	Autoweb *Object* Oriented	W2000 *Object* Oriented	**SOD-M** *Service* Oriented
1. Development paradigm									
Business Model	×	×	×	×	×	×	×	×	✓
Use Case Model	✓	✓	×	✓	✓	×	×	✓	✓
Process Model	✓	✓	✓	✓	✓	✓	×	✓	✓
2. Proposed models Services									
Services Model	×	×	✓ (in the Hypertext)	×	×	×	×	×	✓
Services Composition Model	×	×	✓ (in the Hypertext)	×	×	×	×	×	✓
3. Model notation									
Uses UML	✓ (Partially)	✓	×	✓ (Partially)	✓ (Partially)	✓ (Partially)	×	✓	✓
Defines UML Profile	×	✓	✓ (Partially)	×	×	×	×	×	✓
4. MDA approach									
Abstraction levels	PIM, PSM	PIM, PSM	PIM	PIM	PIM	PIM	PIM	PIM	CIM, PIM, PSM
Mappings between models	✓ (Partially)	✓	×	✓ (Partially)	✓ (Partially)	×	✓	×	✓

current methodologies, it ***defines a new service-oriented approach for the development of ISs***. Thus, SOD-M sets out to develop applications based exclusively on business services, which is what end-users need. This approach facilitates the development of service-oriented applications as well as their implementation using current technologies, such as Web services.

Regarding the models proposed by each of the methodologies, one major contribution of our method should be underlined: the ***definition of a high-level computation independent business model***. In general, all the methodologies analyzed define activities for the capture of requirements, using use case models (OOHDM, UWE, OO-H, W2000), or task diagrams (OOWS, WSDM) representing the interactions between the user and the system. However, it must be emphasized that, from the MDA point of view, the CIMs are not used to represent the details of the system itself, but rather those of the application domain, so as to serve as a bridge between the business experts and the developers of the system. The business model proposed by SOD-M possesses this characteristic and unlike the ones mentioned above, incorporates a value-oriented view, which ensures that the services are related to the business model of the organization using them.

The usage of UML may be said to be common to most of the WIS development methodologies analyzed, although many, like OOHDM, OOWS, OO-H and WSDM, integrate it with their own notations. The WebML methodology also defines its own notation for most of its models. SOD-M ***proposes the use of UML as the only notation for modeling ISs*** (at PIM and PSM level), and defines, furthermore, ***a UML profile to permit the service-oriented development of ISs***. For the CIM level, which is oriented to the business analyst, SOD-M proposes using a value model[19] adopting its own notation and a UML activity diagram in order to model business processes.

Many of the developers of the main WIS development methodologies have worked on the MDA line, with models being defined for the different levels of abstraction of this architecture while some, such as OOHDM, UWE, OOWS, OO-H and Autoweb, specify rules for transformation between them. As explained earlier, SOD-M is part of a methodological framework for the development of ISs based on MDA and therefore ***defines models at the CIM, PIM and PSM levels of the architecture proposed by MDA*, and *proposes mappings between them***.

2.2. *Proposals for service-oriented development*

The service-oriented computing paradigm appeared in response to a fundamental change in the way that companies manage their businesses, based as it is today on business networks with participants providing specialized services to each other.[11] An analysis of the literature related to service-oriented development reveals that currently there are no methodological proposals for defining a service-oriented approach to embracing the whole software development process, although several studies do exist concerning such topics as modeling techniques or methods and other aspects of its development.

Several proposals have emerged, for instance, which can be classified according to their modeling aspect: (a) *proposals for business modeling*, that is, those focussing on the modeling of the business actors, their relationships and the services they offer to others, rather than on their own processes[19,38]; (b) *proposals for the development of business processes*[7,24]; (c) *proposals for the development of Web services*[5,35] and, (d) *proposals for the Web services and their composition modeling.*[42]

Table 2 offers a summary of the main techniques and methods studied stressing the following characteristics: (1) the *modeling aspect concerned* (a, b, c, and d), (2) *correspondence with the MDA architecture level*, and (3) the *notation used.* This table also includes a column for SOD-M, to assist comparison with the other proposals analyzed.

As said above, the analysis of the proposals related to the development of service-oriented systems produces none for their holistic development, but instead a variety of proposals for modeling individual aspects of them. Therefore, as Table 2 shows, there is a fundamental difference between SOD-M and the other proposals considered here, since ***SOD-M is a complete IS development method and, therefore, as such envisages the modeling of all of the elements related to the development of service-oriented systems***. Regarding business modeling, SOD-M proposes using the e^3 value method, for it defines guidelines for identifying and representing the main concepts of a business. At the same time, it facilitates the identification of the business services to be implemented in the IS in order to satisfy the end-user's needs. MDWSD[5] and Service Components[42] are alone in proposing methodological approaches for the development of Web services and, in the case of Service Components, also for their composition. However, insofar as they differ from SOD-M, these proposals are just oriented to low-level design and the implementation of Web services, and do not embrace activities for the identification and modeling of high-level services.

On the relationship between the techniques or methods analyzed and MDA abstraction levels, it should be stressed that, since these proposals do not envisage the holistic development of systems, they correspond, in general, to single levels of MDA. It is thus obvious that the modeling of Web services and their composition occur at a lower level of abstraction, such as the PSM level, while the business modeling and business processes are performed at a higher abstraction level, such as the CIM or PIM levels. Nevertheless, in the case of SOD-M, which is a complete development method based on MDA, ***models corresponding to all of the levels of the MDA are proposed***, thus embracing both high-level business modeling and the modeling of the system at the platform-independent and platform-dependent levels.

Especial mention must be made of BPMN,[7] which has recently become a *de-facto* standard for business process modeling. In SOD-M we do not propose using BPMN as a notation for high-level business process modeling, as we consider that a UML activity diagram[32] may be sufficient to model a business process. The aim of high-level business modeling in the framework of MDA is to allow business analysts

Table 2. Summary of proposals related to service-oriented development.

	Proposals for Service-Oriented Development							
	e^3-value	WIED	BPMN	UML 4BPEL	UML 4WSDL	MDWSD	Service Components	**SOD-M**
1. Modeling aspect concerning								
a. Business Modeling	✓	✓						✓*
b. Business-Process Modeling			✓	✓				✓
c. Web Service Modeling					✓	✓	✓	✓
d. Web Service Composition Modeling							✓	✓
2. Correspondence with MDA levels	CIM	PIM	CIM, PIM	PSM	PIM, PSM	PSM	-	CIM, PIM, PSM
3. Notation used	Own	Own and UML	Own	UML	UML	Own	- (Defines Tasks)	e^3 value and UML

Note: *SOD-M proposes using e^3value as a method for business modeling at the CIM level.

to express and communicate their business processes to all the stakeholders involved in IS development, thus bridging the gap between experts in the domain and experts in the design and construction of ISs.[30] From our point of view, the UML Activity diagram is enough for this purpose, being well known by most business analysts for its similarity to workflow charts. However, as the BPMN meta-model is an extension of the UML 2.0 activity diagram meta-model,[41] the use of BPMN is allowed in SOD-M, so it can be used in our method for high-level business process modeling.

Finally, as explained previously, SOD-M ***proposes the use of the UML notation for the modeling of the IS, and adopts the notation defined by e³value, for the specific case of high-level business modeling***.

In general, this overview of current proposals related with this work leads to the conclusion that, *at present, there are no proposals defining a service-oriented approach to the complete development of information systems.* Many existing WIS development methodologies will be observed to have made efforts to give support to the new technological proposals for the Web (Web services, business process automation, etc). Nevertheless, in general, these methodologies continue with their traditional approaches to WIS development, sometimes including minimal modifications mainly only reflected in the hypertext of the Web application. As SOD-M considers the business services (those functionalities that the end-user of the system needs) at the beginning of the WIS development process, it permits the construction of more easily navigable WISs.[14]

3. A Summary of the Method

SOD-M (Service-Oriented Development Method) is a method for the service-oriented development of ISs. It has the following main features:

- ***It defines a service-oriented approach for the development of ISs*** providing guidelines for building ISs based exclusively on *services*, using them as first-class objects for the whole process of the IS development. As said above, this approach facilitates the development of service-oriented applications as well as their implementation using current technologies, such as Web services.
- ***It is an MDA-based approach*:** it proposes a set of models extending from the CIM level, the highest level of abstraction of the MDA, to the PIM and PSM levels. Thus, by means of mapping between the models, SOD-M provides the benefits of the alignment of high-level business processes with technologies currently available for the SOC paradigm.
- ***It uses UML as the modeling language and defines a specific UML profile for the service-oriented development of ISs.*** This profile, called the SOD-M UML Profile, includes all the modeling elements needed at the PIM and PSM levels for the development of ISs from a service-oriented perspective.

SOD-M focuses on the development of the behavioral aspect of ISs and defines guidelines for building the behavioral models from high-level business modeling. The following sub-sections present a brief description of the new set of concepts proposed by SOD-M for developing the behavior of ISs from a service-oriented approach, and the modeling process proposed. Section 4 provides, though a case study, a better description of the process as well as the models of SOD-M.

