An Evolutionary strategy for transforming the Business Object Model in a Web Services Model

DULCE RIVERO-ALBARRÁN*,**, JUDITH BARRIOS-ALBORNÓZ*,
FRANCKLIN RIVAS-ECHEVERRÍA*,**,^
* Facultad de Ingeniería, Escuela de Ingeniería de Sistemas
Universidad de Los Andes
VENEZUELA
** Escuela de Ingeniería
Pontificia Universidad Católica del Ecuador-sede Ibarra
Ibarra, Provincia Imbabura
ECUADOR
^ Programa Prometeo
Secretaría de Educación Superior, Ciencia, Tecnología e Innovación
Quito, Provincia Pichincha
ECUADOR
milagro@ula.ve, ijudith@ula.ve, rivas@ula.ve

Abstract: - In this work it's presented the definition of technical and semantic considerations that allows transforming a business object model in a set of Web services. It's also presented a case of study that shows how can be applied the evolutionary strategy for defining the web service model. The business object model represents the set of things, entities or resources that handles, generate or consume business processes. A Web services model describes the set of software services that make Web applications integrated and coherent; these applications are necessary to support the implementation of the business processes and contribute to achieving business goals. The proposed transformation strategy makes the specification of technical and semantic relationship between objects and Web services, from two complementary perspectives: (1) refining the organizational task to define technical details, that is, from the business processes level to the technologies level; and then, (2) verify this relationship and semantically validate it in reverse order, i.e., abstracting the support to the business work from the defined Web services. The described specification process is evolutionary as several repetitions of the refinement and abstraction effort is required to achieve a balanced and representative specification of the relationship between business processes and Web services.

Key-Words: - Web-services; Business model; Model transformation; Modeling strategies, business objects.

1 Introduction

In recent decades, the rapid and growing development of information technology and communication (ICT) has become a phenomenon of transformation in the way to operate and manage modern organizations, specifically, in their way of doing business. Thus, management strategies take advantage of these technologies not only for improving performance levels, productivity and profit, but also to mostly satisfy their customers [1]. This is one of the critical effects given by the globalization phenomenon, which is entirely based on the use of ICT to promote aperture and economic, social and cultural exchange between individuals, organizations and countries.

Within this context of technological dependence, organizations and companies today must have an organizational structure and technology infrastructure including Information Systems (IS) (ICT /IS), flexible enough to adjust quickly to changing demands, both in strategies and business objectives, like those arising from the development and upgrading of TIC technologies used by the business. In the first case, there is a downward transfer of the organizational change and adjustment needs that impacts TIC support technologies; in the second case, there is a transfer that leads to upward adjustment of business processes and activities, produced by changes in the ICT / IS infrastructure that supports it [12], [14], allowing to reach a logical balance between technological efficiency and business innovation. It also allows specific business sections, looking for competitive advantages, can safely be modernized [14].

Researches in this area are extensive and covers a wide range of organizational problems, ranging from technologies and systems governance to strategies definition to achieve better alignment of business interests and ICT. This work explores the technical details that describe the link and the transformation between business models and technologies and systems [5], [8], [9], [13]. The two lower levels of Figure 1. Thus, this paper presents a transformation and maintenance strategy of technical and semantic correspondence between the business objects conceptual model and technologies model describing systems that use them.

Diverse studies are currently reported in the literature dealing with this technical-semantic conversion and vice versa issue, from different perspectives. For example, the initial work of Barrios and Nurcan [3] describes a strategy to define the requirements of information and automation systems from the business object model described in levels 2 and 3 in Figure 1. The work of Gmati et al. [9] proposes a strategy for measuring the degree of technical alignment between the elements represented in the business processes level and the information systems level of Figure 1. The problem of the business / ICT alignment, has also been described in some proposals that treat it from different perspectives ranging from the technical level of ICT support, which attempts to measure the degree of correspondence between databases, legacy technical objects from active systems and information requirements of an information system that supports a changing business process [21], to settle at higher management levels where strategic business objectives are defined seeking alignment with ICT strategies that supports it [3], [12]. Both perspectives come out of the context of this work, which is located at an intermediate level, seeking the conceptual correspondence between organizational task (business processes and consequently business objects) and the implementation of solutions that properly match this task (web services), through the definition of business services.

A business service is a business function that can be decomposed into other business services to provide a solution to the company or organization. The main idea is the reuse and composition of these services according to the changing needs of the organizational processes. Business services are implemented through service-oriented architecture (SOA); these are the result of an architectural approach for the organization of ICT resources. Thus, a Web services model or architecture (WSM) describes the set of integrated and consistent Web applications needed to support the business processes of a company in order to meet the stakeholder’s needs. There is no simple recipe for implementing a business services correctly in an organization. There are contextual, cultural, policy and technology infrastructure variables, which are business challenges that the organization must face and consider. That is why today, architects persist in finding a method that handles the perfect combination of technology and best practices for specifying appropriate business services [1], [18].

To achieve this goal it is essential to know how and what is done in business processes, in order to discover what features should be considered in designing services that increase their value and impact. It must ne known how the processes work and which consequences can make changes in performance.

Currently there is a wide range of best practices available for defining business services in the SOA context. These practices are more focused on the results that in the formalities or prescribed theories [15]. This paper presents a combined approach (*top-down-up*) to identify and specify the Web services required by an organization.

Figure 1 contains the organizational vision structured in three abstraction levels, on which this paper, is based. The highest level contains the strategies and objectives pursued by the organization or company; the middle level contains the business processes, which represents the set of processes, activities, actors, resources (objects) and business rules, which are organized and executed so that the objectives and the organization high level strategies are achieved in the best way. The third Level is the lowest and is related to technologies and information systems, their structure and degree of cooperation. It depends on the provided quality support received during the execution of the business processes from level two. At the middle level, business processes, modeled to the level of their activities are directly related to the resources or business objects that are produced, consumed or handled through the business object model (BOM). The BOM represents the different required, produced, processed or consumed resources by the business activities and provides a basis for directly identifying the services or functionalities required to implement in order to support the business. Thus, a business object is an entity or abstract or concrete element that is relevant to the business system

or organization. For example, people, raw materials, equipment, financial resources, data related to an activity or product; suppliers and customers, bank accounts, human resources, are also treated as business objects. A Business Object Model (BOM) contains the entire set of elements or objects relevant to the business, describing, among others, their properties, behavior, structure and type, the set of possible values that may have their attributes in a given time, the variability with which these values may change over time, dependency ratios, aggregation or association between them [3].
An enterprise architecture consists of an Business Process architecture, specified through Business Models (BOM), a data architecture, applications architecture specified by the Web Services Model (WSM) and technology architecture.
In this paper, a business service is identified using various sources of information, such as BOM and business processes activity diagrams, which can identify services that primarily provide information services. Prior knowledge related to functionality provided by existing systems are also used. Thus, an existing functionality with a high degree of use has a high priority for being enabled as a service business.