3.1. *SOD-M concepts*

SOD-M proposes a new service-oriented approach for the development of ISs. It specifies two perspectives or points of view that need to be analyzed in order to develop an IS:

- ***Business Perspective:*** focusing on the characteristics and the requirements of the business in which the IS will be built.
- ***System Perspective:*** focusing on the functionalities and processes that need to be implemented on the IS in order to satisfy the business requirement.

SOD-M provides a set of concepts necessary for modeling both perspectives. Some of them are already known, while others are new and are proposed as part of this work. By means of a UML class diagram, Fig. 1 shows the different concepts on which SOD-M is based, and the relationships between them.

All of the concepts shown in Fig. 1 have a representation in some of the different models proposed by SOD-M. Firstly, those **concepts corresponding to the business perspective** describe the elements inherent to the business, and they will be represented on the CIM models proposed by SOD-M: *value model* and the *business process model*. Secondly, those **concepts corresponding to the system perspective** are elements used to describe the functionalities and the processing of the system, and they will be represented on the PIM and PSM models proposed by SOD-M: *use case model*, *extended use case model*, *service process model*, *service composition model*, *Web service interface model* and *Web service composition model*. Finally, those **concepts corresponding to both perspectives** describe elements that can be analyzed from both points of view, and, as we shall see later, permit the alignment between high-level business models and IS ones.

For the sake of space, all these concepts are described later together with the case study, although their semantics and notation are described in detail in App. A.

3.2. *The process*

The modeling process of SOD-M (see Fig. 2) includes models that are in correspondence with: (a) *the MDA perspectives*: CIM, PIM and PSM level, and (b) *the SOD-M perspectives*: the business and system perspectives.

As Fig. 2 shows, the process begins by building the value model and business process model, and enables specific models for a Web service platform to be obtained

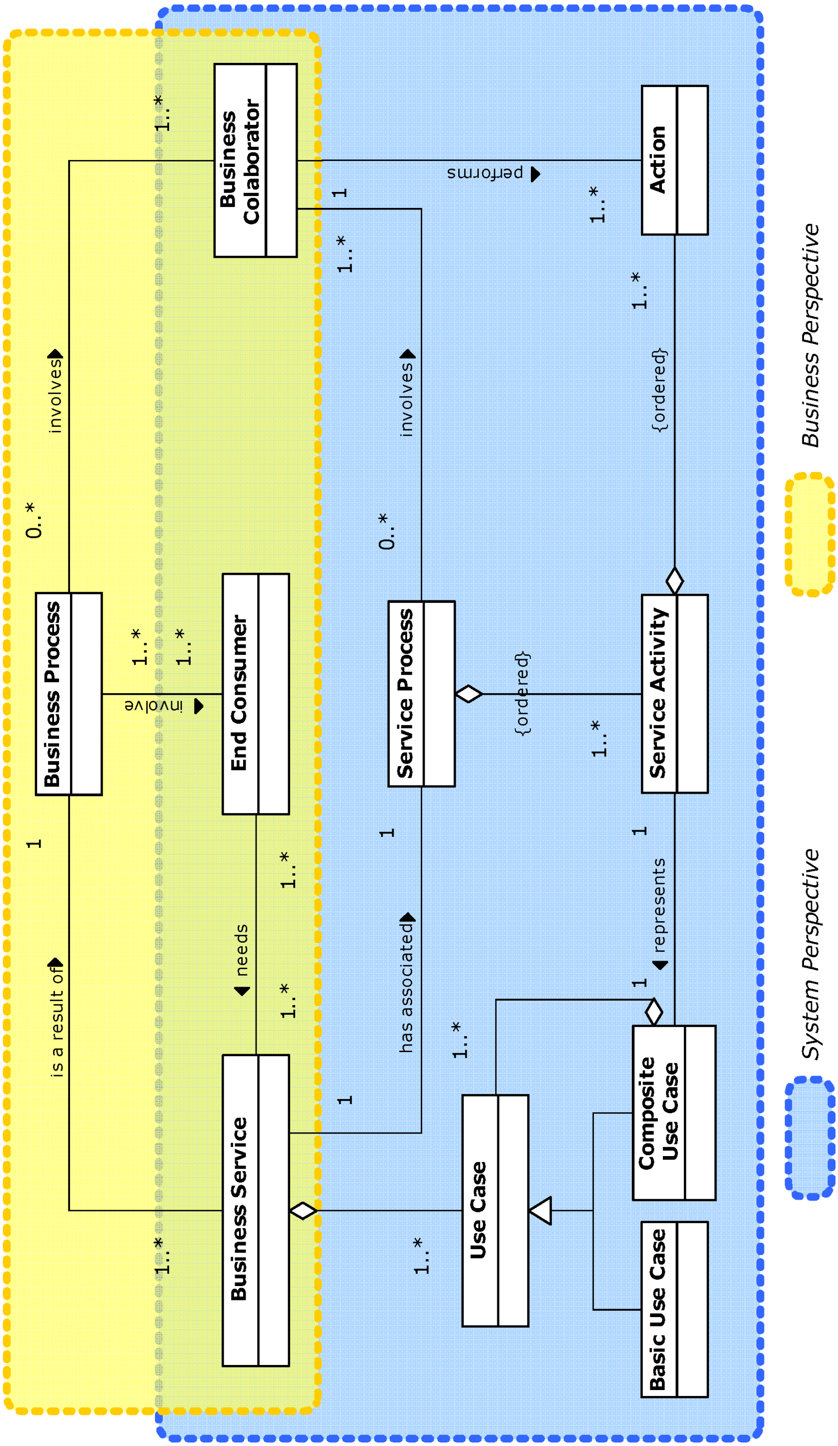

Fig. 1. SOD-M metamodel.

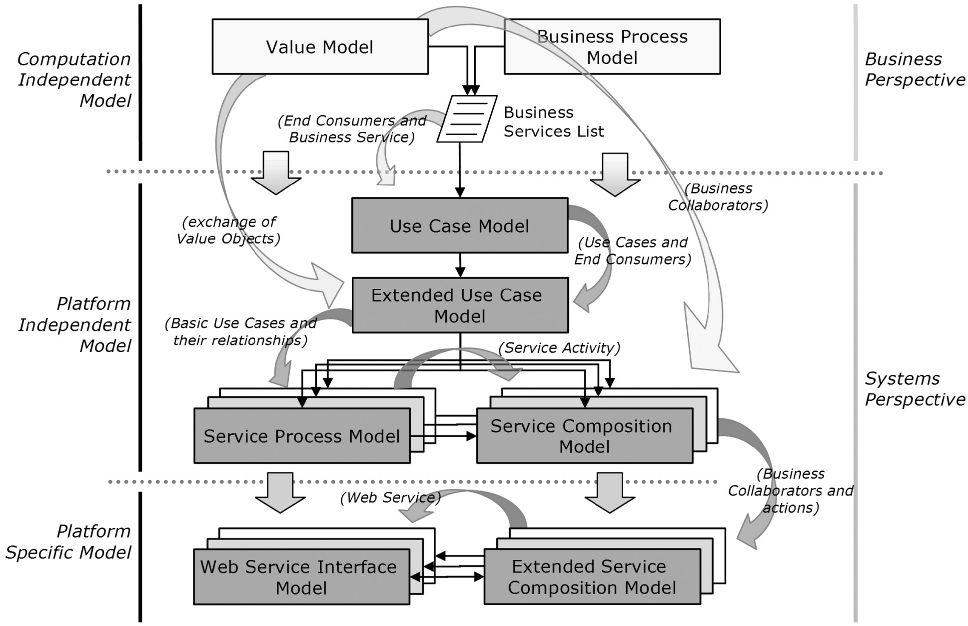

Fig. 2. SOD-M process.

as a result, thereby making it possible to make the transformation to the most typical technology related to the SOC paradigm, Web service technology.

The SOD-M process consists of several steps, each one related to the generation of a different model. From here on, we will describe how to build all these models with reference to the GesIMED case study.

4. Case Study: GesIMED

The case study presented in this paper consists of a *Web IS for medical image management and processing* called GesIMED,[2] which we are developing in collaboration with the GTEBIM research group at Rey Juan Carlos University (RJCU).

At present, medical image modalities such as Positron Emission Tomography (PET) and Functional Magnetic Resonance Imaging (FMRI), allow us to obtain images of quantitative and qualitative cerebral activity. Neuroscience researchers, especially in the clinical field, such as neuroradiologists, neurologists and neuropsychiatrists and neuropsychologists carry out FMRI and PET studies. To do their work, they first need to obtain images and then have them processed in very different ways for improvement, correction and statistical analysis.