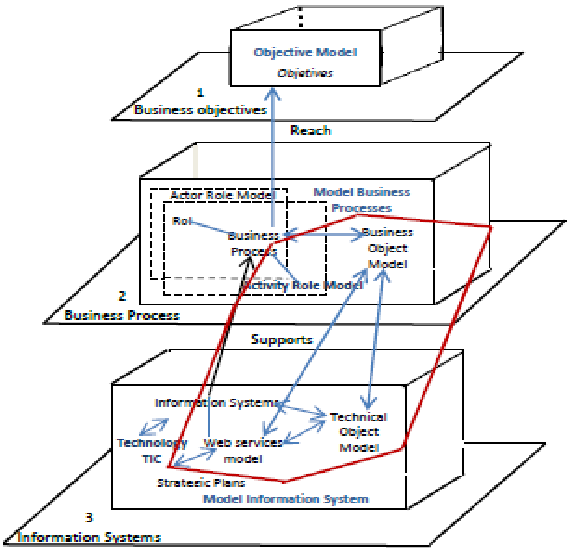

Figure 1. Organizational Vision

The aim of this paper is to present a strategy for specifying Web services (level 3 of information systems - TIC/IS) from the business object model and business processes model (level 2 business process - BP), both models are part of the business model shown in Figure 1. The strategy combines, in a simple and progressively way, the orientation paradigms for processes and business objects with the service oriented paradigm to manage bidirectional conversion complexity between the previously mentioned conceptual models. The strategy is specified as a method or guide for transformation between models so reuse and validation is facilitated.

This paper is organized as follows: Section 2 contains the basic concepts on which rests this research; the business vision and the models that comprise (objectives, processes, activities and business objects and information systems model) is described; This section ends with the description of Web services model principles, and main features. Section 3 describes the evolutionary strategy for discovery, specification and validation of Web services. Section 4 presents the case of study where the transformation strategy for the specification of the WSM was applied. Section 5 presents the conclusions and an outline of future work.

2 Basic concepts

2.1 The Business Model

A Business Model (BM) is understood as an organization or company global and integrated representation, through the representation of its objectives, processes, activities, actors, resources and supporting technologies, and especially through the specification of relations between them [12], [14].
Today it is considered that a BM is a vital decision-making and management level tool for the implementation of such decisions at the basic operational level related to the TIC /IS technologies that support the organization. In Figure 1 it's depicted a BM through three abstraction levels: the high level objectives and strategies of an organization (first level) which define the structure and allow the rest of the organizational levels; processes, activities and rules involved in achieving the objectives; actors responsible for implementation and the resources or business objects required, created, consumed or transformed by these business processes (second level); finally, at the last level, there are located information systems, Web services, databases, HW/ SW support and telecommunications devices that are the TIC infrastructure of the organization and support the execution of business processes, and thus is the level that supports the successful achievement of organizational objectives and strategies set by the high management representatives [3], [12].

2.2 Web Services model

The software service concept is inherited from the business world, where a service is something that a provider offers in order to meet the needs of consumers, each service is provided through a service contract [21]. A service represents the practical knowledge of the users' needs (specifically, service users and customers), and allows the organization to imagine how to respond to this need in ways that improve the service offering and customer expectations. Therefore, it is the business that defines the services to be rendered and applications that will need to meet its objectives.

An integrated services architecture defines business applications as reusable elements with components easy to change according to business functions and how these components are interrelated.

In technology, services are software systems that allow sharing data and functionality across applications over a network, they must allow the composition, be autonomous, and not having state and been able to be found [1]. Additionally, a service has an interface that describes the service contract offered, usually Web APIs that can be accessed over a network (mainly Internet) and executed on the system that hosts them.

A Web service uses the Internet and the World Wide Web, is implemented using a service-oriented architecture (SOA) and more recently REST (REpresentational State Transfer), a style that provides a new option for the use of Web Services. REST attempt to emulate the HTTP protocol or similar protocols by restricting the interface to establish a known set of standard operations (e.g. GET, PUT), this style focuses more on interacting with stateful resources that with messages and operations. SOA meanwhile, uses a standard set of software technologies for data exchange between applications such as SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) and UDDI (Universal Description Discovery and Integration) [15], [18], [19]. SOA facilitates the development of highly scalable systems, enabling organizations to achieve one of its main objectives in relation to ICT, as is the continuing evolution through the evolution of their services.

SOA has become the center of many initiatives, offering increased business agility. In an SOA, the individual functions or business processes are created as independent services with a standard interface that can be accessed by other applications, services and business processes, regardless of platform or programming language. These services can be flexibly combined to support different or changing functions and business processes. It supports the creation of modular applications, which are quickly assembled from new or existing services. It allows creation of highly scalable systems that reflect the business of the organization.

Web services are a way to implement distributed information systems that can be developed in a variety of languages to be used in different types of computer networks allowing that the services can be interconnected or can be consumed by other applications, the success of this interoperability is achieved through the adoption of open protocols and standards. Web services provides support for applications to function optimally.

One advantage of the services is reuse. A service can be shared or used by other services in order to compose more complex services. There are two mechanisms to compose services: 1) orchestration: where a service called director plans and directs other services, this director invokes the sequence of services to produce the desired effect; 2) choreography: where services interact and cooperate without the help of a director [21]. Each Web business service corresponds to business functionality. This functionality must be well defined, be self-contained and have small or not link. The software organization as services allows expose to consumers the functionality, increasing the visibility and adding value to the business. A necessary requirement for reuse, is the level of abstraction of the web service. An abstract service has a low coupling, which facilitates service composition and scalability of the architecture [4].

A key aspect to consider in the selection and specification of services is to define the appropriate level of granularity. The granularity is a relative measure of the extent that a service, from the point of view of required functionality to satisfy a need [21]. In other words, refers to the size and the scope of functionality of a service exposes in web services. Service granularity is the one which can be determined by quantity of functionality encapsulated by it. Service granularity is the one which can be determined by quantity of functionality encapsulated by it. It can be coarse grained or fine grained. A fine grain service targets small units of functionality or small amounts of data exchange. Therefore, to build complex business processes, companies would have to organize large amounts of these services to effectively automate a process: difficult and complex activity. For example, an operation to browse to a catalog by item number or item wise. On the other hand, a gross -grained service, involves larger or more complex functionality into a single interface, reducing the number of service requests needed to perform a task, but in contrast, it is possible to return excessive amounts of data, as well as being difficult to use again, or change to meet new requirements. For example, returning the whole catalog entries in a set of categories in online shopping system [7].

In designing web services model, a set of fine granulates services specification to be consumed by other services or gross granulated customers is required. Design quality, and therefore the services, is affected by the division or size of the granularity of services. Quality includes many aspects such as flexibility and efficiency. A principle to consider a model of services is simplicity "The simpler architecture is defined, the easier it is to run on it. Level of granularity depend on the level of management"

A service can then be composed of many atomic services and many composite services. The granularity of services increases as the services are increased. The composition hierarchy generated is then a nest of gross granulated which is aligned to the technical and business processes.

The set of Web services that support the business processes of an organization it is called Web Application Architecture. This architecture supports (information processing services) to the processes of an organization; its specification is called Web Services Model (WSM). This model describes the Web services composition hierarchy, which handles at its base the lowest level services for the objects management and, at higher levels of the hierarchy a set of complex services or gross grained services is located. Finally, at the top of the hierarchy the set of services that make up the applications and give direct support to business processes are specified, as shown in Figure 2.

The WSM specifies the set of services, relationships and interfaces. The services that make up this model, according to its granularity and responsibility are classified as:

1. Object Management Services: They are fine grained services, responsible for transformation or management (create, observe, transform) of business objects.
2. Business services: They are gross grained services that support one or more business processes. Consist of a set of services that are invoked in a specific sequence to satisfy a business requirement. In terms of SOA, these services are composed of a series of operations, services, which are executed in sequence according to certain business rules. These services consist of one or more object management services, business services defined in previous interactions or reusing services (wrap) using some legacy functionality.
3. Business applications: They are the grossest grained services that define the set of services that support business processes. They are built by composing a set of business services.

2.3 Correspondence between levels and submodels of a BM

As shown in Figure 1, the of Business Process (BP) model of Level two, is composed, among other submodels, of the business objects model (BOM), which represents the set of resources that are created, manipulated, transformed and consumed during the execution of business processes. This BOM model has its equivalent in the third level, which represents the set of technical business objects models (TBOM). The technical objects model in the third level, specify the business objects that are persistent and are required by the information systems to support the execution of business processes and decision-making throughout the organization. In addition, this technical TBOM model may include other objects that are representative of data related to the internal or external actors, and other entities of the context in which the organization operates.

Figure 2 shows the model concepts used on third level of the BM, the information systems. This model extends the information systems model presented in [3], incorporating the new entity required for implementing Web services, called Web services model (WSM). The proposed strategy allows identifying the services that make the WSM establishing a coherent and consistent correspondence between concepts in Figure 1.

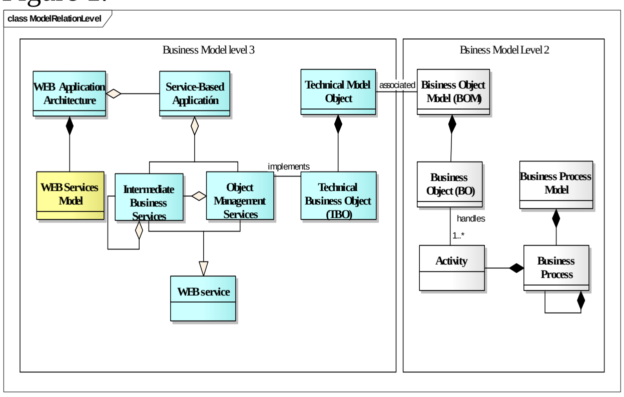

Figure 2: Correspondence between the concepts of business processes and information systems

3 Evolving strategy for transforming business objects into web services of a WSM

The transformation strategy recommends a sequence of steps to perform consistently over time to gradually transform the Business Object Model (BOM) in a Web Services Model (WSM), considering the TBOM concepts of the third level from figure 2. The proposed strategy sets out a series of activities to discover and specify the Web services required by the processes of an organization.

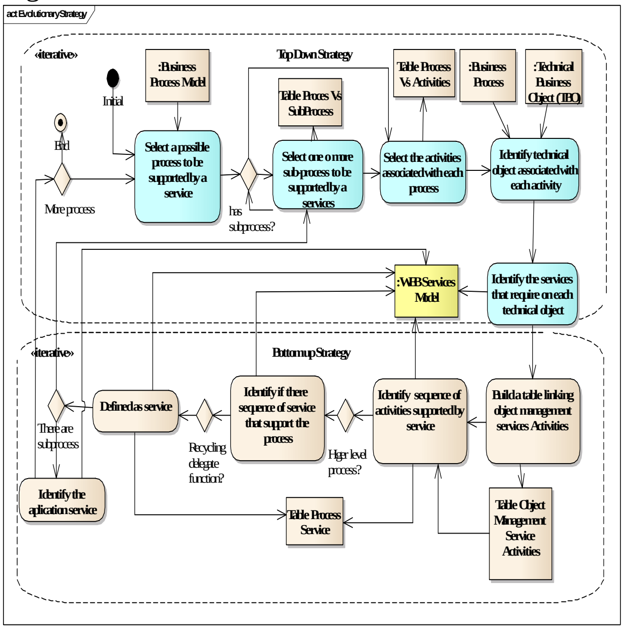

Figure 3: Activities diagram of transformation strategy

The strategy consists of two phases, the identification phase of the processes to be supported by services, known as Top-down strategy and service specification phase, called bottom-up. Figure 3 shows the sequence of activities defined in the transformation strategy.