GesIMED[17] is a Web IS operating successfully in several medical and research centres in Madrid, Spain, facilitating the work of neuroscientists and clinicians. It aims to offer neuroscience researchers (a) *an easy-to-access historical medical image store* and (b) *a duly standardized method for processing and analyzing the images.*

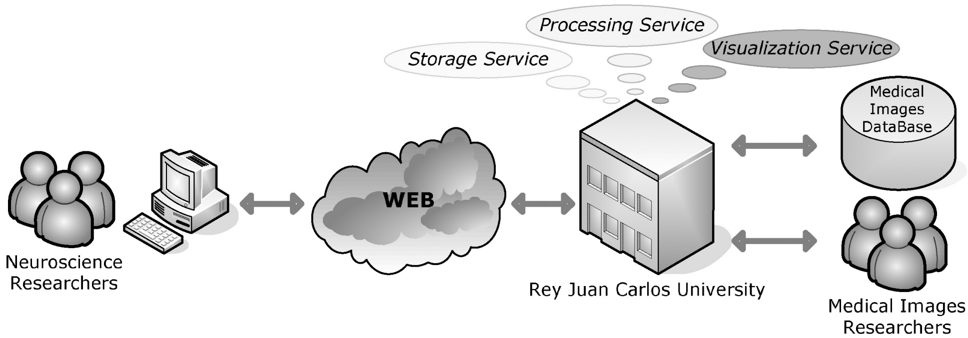

Fig. 3. Layout of the GesIMED system.

It also stores the processed and original images together in a database. Figure 3 shows a layout of the GesIMED system.

The medical images researchers at RJCU offer three specific services to neuroscience researchers:

- ***Storage Service:*** A database that stores the images and the results of the processes performed on them. Neuroscience researchers may access and consult them (by study, pathology, etc.) for diagnostic purposes, research or even teaching.
- ***Processing Service:*** Two kinds of processing are offered to neuroscience researchers: analysis and segmentation. The former consists of the measurement of physical and physiological parameters, while the latter refers to image segmentation, a typical example of which would be the extraction of the bony part from a radiological image. Researchers can request processing of images stored in the database and upload medical images to be processed.
- ***Visualization Service:*** The visualization process consists of image reconstructions (e.g. creation of 3D images from 2D images) and multimodality functions (e.g. creation of both anatomical and functional image views within a single image). Following the same procedures as in the processing service, researchers may request visualizations of medical images stored in the database or upload them for viewing later.

4.1. *Business modeling: CIM level*

CIM models are used to describe the environment in which the system will be used, with no direct reference to how the system will be implemented.[30] In this section, we shall describe the CIM models using the techniques proposed by SOD-M (see Fig. 2): the value model, the business process model and the business service list.

4.1.1. *Value model*

This model represents a business case graphically as a set of value exchanges and value activities performed by business actors and allows us to understand the business environment in which the IS will be used. It also allows us to identify the *end-users* of the business, the *business services* that will be offered by the systems to satisfy the needs of the consumers and the *business collaborators*, those entities that collaborate in providing the business services.

The value model is obtained by applying the e^3*value* business modeling method,[19] which defines a set of concepts, originating in business administration sciences and marketing, and proposes its own specific notation for their representation in a value model. A value model shows *actors* exchanging things of economic value (*value objects*) with each other. It is also possible to represent a *market segment*, a set of actors that ascribe value to objects from an economic perspective. The value model furthermore shows *value activities*, activities performed by an actor who expects them to be profitable. A value activity may be, for example, a service offered to an actor.

Using the e^3value method, we have explored the business model of the GesIMED system, representing actors who wish to exchange values with another actor. Figure 4 shows the value model obtained, where RJCU is shown as an actor. Neuroscience researchers (i.e. clinicians, neuroradiologists, neurologists, etc.) and radiologists at medical centres are identified as market segments. The services offered by the University (storage [SS], processing [PS] and visualization [VS]) are value activities, which it expects to be profitable. The value objects are the images, the results of image processing and of the visualization process, the access and querying possibilities, and the fees. Usually neuroscience researchers request research projects to finance the acquisition of medical images and their studies. They therefore pay fees (directly or indirectly) to the University for processing and visualizing the medical images, as well as for accessing and querying the database. They also pay fees to medical centres to obtain the images.

A value model permits the representation of *dependency paths*, which enhance the understanding of a business idea by showing all value exchanges triggered by the occurrence of an end-consumer need. A dependency path has a direction and consists of dependency nodes. A dependency node is a start stimulus (represented by a bullet), an AND-fork or AND-join (short line), an OR-fork or OR-join (triangle), or an end node (bull's eye). A stimulus represents a trigger for the exchange of economic value objects, and an end node represents a model boundary. Figure 5 shows the dependency paths for the GesIMED system.

The dependency paths identified in Fig. 5 show the followings situations. Firstly, neuroscience researchers need to obtain the images from medical centres in return for a fee; this path is indicated by the letter (a) in Fig. 5. Then they contact the University to obtain processed images and visualizations. Therefore, to obtain processed images from the University — the path indicated by the letter (b) in Fig. 5 — researchers send the images to be processed and pay the fee. The images are sent to the storage service where they are stored. The processing service obtains them from the storage service and in return the results are then stored in the database. A similar dependency path occurs when a neuroscience researcher needs

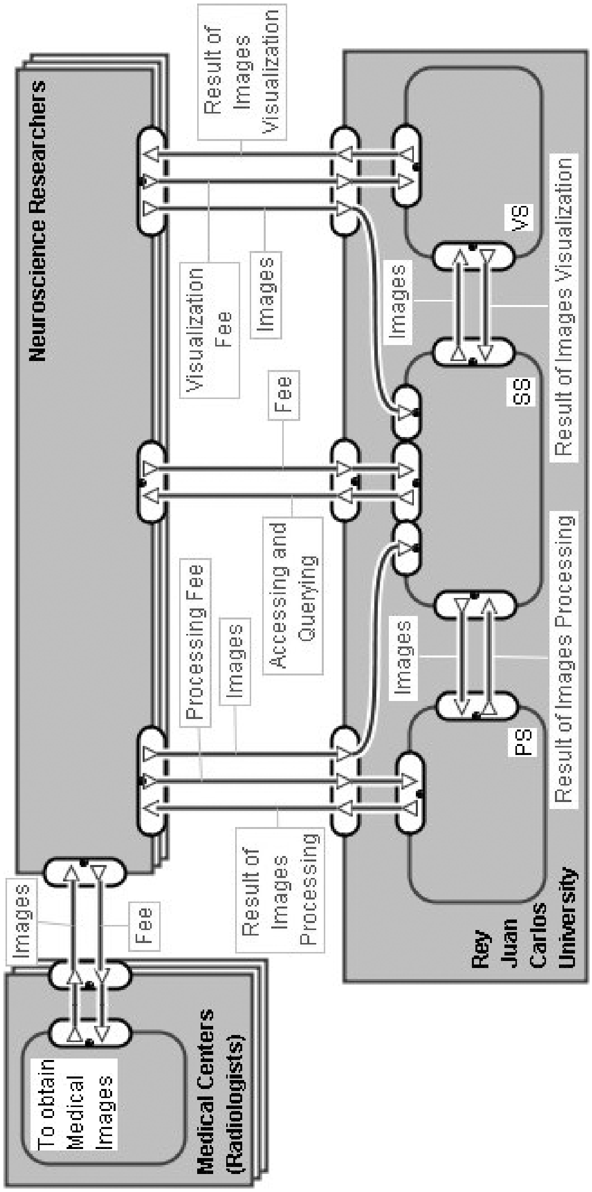

Fig. 4. Value model of the GesIMED system.

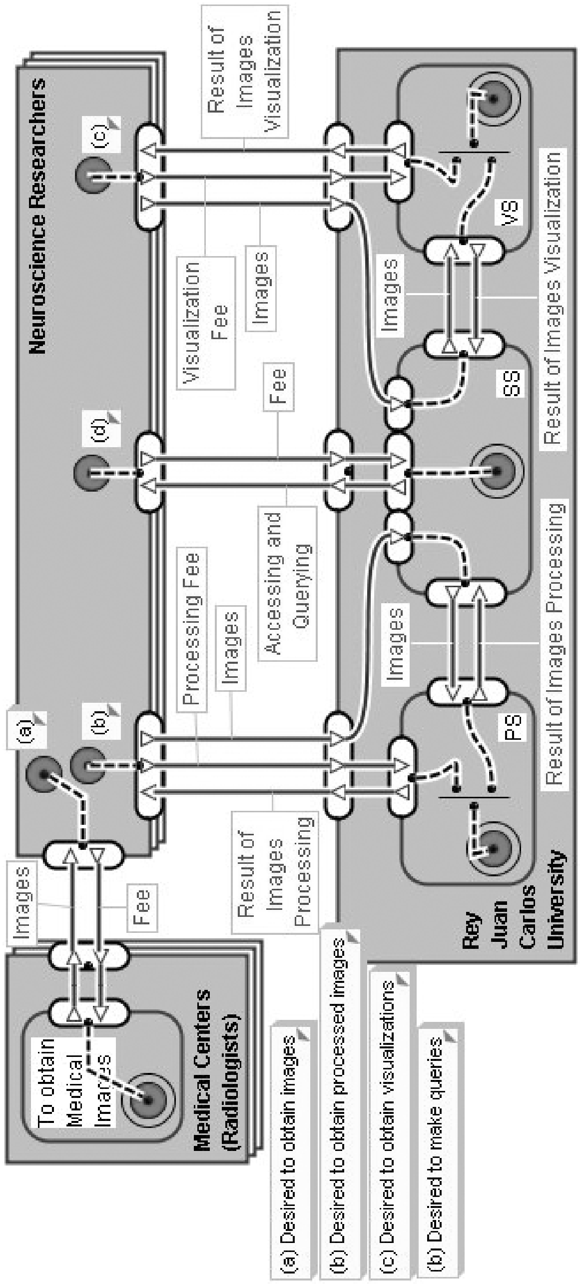

Fig. 5. Dependency paths of the GesIMED system.

visualizations — the path indicated by the letter (c) in Fig. 5. Finally, researchers may access and query the database in return for an access and inquiry fee; this path is labeled (c) in Fig. 5.