3.1 Top-Down Application of Strategy

The strategy begins with the Business Process Model and using a top-down approach, begins with the process model, descending over the models describing the sub-processes, up to level of activities. In this path it will be identified the processes or sub-processes whose characteristics (business objects management, previous experience indicates that can be supported by some service, areas that are supported by automated systems, among others) may be candidates to be supported by services.

Upon reaching the activity level, the identification and specification process of the lower level services, called object management services starts.

3.2 Bottom-up Implementation of strategy

Identified all the services associated with technical objects and using a bottom-up approach it's proceeded to identify the business services. Finally, the highest level services, application services processes are defined, linking these services with the business logic. Each service is specified by its interface, it will be described the operation that the service will provide, the data input and output that service will provide. Implicit criteria in this process of discovery and specification will be detailed in the following sub-sections, Figure 4 shows the processes to run for the identification of services.

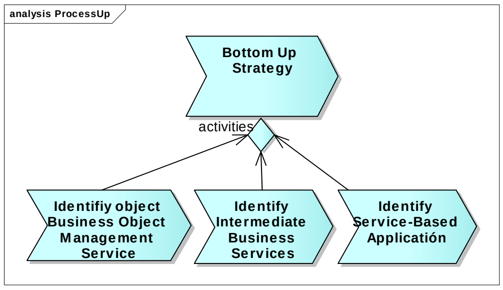

Figure 4: Processes for identifying services.

The top-down/bottom-up process is repeated as often as necessary, at each iteration is refined and/or extended the existing model. The strategy ends when all business applications based on Web services (definition of logical workflow), that the organization needs to support all business processes, are identified.

Figure 4 shows the flow diagram, the UML Business notation [16] is used. In this diagram can be identified: 1) the input that strategy requires, which are Business Processes Models (BPM) and Business Objects Models (BOM); 2) The output, which is the Web Services Model (WSM).

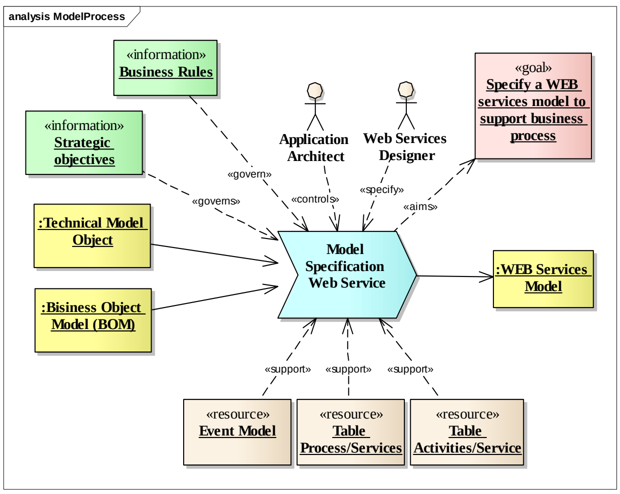

Figure 5: Processes Diagram for specifying Web Services

3.3 Processes to identify services

Process: Identify Business Object Management Services

It is used to identify the services responsible for managing (watch, create, consume, transform) business objects. These are services that are reused by higher-level services, requiring to manipulate technicians' objects. It begins with the selection of the activities associated with the processes or sub processes that can be supported by ICT.

The activities diagrams serve to identify the technical objects and the actions that each activity performs on these objects. With this information, is built a matrix relating technical Objects vs. activities, along with the actions taken (change state, notify, search state, among others). Once they are identified, it is proceeded to specify the interface of each service. These are the fine-grained services and are located at the base of the services hierarchy.

Process: Identify Intermediate Business Services

There are identified the services required to support the business processes or sub processes identified as candidates to be supported by services when the top-down strategy was applied. If a hierarchy of sub processes is identified, it is possible that a hierarchy of services is generated, where the lowest level intermediate services are specified composing some objects management services. Then, using an iterative process, if necessary, intermediate business services of gross grain (higher level) are identified and are specified from the composition of lower level services (intermediate business services and object management services). Also, if there exists any legacy applications, those functionalities that are highly used by these processes are identified. With this information, it is proceeded to identify and specify the business services; then they are incorporated into the services hierarchy, from lowest-level service to the highest level of the hierarchy gross grained (composition).

To identify business services, there are used the sub processes diagrams, activity diagrams from business process model of the second level of Figure 1 and the object management services.

Specifically, the strategy is to select between candidates' processes a sub process. If there is a processes hierarchy the lower level, and together with the activity diagram associated with it and object management services. Some sequence of services to support this process is identified, for example, a process for approving applications consists of the services for changing the status of an application and notify to the applicant. If there is a sequence, it is identified as a possible service. Having analyzed a set of processes that are related to each other (using services of common objects), it is observed if there are sequences of similar services including, if so, these sequence services become business services.

This process is applied iteratively analyzing, if any, higher levels processes. In these iterations, activity diagrams are replaced by sub processes diagrams of lower levels. A fundamental principle to consider for identifying services is the reuse; it must be careful when applying, because it must be ensured that services of gross grained will not affect the flexibility and scalability of the architecture. This requires adapting to changes in the objectives, rules and processes that occur over time. It is also possible to identify services that have no justification for reuse, but they have justification because it easily allow incorporating changes in an organization.

In this process it's necessary to build a matrix of relationship between objects management services. vs. sub-processes, this matrix identifies the object management services sequences, determining which are repeated sequences, which services are used in different sequence. The sequences that are repeated and whose low-level services are not used in other sequences become intermediate service, thus more gross-grained services.

The sequences that are repeated but using services that are used by other sequences are constructed from the composition of fine-grained services.

This process is repeated iteratively with the higher levels sub processes, services. vs. sub processes matrices are constructed, common patterns (invocation of the same business services or objects management) are identified in different sub processes and business services at the highest level, gross grained are specified.

It is important to know if an already identified possible business service is multiple use or not. It can be said that the best services are the most reusable, yet it is not true in some cases. Having just fine-grained services, leads to tremendous overhead and inefficiency. Obviously if there is a small collection of gross grained services that can be used in multiple scenarios it is a better choice; however, solving the problem of granularity in WSM is critical, and according to it will depend the success of an architecture. The more can be taken advantage of a service for multiple processes, the more useful it is. If it is possible to include a service within several different processes, it must be asked whether or not has the right level of granularity.

Additionally, it is possible to identify some services that are only used by a process (no reuse). But if these services may experience permanent changes in requirements must be considered as possible candidates, there are situations where agility instead of reuse is the driving force for services design. The fact that a service is subject to changing requirements drives the design of the service rather than their reusability.

On the other hand, an important element to consider if there are already developed or legacy applications, is to meet the 80/20 rule which states that 20% of the existing functionality in any given system is used 80% of the time. This makes that this 20% of functionality will become candidates for service. If it can be identified those features of the existing systems with 20% of heaviest use, they can be enabled as business services providing a greater return on investment to the organization. These features can generally correspond to an already identified service, if so it will be necessary to identify the function to be reused and avoid duplicating their specification.

Process: Identify service-based applications

The set of applications required to support business processes are identified. To identify the required applications, the business services consumed by each process are identified. This requires a relationship matrix for Intermediate Business Services vs. process. There are identified which services are consumed by a single process and services that are consumed by different processes.

Services consumed by a single process will be part of the functionality required by the business application. Each service group will form a business application. In those services that support different process, the granulated should be reviewed, to ensure that changes in objectives or rules in any of the processes can be incorporated easily, without affecting the processes sharing this service. Figure 6 shows how, through the strategy is gradually building the Web Services Model (WMS).

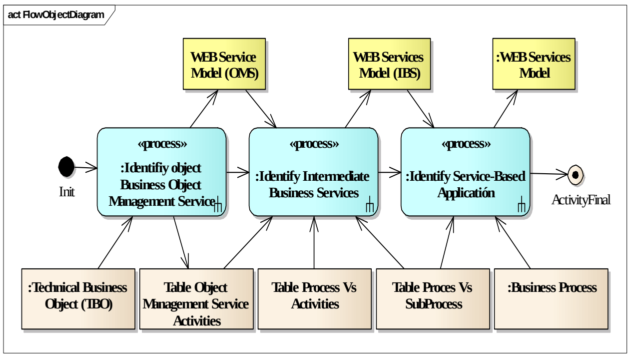

Figure 6: Flowchart of objects during the execution of the strategy

After the bottom-up strategy is finished, there are selected another processes to implement again the top-down / bottom up strategy. In each iteration arrays that have been generated are updated, and if necessary the identified services are also updated.

4. Evolutionary strategy of transformation: case of study

The strategy has been used in the development of several studies at the School of Systems Engineering [11], [16] and in development projects to public bodies. The case of study presented here shows how the transformation strategy for designing application architecture applied to the Department of Public Land Transportation. This department is responsible for the approval and monitoring procedures to be performed by all organizations providing the service of public land transportation.

The Department of Transportation, in one of its areas, is responsible for approving the various requests made by any organization that provide this service, some of them are: operation permit application, operation permit renewal, change of vehicles, among others. For the approval of each application it should be validated that meets the requirements, and approve the established inspections.

For purposes of case of study, given the complexity of the system, it only will be presented how the strategy was applied to the process of application and approval of operation permit of public transport. Figure 7 shows the process diagram of the applications management area. As it's shown, 3 processes are performed: the requests receipt, the request management process, responsible for the inspection, verification and approval and the process of statistics generation related to made requests.

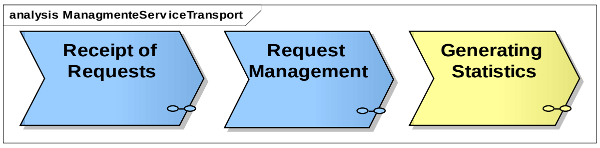

Figure 7: Process Diagram for procedures management in public transport services

In the diagram of Figure 7, as in all subsequent diagrams, they are indicated in blue color the processes or sub processes selected for this case study. The table 1 shows the Processes Vs Actors table, in which the actors involved in the process are identified and which processes and actors can be supported by a service.

Process Vs Users	Receipt of Requests	Request Management	Generating Statistics
Receptionist			
Employeer		X	
Inspector		X	
Control employee		X	
Area leader		X	X
Employee legal verification		X	
Land transport Gerente		X	X
President INTT			X
Legal representative	X		
User			

Table 1: Process vs Actors table with the processes to be supported by the services

Figure 8 shows the process diagram where the sub-processes performed in the process for receipt of requests specified by analyzing the process and in accordance with the provisions of the Top Down strategy is presented, all sub-processes are identified that they can be supported by a service. Table 2 shows the Processes vs Sub-Processes matrix where the sub processes that can be supported are specified by a service.

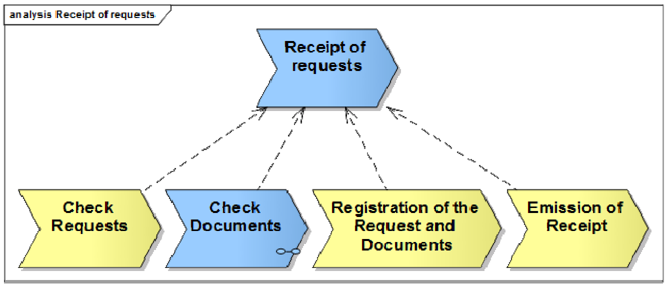

Figure 8: Sub-Processes Diagram for Applications Receiving Process.

Process Vs Subprocess	Receipt of Requests
Check Requests	X
Check Documents	X
Registration of the Request and Documents	X
Emission of Receipt	X

Tabla2: Process Vs Sub processes Matrix, identifying processes to use services

Figure 9 shows the activity diagram of the validating documents process, with this diagram two matrices are generated: the Process vs. Activities matrix which is shown in Table 3 and the Activities vs. TBOM matrix, presented in Table 4. For the preparation of this matrix, the technical objects TBOM model, shown in figure 10 was used.