4.1.2. *Business process model*

The business process model is used to understand and describe the *business processes* related to the environment in which the system to be built will be used. A business process describes a set of tasks that need to be performed to achieve a given business result.[36] SOD-M proposes the representation of the business process model through a UML activity diagram. This model allows the identification of the *business services* that will be offered to the end-consumers of the IS. Each activity or set of activities of this model can give rise to the identification of one or more *business services*.

Figure 6 shows a business process model developed for the GesIMED WIS case. This model is built by identifying the set of tasks that need to be performed to give certain business results. It shows the set of activities that the neuroscience researchers must carry out to perform their research tasks. Thus, the researchers must first obtain the images, requesting them from various medical centres, or accessing specialized medical image databases providing researchers with different types of medical images. Having acquired the images, the researchers ask different centres, such as the RJCU, to process them. Note that in the previous model, the value model, no business processes are represented, although it is possible to use the business knowledge acquired from it to build the business process model. So, for instance, in this model we represent a neuroscience researcher's possibility to "get

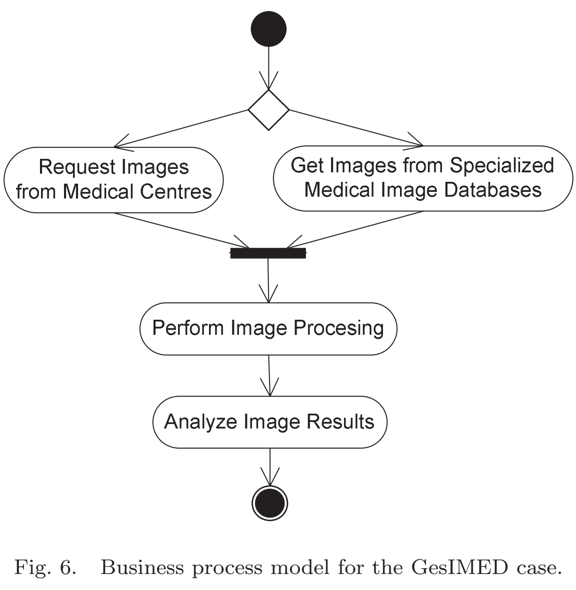

Fig. 6. Business process model for the GesIMED case.

medical images" from the medical centers, and to "get processed medical images" from RJCU. But it should be noted that both activities have been previously identified in the value model (see Fig. 5) although in the business process model, unlike the value model, we show the order in which these activities must be carried out.

4.1.3. *Business service list*

The list of business services is a textual description of all of the services forming part of a business and offered to satisfy the needs of its end-consumers.

It is to be noted that the above models are computationally independent models, which therefore do not include the concept of IT systems. In order to model the behavior of IT systems, (in this case the WIS supporting the GesIMED), their scope must first be determined. The purpose of the list in Table 3 is to collect all the *business services* offered to each end-consumer to later identify those that will be supported by the IT systems.

Table 3 shows the business service list for the case of GesIMED WIS, where the end-consumer is the neuroscience researcher. It is obtained by identifying the initial stimulus of the dependency paths from the value model, (see Fig. 5) and those activities of the process permitting the satisfying of one or more of the needs of end-consumers of the business from the business process model (see Fig. 6).

4.2. *System modeling: PIM level*

The PIM models are used to model the functionality and structure of the system, but independently of the technological details of the platform upon which it will be implemented.[30] SOD-M proposes different models for the behavior modeling of the system at the PIM level. Given that SOD-M defines a service-oriented approach for the development of ISs, such models focus on the identification of the *business services* to be offered by the system, and on the identification of the functionalities and processes needed to carry them out. The models proposed by SOD-M at PIM level, described in the following lines, are the *use case model*, *extended use case model*, *service process model and service composition model.*

4.2.1. *Use case model*

As a first model of the system, SOD-M proposes building a use case model, representing only the *business services* to be implemented in the system, along with

Table 3. List of business services for the GesIMED case.

End Consumer	Business Services
Neuroscience Researcher	Service for obtaining medical images Service for medical image processing Service for medical image visualization Service for medical image querying Service for medical image analysis

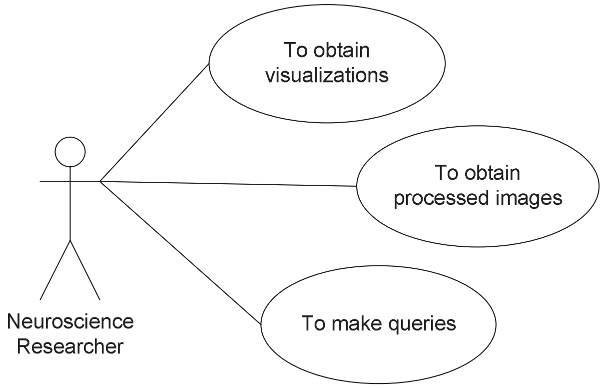

Fig. 7. Use case model of GesIMED WIS.

their relations with the *end consumers* or users of the WIS. So, SOD-M proposes representing this model by means of the UML use case diagram technique,[32] but with a different building approach, that is by identifying business services instead of typical use cases.

In this model we identify the concepts of *end consumer*, represented as an actor; and *business service*, represented as a use case. It must be underlined that, in this model, only the business services to be offered through the system are represented.

Figure 7 shows the use case model for the GesIMED WIS, where the "neuroscience researcher" is represented as an actor who is the end consumer of the services to be implemented. The model also represents as use cases the different business services that will be offered by the GesIMED WIS to the neuroscience researchers: "obtain processed images", "obtain visualizations" and "make queries". Such business services have been identified by taking into account those on the list provided by the GesIMED WIS.

4.2.2. *Extended use case model*

The extended use case model is also a behavioral model of the system, used for modeling the functionalities (services) required by the system to carry out the business services. Naturally, following a service-oriented approach it is important to identify the services comprising the business services, and also to identify the process (workflow) in which the services need to be composed. The services are represented in this model as use cases.

This model is also represented by means of the UML use case diagram technique.[32] In this model, we represent the concepts of *end consumer*, represented as an actor, *basic use cases*, which represent a set of actions performed by the system in order to carry out part of a business service, and *composite use case*, representing a set of actions performed by the system which can be broken down into different basic use cases. The basic and composite use cases are represented in

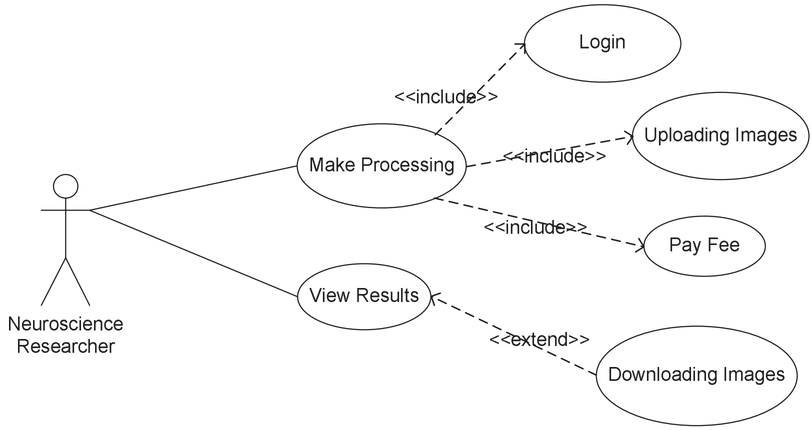

Fig. 8. Extended use case model for the business service "obtain processed images".

this model as use cases, but these are stereotyped using ≪*comp*≫ to indicate they are use cases that can be broken down into component use cases.

This model also suggests the use of *include* and *extend* relations among the different use cases identified. Their semantics in this model are the same as in the traditional UML use case model: "An *include* relation specifies the existence of a flow of events in which the base use case includes the behavior of the other" and "an *extend* relation specifies that the behavior of a base use case can be optionally extended by the behavior of another use case".[32]

As an example, Fig. 8 shows the extended use case model, which represents all the functionalities required to carry out one of the business services identified in Fig. 7: "obtain processed images". So, as Fig. 8 shows, to get images processed, the neuroscience researcher must be logged into the system, pay the querying fee and upload the images. Once processed, the resulting images can be downloaded.