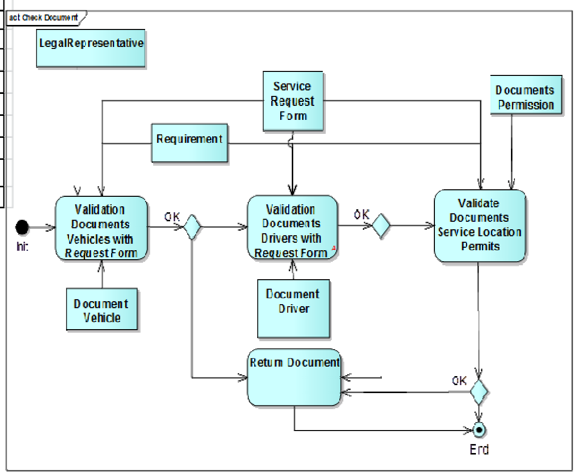

Figure 9: Sub-processes activities Diagram. Assests delivered requires

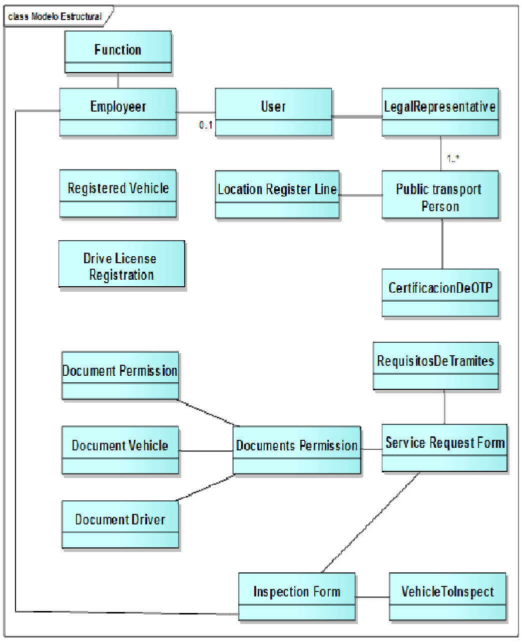

Figure 10: Business Objects Diagram

Activites Vs SubProcess	Check Documents
Validation Documents Vehicles with Request Form	X
Validation Documents Drivers with Request Form	X
Validate Documents Service	X
Return Document	

Table 3: Activities Vs Subprocesses Table, with Activities and Subprocesses to be supported by some service.

After identifying the activities that require a service for managing the technical objects, the *top down* strategy ends.

Then, it continues with the *botom-up* strategy: the services required for the management of TBOM are identified, according to the specifications presented in Table 4. The services are shown in Table 5.

Activities Vs TOBM	PublicTransport Person	Service Request Form	LegalRepresentative	Document Driver	Document Vehicle	Document Permission	Requirements
Validation Documents Vehicles with Request Form	X	X	X		X		X
Validation Documents Drivers with Request Form		X		X			X
Validate Documents Service Location Permits		X				X	X

Table 4: Activities Vs TBOM Table with activities and technical objects to be supported by a service

Activities Vs TOBM	PublicTransport Person	Service Request Form	LegalRepresentative	Document Driver	Document Vehicle	DocumentPermission	Requirements
Validation Documents Vehicles with Request Form	find() create()	FindRequest() ChangeStateVehicle() generateRequest()	find()		findDocumetVehicle() loadDocumentVehicle()		findRequirementsVehicle()
Validation Documents Drivers with Request Form		ChangeStateDriver()		findDocumetDriver() loadDocumentDriver()			findRequirementsDriver()
Validate Documents Service Location Permits		ChangeStateLocator()				findDocumetLocator() loadDocumentLocator()	findRequirementsLocation()

Table 5: Activities vs. services required to manipulate TBOM

In Figure 11, in blue, it is shown in the WSM that contains the services identified in this iteration. Then, following the strategy upstream and using Table 3 and WSM of Figure 11, services required are defined to support the activities of the sub-process Check Documents (see Figure 8). Table 7 presents the discovered services in this step. In this table it should be identified if other services are new, i.e., must be developed as new, otherwise, are services that will be generated from the composition (choreography or orchestration) of existing fine granulated services, the figure 11 shows in yellow services defined in table 7.

Continuing the *bottom up* strategy, using the WSM created until this step and Table 2, there are identified the services required to support the sub-processes in the receiving applications process. Table 8 presents the services defined for each sub-proccess.

Activites Vs SubProcess	Check Documents
Validation Documents Vehicles with Request Form	ValidateRequestVehicel(**findDocumetXVehicle, findRequirementVehicle()**) (NEW)
Validation Documents Drivers with Request Form	ValidateRequestDriver(**findDocumetXDriver,findRequirementsDriver()**) (NEW)
Validate Documents Ser	ValidateRequestLocator(**findDocumetLocator,findRequirementsLocation()**) (NEW)

Table 7: Activities vs Sub-processes Table identifying which services support an activity

Process Vs Subprocess	Receipt of Requests
Check Requests	findRequest()
Check Documents	checkDocumentVehicle(**ValidateRequest(), ChangeStateVehicle()**) - checkDocumentDriver(**ValidateDocumetXDriver,ChangeStateDiver()**) - checkDocumentLocator(**ValidateDocumetLocator(),ChangeStateLocator()**)
Registration of the Request and Doc	**X (top-Down)**
Emission of Receipt	**X (top-Down)**

Table 8: Processes vs Sub-processes, identifying which services support a sub-process

Actor Vs Process	Receipt of Requests
Receptionist	reciveRequest(CheckDocumentVehicle(ValidateRequest(),ChangeStateVehicle()) CheckDocumentDriver(ValidateDocumetXDriver,ChangeStateDiver()) CheckDocumentLocator(ValidateDocumetLocator(),ChangeStateLocator())) - generateRecive()
Legal representative	doRequest(doForm(),LoadDocumentVehicle(), LoadDocumentDriver(), LoadDocumentLocator(), generateForm())

Table 9: Actos vs Sub-Processes

Finally, using the above WSM and Table 1 containing the information of the actors involved in each sub-process and the sub-processes are identified as candidates to be assisted by a service, they identify and specify the required services. Table 9 presents the results obtained in this last step, in Figure 11, with green, the services defined in the WSM are shown for this activity.

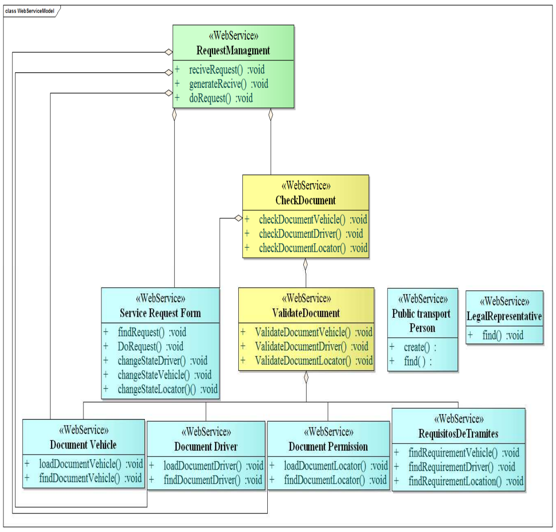

Figure 11: Web Services Model for Receiving applications process

The strategy continues applying the Request Management process, Figure 12 shows the process diagram of this process; Figure 13 shows the activity diagram for the issuing order for inspection sub-process.

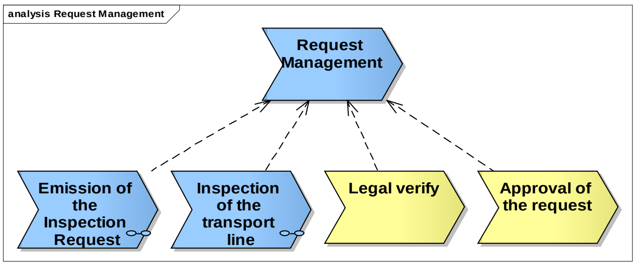

Figure 12: Sub-processes diagram of Request Management Process

The results obtained after applying the evolutionary strategy in both directions, are shown in Tables 10, 11 and 12. In each table, the results obtained are identified with the acronym T-D when the strategy is implemented in *top-down* direction, and the results obtained in implementing the strategy in the *bottom up* direction are identified with the acronym (BU).

Figure 14 shows in the resulting WSM from this iteration. The services identified in pink are implemented using legacy applications (*wapper*). The final resulting WSM is shown in Figure 15, this is obtained once the necessary abstractions was performed to define the abstract services, specifically an abstract service was defined for document management, which is then specified to make a process abstraction to different services for vehicles document handling, drivers, location, and the legal representative documents of the provider of public transport services.

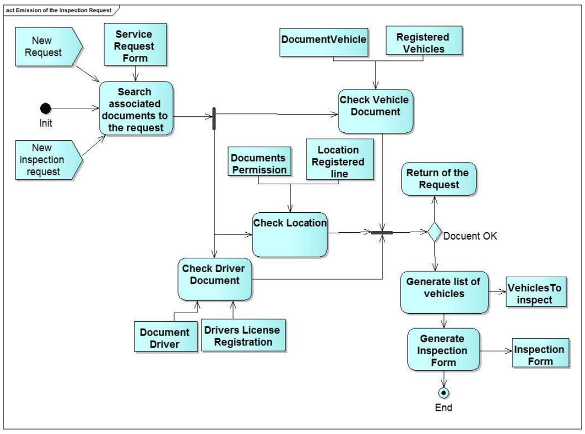

Figure 13: Activity diagram of Emission of inspection request sub-proccess

Activities Vs TOBM	Service Request Form	Document Driver	Documents Permission	LocationRegisteredLine	Document Vehicle	RegisteresVehicle	VehicleToInspect	DriverLicense Register	Inspetion Form
Search associate documento to request	FindRequest (B-U)								
Check driver document		findDocumetXDriver() loadDocumentXDrive (B-U)						findDriverLicense() wrraper existing	
Check location			findDocumetLocator() - loadDocumentLocator()	findLocator() (B-U)					
Check vehicle document					findDocumetXVehicle() loadDocumentXVehicle (B-U)	findVehicle() wrapper existing application (B-U)			
Generate list of vehicle							searchAll registerVehicle() (B-U)		
Generate Inspection Form									doInspectionForm() - findInspectionForm()

Table 10: Activities Vs TBOM with the services required to manipulate TBOM

Process Vs SubProcess	Request Managment
Emision of the inspection request	**X (T-D)** **VerifyDriver(findDocumentDriver(), findDriverLicense()) VerifyLocator(findDocumentLocator), findLocator()) VerifyVehicle(findDocumentVehicle(), findVehicle()) (B-U)**
Inspection of the transport line	**X (T-D)**
Legal verify	**X (T-D)**
Approval of the request	**X (T-D)**

Table 11: Process vs Sub-process identifying which services support a sub-process

Actor Vs Process	Request Management
Employee	**X (T-D)** **DoOrderInspect**(VerifyDriver,VerifyLocator,VerifyVehicle,registerVehicle,DoInspectionForm) **(B-U)**

Table 12: Actor vs process identifying which services support a process

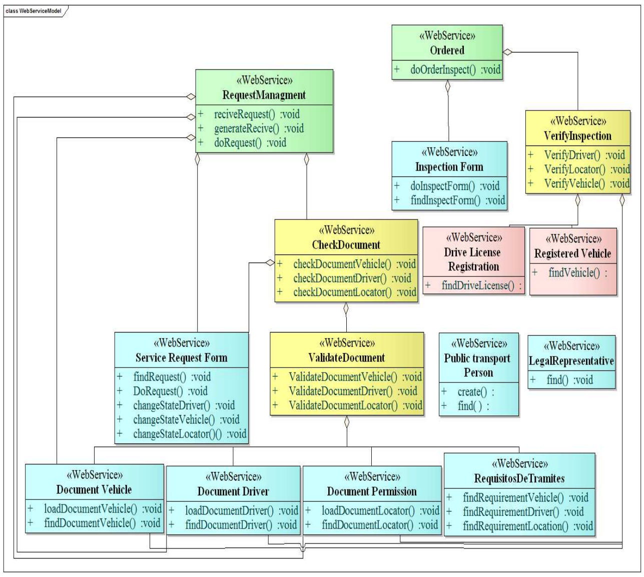

Figure 14. WSM for the receiving applications process and inspection order generation

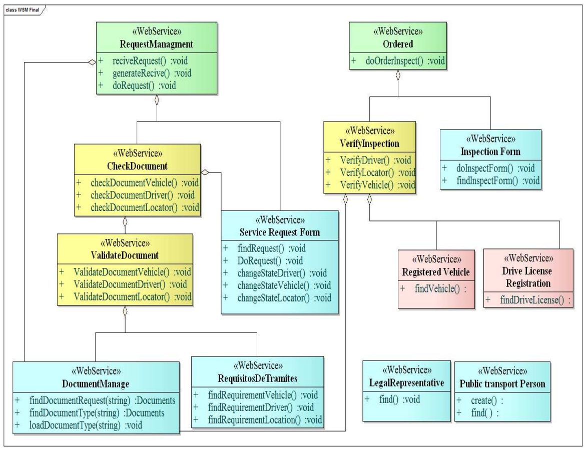

Figure 15. WSM final de recepción de solicitudes y generación orden de inspección

4 Conclusions

With this technological research based on the definition, repetition, validation, improvement and adjustment of different alternatives for the specification of Web services from the business process model, an evolutionary strategy is proposed to define models of web services from business models. These development have validated the strategy contributes to the definition of an application based services, following the concepts shown in Figure 2. Likewise confirms that the identification of a correct granulated services, allows a balance between the numbers of calls between services and reuse, allowing to define a scalable architecture and flexible services to suit changing goals and processes.