4.2.3. *Service process model*

This model is used for the representation of a *service process*, and therefore shows the set of logically related activities that need to be performed in the system to carry out a business service. So, the activities of this model represent a behavior that is part of the workflow needed for the performance of a business service.

SOD-M proposes representing this model using the UML activity diagram technique.[32] In this model, two concepts are represented: *service process* and *service activity*. The service process is represented by the model itself. A service activity represents a behavior that is part of the execution flow of a business service, and is identified in this model as an activity.

Figure 9 presents the service process model, used to represent the workflow needed to perform the business service "obtain processed images" in the system, and shows that the activities have been obtained from the basic use cases represented in

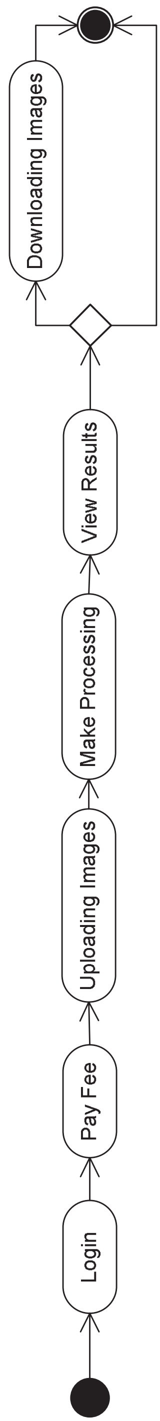

Fig. 9. Service process model for the business service “obtain processed images”.

the previous extended use case model (Fig. 8). The workflows between the service activities are represented by specifying the order of operation between them, as well as alternative or parallel activities.

4.2.4. *Service composition model*

This model extends the representation of the *service process* modeled in the previous model. The service composition model represents the workflow needed to carry out a service, but in a more detailed way, i.e. identifying those entities that collaborate in the business processes, performing some of the actions that are necessary to carry out the business services (the business collaborators) and the *actions* each of them performs. Each action identified in this model describes a fundamental behavior unit that is part of a service activity, and which represents some transformation or processing in the system being modeled.

SOD-M proposes representing this model using the UML activity diagram technique.[32] In the model, the workflow needed to carry out a business service is represented and three of the concepts previously presented in Sec. 3.1 are identified: the *service process* — the workflow itself, the *actions* — represented in the model as an activity — and finally the *business collaborators* — displayed as a partition in the activity diagram. The *actions* identified in this model are distributed between the different collaborators of the business (partitions) represented in it. As said above, a *business collaborator* can be either internal or external to the system being modeled. When the collaborator of the business is external to the system, the partition of the activity diagram that represents this collaborator is stereotyped with the word ≪*external*≫.

Figure 10 shows the service composition model, which represents the composition processes of the different actions needed to carry out each "obtain processed images" business service. To build this model, we took as input the element identified in the service delivery process model as well as some elements identified in the value model, such as business actors, market segments and value activities.

The first step required to build this model is to identify the business collaborators who carry out operations related to the services offer by the WIS. The business collaborators are obtained from the value model, taking into account the actors, market segment or value activities that offer or receive value objects when an end-consumer need occurs. In our case, we have identified three business collaborators: RJCU, which owns the IT system and is responsible for implementing the fundamental operations of the GesIMED WIS, the "storage service" (a value activity in the value model), which performs the query processing and medical image management; and the "processing service", which processes the images.

The next step is to *break down the service activity* identified in the service process model *into the activity operations supported by each service activity.* In our example, the service activity "pay querying fee" is divided into two activity operations: "validate credit card" and "register payment"; and the "make a query"

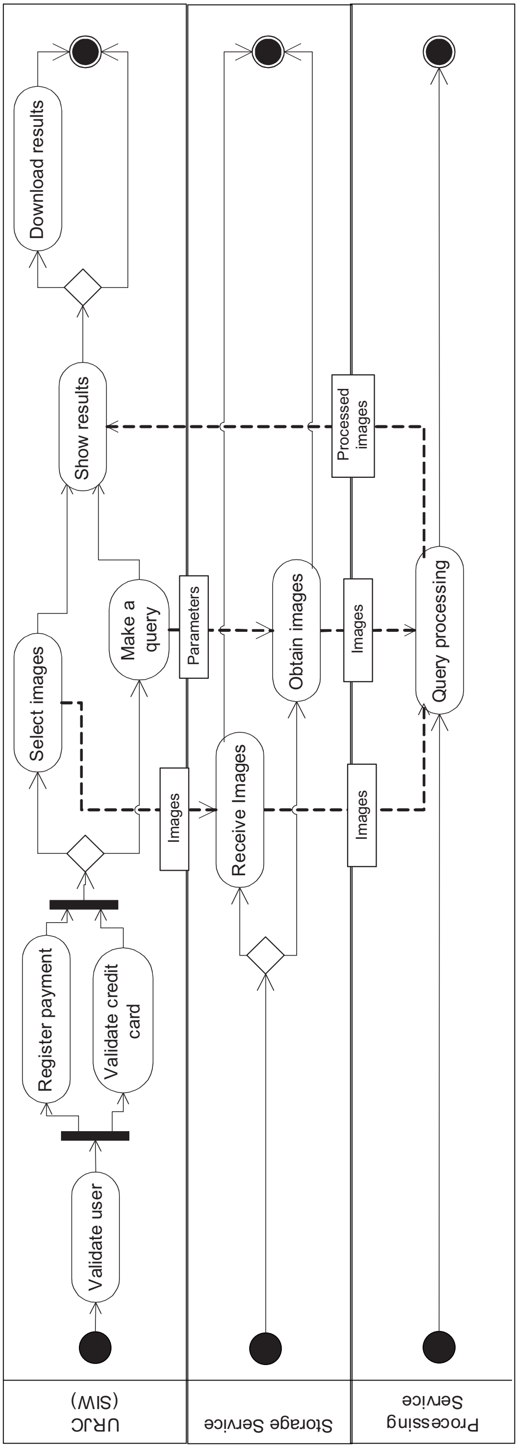

Fig. 10. Service composition model for the business service “obtain processed images”.

service activity is divided into: "make a query" and "query processing". The rest of the service activities involve single activity operations only. Once the activity operations have been identified, they have to be distributed within the partition.

The actions are obtained from the service activities, and those needed for the operation of each one are identified. Thus, for example, for the activity "perform service payment", two actions are identified: "validate credit card" and "register payment". Once all of the actions are identified, they are distributed among the different partitions of the activity diagram according to the business collaborator carrying out the operation.

4.3. *System modeling: PSM level*

The PSM models are used to combine the specifications contained in the PIM models with the details of the platform chosen for implementing the system. As Fig. 2 shows, SOD-M defines two models at the PSM level, in which some concepts described in the above models are combined with specific details of the platforms based on Web services. These models are: the *Web service interface model* and the *extended service composition model*.

4.3.1. *Extended service composition model*

This model extends the service composition process, identified in the model above, by adding specific details of Web-services-based platforms.

We represent this model by means of the UML activity diagram technique.[32] Given that SOD-M defines specific models for Web-services-based platforms, this model sets out to indicate explicitly those actions that are (or will be, where they are not yet implemented) supported by Web services. A *Web Service* is represented in this model as an activity. The *Web Service* element has as a tagged value the set of operations that it performs (operation name, input and output parameters).

Building this model, the system designer will decide which actions of the service composition process will be implemented as Web services. To do this, the following main criteria must be taken into account: (a) *the possibility of reutilization of each action,* e.g. an action that executes the login; (b) *the possibility of communication with external collaborators*, e.g. an action performed by a business collaborator which is external to the system; and (c) *the possibility of performing calls to Web services already implemented,* e.g. an action that checks the validity of a credit card. Such actions will be represented in this model as *Web Services* by means of the ≪WS≫ stereotype.