Since technical objects represent business objects, it can be said that proposed WSM is flexible and easily adapted to have implicit connection with the BOM of second level in business processes. BOM contains the data, information and business resources that manage the members of the organization for the implementation of its activities, making adjustments as required to perform on the WSM because of changes in working or operating in the organization, they can be quickly specified, developed and integrated into the operating ICT infrastructure in an organization, without major difficulties.

This strategy is part of a set of strategies for transformation between levels of the business model, which have been working and refining. Future works will focus on formalizing the method (product, process and actors models) for transforming conceptual business process models (level 2) in information systems models (level 3). The definition of a tool that supports the selection process of transformation and validation strategies will be part of future work.
The case of study apply evolutionary transformation strategy to define the WSM, in this case it was shown how to discover or identify the required services, and the strategy identifies the appropriate granularity for each service. Identifying the service to be reused being these smaller granularity, and more complex services, not normally reusable, with larger granularity.

Acknowledgment: Authors want to thanks the support given to this project by the Secretaría de Educación Superior, Ciencia, Tecnología e Innovación of Ecuador and Prometeo Program.

References:

[1] G. Allmendinger, R. Lombreglia, Four strategies for the age of smart services, Harvard business review, vol. 83, S. 131-4, 136, 138, 2005.

[2] Paige Batzan y Amy Phillips. Business Driven Information Systems. McGraw Hill. Irwin. 2008.

[3] J. Barrios, S. Nurcan "Model Driven Architectures for Enterprise Information Systems". Proceedings of the 16th Conference on Advanced Information Systems Engineering, (CAISE'04), Springer Verlag (pub), 2004.

[4] Jason Bloomberg. The concrete abstraction of the business service. http://searchsoa.techtarget.com/tip/The-concrete-abstraction-of-the-business-service

[5] B. Claudepierre, S. Nurcan, ITGIM: An intention-driven approach for analyzing the IT Governance requirements, The 3d International Workshop on Requirements, Intentions and Goals in Conceptual Modeling (RIGiM 2009), in conjunction with the 28th International Conference on Conceptual Modeling (ER 2009), 9-12 November 2009, Gramado, Brazil.

[6] Eriksson, H.E., & Penker, M. "Business Modeling with UML: Business Patterns at Work". John Wiley & Sons. (2000).

[7] Geetha J., Karthikeyan T. A Tool for Measuring SOA Service Granularity. The International Conference on Communication, Computing and Information Technology (ICCCMIT) 2012

[8] Frederic Georgel. IT Gouvernance. Management stratégique d´un système d´information. 2da. Edition. Dunod 2006.

[9] Islem Gmati, Irina Rychkova, Judith Barrios, Selmin Nurcan. Return on Experience of the Implementation of a Business-IT Alignment Approach: Theory and Practice. Workshop on Business/IT Alignment and Interoperability (BUSITAL 2010), Fifth Internati onal Conference, June 2010, Hammamet, Tunisia.

[10] I. Gmati, I. Rychkova and S. Nurcan "On the Way from Research Innovations to Practical Utility in Enterprise Architecture: The Build-Up Process",International Journal of Information System Modeling and Design (IJISMD) Volume 1, Issue 3, July-September 2010.

[11] Medina L. Fausto. Especificación de la Arquitectura de Información del Postgrado en Computación de la Facultad de Ingeniería de la Universida de Los Andes. Proyecto de Grado. Mérida Venezuela 2011.

[12] Jonás Montilva. C. y Judith Barrios. A. BMM: A Business Modeling Method for Information Systems Development. CLEI Electronic Journal. Volumen 7 No. 2. Paper 3. December 2004.

[13] S. Nurcan, Business Process Modeling for developing Process Oriented IT Systems, Information Resources Management Association (IRMA), Business Process Management Tools and Technologies track, New Orleans, USA, May 2004.

[14] S. Nurcan, J. Barrios. "Enterprise Knowledge and Information System Modeling in an Evolving Environment" Proceedings of the EMSISE workshop in the OOIS conference, Geneva, Switzerland, September 2003Frederic Georgel. IT Gouvernance. Management stratégique d´un système d´information. 2da. Edition. Dunod 2006.

[15] The Open Group : Service Oriented Architecture : SOA Features and Benefits http://www.opengroup.org /soa /source-book/soa/soa_features.html

[16] Olivieri R. Alana M. Diseño de una Arquitectura Empresarial para la Dirección de Asuntos Estudiantiles de la Universidad de Los Andes. Proyecto de Grado. Mérida Venezuela 2012

[17] Orantes, Sandra D.; López, Máximo; Gutiérrez, Agustín F.; Arquitecturas empresariales: gestión de procesos de negocio vs. arquitecturas orientadas a servicios ¿se relacionan?. Tecnura (2009): 136-144.

[18] World Wide Web Consorsium. http://www.w3.org/. Revisado el 14/07/2012.

[19] W3C Consortium. Web Services Architecture.
http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/#whatis.

[20] Xie Zhengyu, Dong Baotian, and Wang Li, "Research of Service Granularity Base on SOA in Railway Information Sharing Platform", in Proceedings of the 2009 International Symposium on Information Processing (ISIP "09) Huangshan, P. R. China, August 21-23, 2009, pp 391-395.

[21] ZapThink, LLC. How to define a business service. y http://www.zapthink.com (2007)ZapThink, LLC. How to define a business service. y http://www.zapthink.com (2007)