Figure 11 shows the extended service composition model for one of the business services to be implemented in the GesIMED WIS: "obtain processed images". In the example, the action "validate credit card" is represented as a Web service, since it is already a standard Web service provided by several organizations. Also the actions performed by "storage service" and "processing service" are represented as

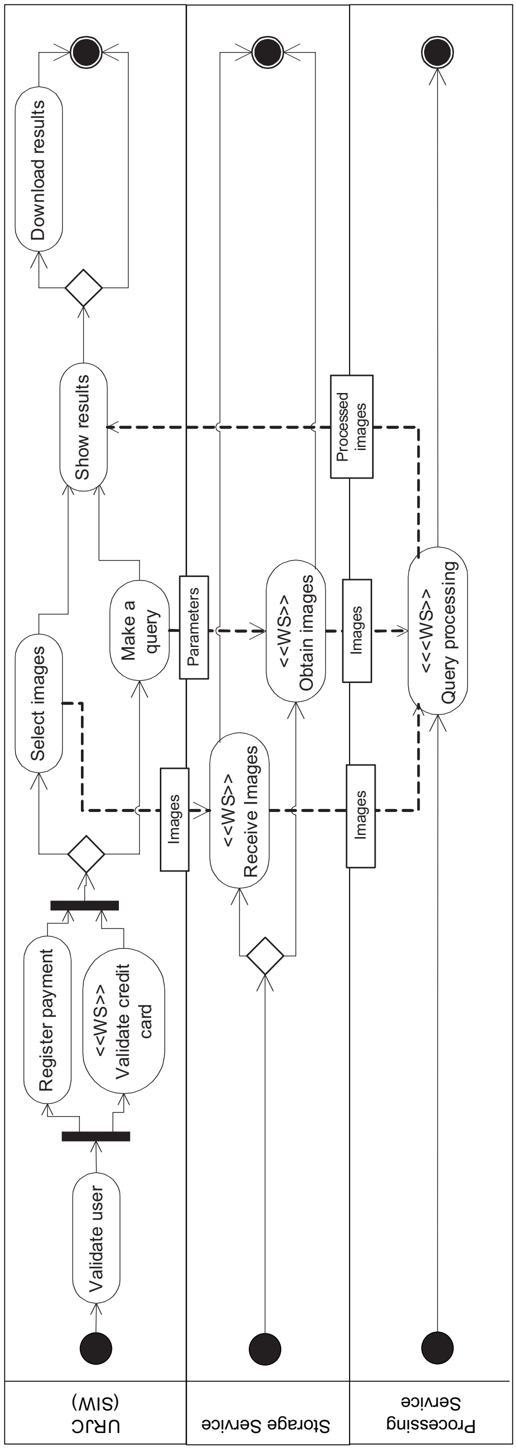

Fig. 11. Extended service composition model for the business service "obtain processed images".

Web services, since this aids the communication between the central server of the application and the different services provided by them.

4.3.2. *Web service interface model*

This model is used to describe the interface of the Web services that will be used in the delivery of each of the services offered by the system. The Web service interface model is based on the WSDL (*Web Service Description Language*) standard.[12] WSDL is the language proposed by W3C to describe Web services and allow us to describe the service interface in an XML format. The Web service interface model proposed allows us to obtain a graphical representation of the interface of a Web service, from which it is possible to generate the corresponding WSDL code automatically.

SOD-M proposes representing this model by means of the UML class diagram technique,[32] to this end defining a set of new modeling elements represented in the model, thus extending the UML class model elements. The modeling elements included correspond to the components defined by the WSDL standard for the representation of the interface of a Web service. A more detailed description can be found in Ref. 28, where it was presented.

As an example, Fig. 12 shows the Web service interface model for one of the Web services used in the development of the GesIMED WIS, credit card validation, (available at: http://webservices.imacination.com/validate/Validate.jws?wsdl). Figure 12 shows that this model allows a description of the Web service interface based on the concepts of the WSDL standard. From this model, it is possible to generate the corresponding WSDL code automatically.

4.4. *Implementing and using GesIMED WIS*

The GesIMED WIS was implemented as a Web application for several reasons: on one hand, it provides facilities in terms of connectivity, accessibility, usability and portability, while on the other hand, most of the research staff at hospitals are familiar with the Internet. Figure 13 summarizes the overall architecture, which was based on that proposed by .NET and J2EE. The application is structured in three layers: *presentation*, *behavior* and *persistence*.

4.4.1. *Presentation layer*

This layer provides two functionalities: ***Web user interface*** and ***image file uploader***. The Web user interface was implemented using Microsoft.NET framework, ASP.NET and C. Files are uploaded using JAVA API's. Figure 14 shows some screenshots of GesIMED (*MEDiMAN* is the name for English version of GesIMED). As the application is purely Web-based, no client software needs to be installed other than a standard Web browser.

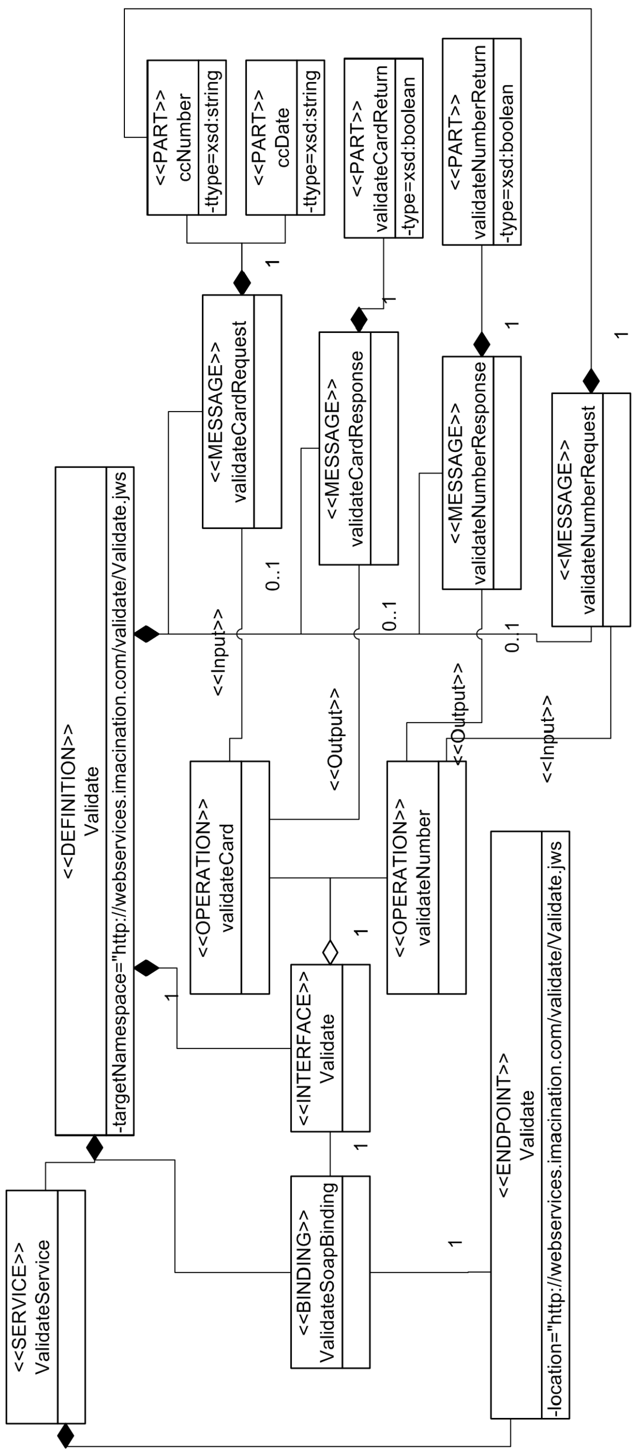

Fig. 12. Web service interface model for the Web service “validate credit card”.

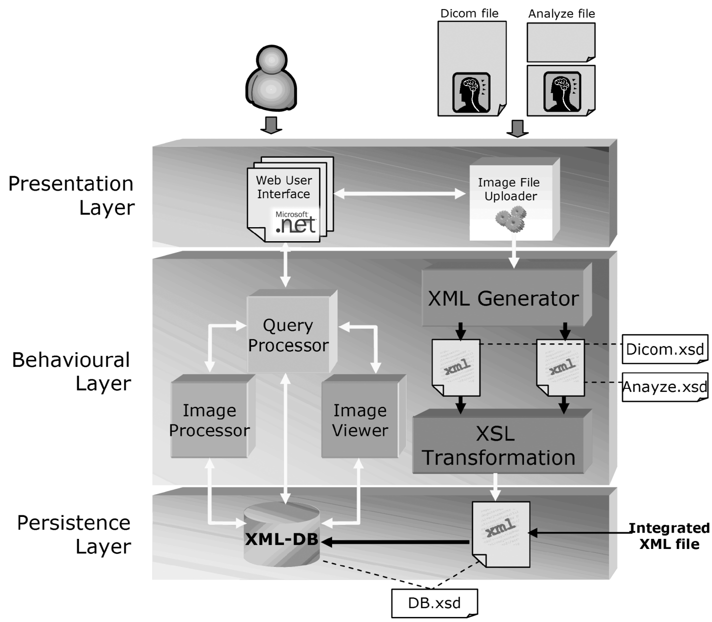

Fig. 13. Architectural description of GesIMED.

4.4.2. *Behavior layer*

The behavior layer was built on the basis of the models described above. The components identified in it implement the actions needed to carry out each of the business services provided by the system. So, for example, the *query processor* implements the actions corresponding to the "storage service" in Fig. 12. Moreover, the XML Schemas are modeled using the UML extensions proposed in Ref. 40.

The ***Query Processor*** is the component responsible for building the queries in order to execute them on the DBMS (database management system), and showing the result of the queries through the Web-user interface. To make the queries, the user fills in the Web application forms, whereupon this component transforms the data into query expressions in XPath.

The ***Image Processor*** module permits different types of processing of medical images, such as registration, filtering and statistical analysis. The software procedures are stored in the database, and can be selected by a query and launched automatically to process a set of images. The results and information on the procedures used for obtaining them are stored jointly in the database for further browsing or later inspection.

The ***Image Viewer*** shows the results as parametric mappings superimposed over anatomical images.

A description of the other components involved in the behavior layer of the architecture presented, *XML Generator* and XSL *Transformation*, may be found in Ref. 16.

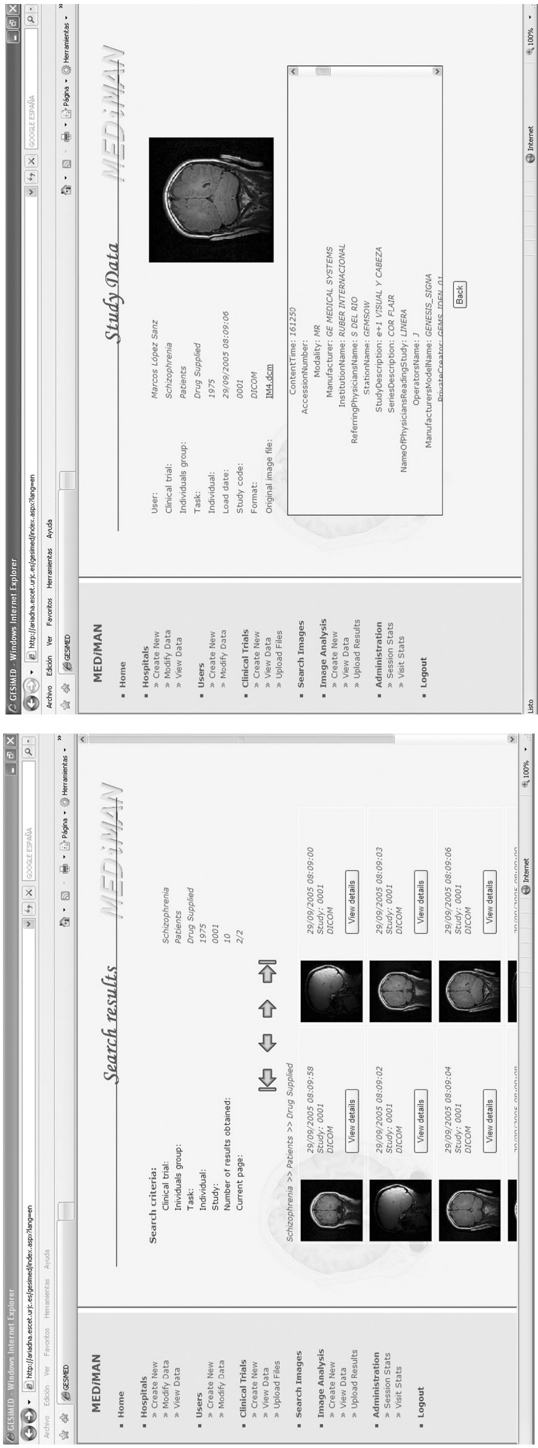

Fig. 14. GesIMED screenshots.

4.4.3. *Persistence layer*

In this case, a commercial solution based on Oracle XML DB extension has been used because of its reliability in terms of query ability, optimized updates and strong validations. A more detailed description of this layer can be found in Ref. 25.

4.4.4. *A real usage scenario of GesIMED*

In order to provide a clearer picture of the types of research tasks or clinical trials usually managed by the GesIMED system, this section describes an usage scenario. It deals with an study carried out in the Neurology Department of Fundación Hospital de Alcorcón (Spain), the aim of which was to study the response of the occipital cortex to light stimuli as a measurement of photophobia and cortical excitability in migraine sufferers. Twenty migraineurs not then undergoing attacks (8 with aura and 12 without) and 20 controls were studied with FMRI for four different light intensities, for each of which eight axial image sections of 0.5 cm covering the occipital cortex were acquired. Activation of the occipital cortex was quantified for each light stimulus by measuring the number of voxels (area) and percentage of change in baseline signal intensity.

Several post-study analyses were carried out on the data obtained, including:

- FMRI data analysis to determine the differences regarding the number of activated voxels (extension of activation) in standarized images to MNI brain (Montreal Neurological Institute brain) and the intensity of activation for each voxel (BOLD signal wave amplitude).
- FMRI data series realignment and re-slice to correct for motion artifacts.
- Statistical analysis.
- Etc.

A more detailed description of the neurological study is beyond the scope of this paper, but is available in Ref. 29, where the authors have submitted their work in the neurology area.

To manage the study described above with GesIMED, the following steps were followed in the system:

(a) *Uploading the obtained images into the medical images DB supported by GesIMED.* For that, first, people belonging to the Hospital de Alcorcón was register as user of the GesIMED; then, a new clinical trial entitled "Photo-reactivity of the occipital cortex measured by FMRI in Migraine" was created into the DB including a brief definition of the objective of the research task; finally, the obtained medical images were uploading in DICOM (Digital Imaging and Communication in Medicine) format.[1]

(b) *Executing processing service.* After the images were available to the medical images DB, the tasks described in Fig. 10 were carried out into the system each time a new processing needed to access to these images.

After this processing, the result files of the processing service were also uploaded to the medical images DB together with the original images. These files included the corrected images, statistical data series, text documents, etc.

The current version of the GesIMED system involves several medical centres and medical research institutions in Spain, such as Ruber International, Fundación Hospital de Alcorcón, and the University Hospital of Salamanca, among others. The database is stored on a RAID-5 magnetic disk system comprising twelve disks with a full capacity of 1.8 Terabytes (TB). The average volume of work managed monthly by GesIMED is about 32,000 images of different modalities, with an estimated average of 4 clinical trials a week performed on 2 different patients, each trial generating an average of 1,000 images. An evaluation about the performance of GesIMED system included a study of response times, for which we used different types of queries can be accessed.[25]

5. Conclusions and Future Work

In this paper, we describe SOD-M, a *service oriented approach to IS development*. It defines a model-driven process starting from a high-level business model and makes it possible to obtain a Web services composition model. The main characteristic of the method is the identification of the "*business services*", a complex functionality that has to be offered by the IS in order to satisfy some specific need of an end-consumer, as starting elements for the software construction. The method thus provides guidelines to identify how such business services can be obtained through Web Service composition, facilitating the alignment of the high-level business processes with the technologies currently available for the SOC paradigm.

To validate SOD-M, we tested its applicability to a set of different cases studies. In this paper, we discuss GesIMED, a case study of a WIS for medical image management and processing, first exploring the business idea of the WIS and then explaining how to obtain, as result, a Web service composition model. We describe how new concepts are identified, how models are built and also explain how to make the transformation between the models. The case study presented in this paper, along with other cases that we are currently developing (a CRM system for the Social Security in Spain and a system for participatory budget elaboration support), has helped us in the validation process for the method proposed, allowing us to refine the model and define some new stereotypes.

In addition, through the case study, we have been able to show that the Web service composition model provides a better understanding of the processes that need to be implemented. Nevertheless, in order to complete the proposals of MDA and generate code automatically from models, we consider it necessary to define more detailed and technology-dependent models. For this reason, we are currently studying some specific technological platforms for Web service development, in order to define the mechanism needed for the code generation on such platforms.

Finally, we are also working on the implementation of the proposed models and mappings in an MDA tool supporting the SOD-M method. The tool, called M2DAT, will make the mappings between models easier and, of course, will facilitate code generation from CIM models.

Acknowledgments

This research was carried out in the framework of the projects: Model-CAOS (TIN2008-03582), financed by the Spanish Ministry of Science and Technology; M-DOS (URJC-CM-2007-CET-1607), financed by Rey Juan Carlos University; and Agreement Technologies (CONSOLIDER CSD2007-0022). We would like to thank Juan Antonio Hernández Tamames, Cesar Acuña and Marcos López for their collaborations in the case study.

Appendix A. SOD-M Concepts

This appendix includes all the concepts defined by SOD-M for the service-oriented development of IS. For each one of the concepts, we describe its semantic, the models in which is represented and, for the elements that are represented by means of a UML diagram, the extended UML metaclass and its notation.

Business Process	
Semantics	Represents a set of logically related tasks that are carried out to achieve a given business result.
Model	Business Process Model

Business Service	
Semantics	Represents a complex functionality that needs to be offered by the IS, in order to satisfy some specific need of an end-consumer.
Model	Value model, Business Process Model, Use Cases Model.
Extend	UML metaclass “useCase” (in the Use Case Model)
Stereotype	None

End Consumer	
Semantics	Represents an entity that needs and consumes the services of a business.
Model	Value model, Use Cases Model.
Extend	UML metaclass “Actor” (in the Use Case Model)
Stereotype	None

Business Collaborator	
Semantics	Represents an entity that collaborates in the business processes of an organization, performing some of the actions that are necessary to carry out the business services.
Model	Value model, Service Composition Model.
Extend	UML metaclass "Activity Partition" (in the Service Composition Model)
Stereotype	≪external≫ - when it is an external to the IS - (in the Service Composition Model)

Basic Use Case	
Semantics	Represents a set of actions performed by the system in order to carry out part of a business service.
Model	Extended Use Case Model
Extend	UML metaclass "Use Case" (in the Extended Use Case Model)
Stereotype	None

Composite Use Case	
Semantics	Represents a set of actions performed by the system in order to carry out part of a business service, which can be broken down into different use cases, which can in turn be basic or composed.
Model	Extended Use Case Model
Extend	UML metaclass "Use Case" (in the Extended Use Case Model)
Stereotype	≪comp≫

Service Process	
Semantics	Represents a set of logically related activities necessary for carrying out a business service.
Model	Service Process Model

Service Activity	
Semantics	Represents a behavior that is part of the execution flow of a business service.
Model	Service Process Model
Extend	UML metaclass "Activity" (in the Service Process Model)
Stereotype	None
Action	
Semantics	Represents a fundamental behavior unit that is part of a service activity, and that represents some transformation or processing in the system that is being modeled.
Model	Service Composition Model
Extend	UML metaclass "Activity Node" (in the Service Composition Model)
Stereotype	None
Web Service	
Semantics	Represents an action that which is (or can be) implemented by means of a Web Service.
Model	Web Service Composition Model
Extend	UML metaclass "Activity Node" (in the Web Service Composition Model)
Stereotype	≪WS≫

References

1. ACR-NEMA, The DICOM Standard (2008), http://medical.nema.org/.
2. C. Acuña, E. Marcos, V. de Castro and J. A. Hernández, Web information system for medical image management, in *Proc. 5th International Symposium on Biological and Medical Data Analysis*, LNCS, Vol. 3337 (Springer, Spain, Barcelona, 2004), pp. 49–59.
3. T. Andrews, F. Curbera, H. Dholakia, Y. Goland, J. Klein, F. Leymann, K. Liu, D. Roller, D. Smith, S. Thatte, I. Trickovic and D. Weerawarana, Business Process Execution Language for Web Services Version 1.1 (2007), http://www.ibm.com/developerworks/library/specification/ws-bpel/.
4. R. L. Baskerville, Distinguishing action research from participative case studies, *Journal of Systems and Information Technology* **1**(1) (1997) 25–45.
5. K. Baïna, B. Benatallah, F. Casati and F. Toumani, Model-driven web service development, in *Proc. Advanced Information Systems Engineering*, LNCS, Vol. 3084 (Springer, Latvia, Riga, 2004), pp. 290–306.
6. M. Bichler and K. J. Lin, Service-Oriented Computing, *IEEE Computer* **39**(3) (2006) 99–101.

7. BPMI Organization, Business Process Modeling Notation Version 1.1 (2008), http://www.omg.org/spec/BPMN/1.1/pdf.
8. M. Brambilla, S. Ceri, S. Comai, P. Fraternali and I. Manolescu, Specification and design of workflow-driven hypertexts, *Journal of Web Engineering* **1**(2) (2003) 163–182.
9. M. Brambilla, S. Ceri, S. Comai, P. Fraternali and I. Manolescu, Model-driven development of web services and hypertext applications, in *Proc. of the 7th World Multi-Conference on Systemics, Cybernetics and Informatics*, *SCI2003* (Orlando, Florida, 2003).
10. M. Brambilla, S. Ceri, P. Fraternali and I. Manolescu, Process modeling in web applications, *ACM Transaction on Software Engineering and Methodology* **15**(4) (2006) 360–409.
11. F. Curbera, R. Khalaf, N. Mukhi, S. Tai and S. Weerawarana, The next step in Web services, *Communications of ACM* **46**(10) (2003) 29–34.
12. E. Christensen, F. Curbera, G. Meredith and S. Weerawarana, Web Service Description Language (WSDL) 1.1 (2001), http://www.w3.org/TR/wsdl.
13. V. De Castro, E. Marcos and M. López Sanz, A model driven method for service composition modeling: A case study, *International Journal of Web Engineering and Technology* **2**(4) (2006) 335–353.
14. V. De Castro, M. Genero, E. Marcos and M. Piattini, Including routes in web information systems as a way to improve the navigability: An empirical study, in *Proc. of Web Information Systems Engineering, International Workshops on Web Usability and Accessibility*, LNCS, Vol. 4832 (Springer, France, Nancy, 2007), pp. 505–510.
15. P. Díaz and I. Aedo, Towards efficient web engineering approaches through flexible process models, *Journal of Systems and Software* **80**(8) (2007) 1375–1389.
16. J. A. Hernández, C. Acuña, V. de Castro, E. Marcos, M. López and N. Malpica, A WEB-PACS for multi-center clinical trials, *IEEE Transactions on Information Technology in Biomedicina* **11**(1) (2007) 87–93.
17. GesIMED System (2008), http://ariadna.escet.urjc.es/gesimed/.
18. A. Ginige and S. Murugesan, The essence of web engineering — Managing the diversity and complexity of web application development, *IEEE MultiMedia* **8**(2) (2001) 22–25.
19. J. Gordijn and H. Akkermans, Value based requirements engineering: Exploring innovative e-commerce idea, *Requirements Engineering Journal* **8**(2) (2003) 114–134.
20. P. Harmon, The OMGs Model Driven Architecture and BPM, *Newsletter of Business Process Trends* **5**(2) (2004), Available at: www.bptrends.com/publicationfiles/05-04 NL MDA and BPM.pdf.
21. IBM Research, Service Science, Management and Engineering, Services Definition (2008), http://researchweb.watson.ibm.com/ssme/services.shtml.
22. N. Koch, A. Kraus, C. Cachero and S. Meliá, Integration of business processes in web application models, *Journal of Web Engineering* **3**(1) (2004) 22–49.
23. I. Manolescu, M. Brambilla, S. Ceri, S. Comai and P. Fraternali, Model-driven design and deployment of service-enabled web applications, *Transaction on Internet Technology* **1**(5) (2005) 439–479.
24. K. Mantell, From UML to BPEL model driven architecture in a web services world, *Developer Works* (2005), http://www-128.ibm.com/developerworks/webservices/library/ws-uml2bpel/.
25. E. Marcos, C. Acuña, B. Vela, J. M. Cavero and J. A. Hernandez, A database for medical image management, *Computer and Programs in Biomedicine* **86**(3) (2007) 255–269.

26. E. Marcos, P. Cáceres and V. De Castro, An approach for navigation model construction from the use cases model, in *Proc. of Conference on Advanced Information Systems Engineering, CAISE FORUM* (Latvia, Riga, 2004), pp. 83–92.
27. E. Marcos, C. Acuña and C. Cuesta, Integrating software architecture into a MDA framework, in *Proc. of 3rd European Workshop on Software Architecture*, LNCS, Vol. 4344 (Springer, France, Nantes, 2006), pp. 27–143.
28. E. Marcos, V. De Castro and B. Vela, Representing web services with UML: A case study, in *Proc. of the 1st International Conference on Service Oriented Computing*, LNCS, Vol. 2910 (Springer, Italy, Trento, 2003), pp. 17–27.
29. H. Martín, M. Sánchez-del-Rio, J. Álvarez, J. Hernández, J. Gálvez and J. A. Pareja, Photo Reactivity of the Occipital Cortex Measured by fMRI-BOLD in Migraine, *Official Journal of the American Academy of Neurology* (Submitted).
30. J. Miller and J. Mukerji, MDA Guide Version 1.0.1 (2003), Document number omg/2003-06-01, http://www.omg.com/mda.
31. OMG, OMG Model Driven Architecture (2001), Document number ormsc/2001-07-01, http://www.omg.com/mda.
32. OMG, UML Superstructure 2.0 (2003), OMG Adopted Specification ptc/03-08-02, http://www.uml.org/.
33. M. Papazoglou and D. Georgakopoulos, Serviced-oriented computing, *Communications of ACM* **46**(10) (2003) 25–28.
34. M. Papazoglou, P. Traverso, S. Dustdar and F. Leymann, Service-oriented computing, Research Roadmap (2006), ftp://ftp.cordis.europa.eu/pub/ist/docs/directorate_d/st-ds/services-research-roadmap_en.pdf.
35. A. Parikh and R. Pradhan, Modeling augmented web services Java Skyline Magazine for Java Developers (2003), http://www.javaskyline.com/WebServicesModeling.html.
36. R. Pressman, *Software Engineering. A Practitioner's Approach* (McGraw-Hill, 2005).
37. H. A. Schmid and G. Rossi, Modeling and designing processes in e-commerce applications, *IEEE Internet Computing* **8**(1) (2004) 19–27.
38. R. Tongrungrojana and D. Lowe, WIED: A web modeling language for modeling architectural-level information flows, *Journal of Digital Information* **5**(2) (2004).
39. L. Verner, BPM the Promise and the Challenge, *Queue of ACM* **2**(4) (2004) 82–91.
40. B. Vela, C. Acuña and E. Marcos, A model driven approach for XML database development, in *Proc. of 23rd International Conference on Conceptual Modeling, ER2004,* LNCS, Vol. 3288 (Springer, China, Shanghai, 2004), pp. 780–794.
41. S. Withe, Process modeling notation and workflow patterns, *Business Process Trends* (2004), http://www.bptrends.com/publicationfiles/03-04 WP Notations and Workflow Patterns - White.pdf.
42. J. Yang and M. Papazoglou, Service components for managing the life-cycle of service composition, *Information Systems* **29** (2004) 97–125.