PROCESS ORIENTED DESIGN SERVICE: CASE STUDY FOR AUTOMATED INFORMATION SYSTEMS

Valter Castelhano de Oliveira, valter.oliveira@facti.com.br, valter.oliveira1@fatec.sp.gov.br
Fundação de Apoio a Capacitação em Tecnologia da Informação
LaPCiS – Centro Paula Souza
José Reinaldo Silva, reinaldo@usp.br
Escola Politécnica da Universidade de São Paulo

Abstract: *Manufacturing and system integration has changed in the current century, from classical models – product oriented - to information systems (IS) based models, process oriented. At the same time a new approach has emerged in the literature where the product-oriented approach, very suitable in manufacturing, has been substituted by a service-oriented approach, also based on processes. However, there is still a lost connection between the IS and the service oriented approach besides the fact that both could be process-oriented, and that such connection could benefit manufacturing automation. In this work such a link is analyzed, which brings a vertical integration, that is, the one between shop floor, engineering and business process administration. It is also shown that a IS approach to that integration brings up the problem of service orientation as a strategy to keep system levels detached while reducing the complexity of the whole artifact. The resulting service information system (SIS) is flexible and process oriented, what can fit better the goals captured during requirements elicitation and business process (BP) modeling.*

Keywords: *information systems, service science, SSME, automation, systems design, business process.*

1. INTRODUCTION

Services are becoming a trend around the world and most of the economies of developed countries are being directed to services, what in some cases account for 70% of the GDP of these countries (Bitner and Brown, 2008). An increased volume of services, generally encapsulated inside complex systems, opens a great space for studies seeking for theoretical foundations that support service improvement in quality and efficience, which entitle organizations to reach competitive advantage.

The service science emerged by the end of the twentieth century as a deep paradigm shift that would affect the way to do business and to mount the financial structure of organizations (IBM, 2004). Service science is a discipline in training, looking for academic foundations and scientifically proven methods to provide improvements in efficiency and quality of services. That opens space for innovation to go beyond a trial and error approach, currently prevailing in the service sector, and to create conditions for a systematic and scientific improvement (Lusch et al., 2008) (Spohrer el al, 2008) (Cambridge, 2007) in systems integration.

We can witness a shift from the paradigm that focuses on goods to one that focuses on services, and engineering is the basis for this paradigm shift (even if a lot of effort is addressed today to business). Thus, we face a consequent demand for a design processes that are fully focused on functionality and collaboration, linking the service and its benefictiaries. Therefore, a mechatronic design, which has a clear interdisciplinary basis, is the reference model for engineering services. Although this kind of design combines different techniques and methods, functionality is always its main reference.

The growing importance of information technology as an integrator of service and automation favors the increasing research interest in this area, especially in the foundation of its process development (Bardhan et al., 2010). On the other hand, the development of IS requires also more attention due to its interactivity. The end user expects an outcome that meets their daily needs and facilitate their work, while funders expect an appropriate return of investment and the development team hopes to provide an artifact of quality within the stipulated time and cost that fits the real needs of users and funders (Leffingwell and Widrig, 2000).

During the IS development the initial steps generate the majority of the problems, even when the consequences are detected in other stages of development. Paradoxically, the initial stages are less costly because they do not require large spend on goods and services. These steps include the phase of requirements engineering reconized as the most important in the development process (Kotonya and Sommerville, 1998) for techinical and economic reasons. According to Carr (2000), the cost to repair a requirement defect, when the system is already in production, can be up to 500 times greater than if the problem was detected and treated during the requirements phase.

This work analyses the connection between IS and the service-oriented approach, what brings a vertical integration, that is, the one between shop floor, engineering and business process administration. It is also shown that a IS approach to that integration brings up the problem of service orientation as a strategy to keep system levels detached while reducing the complexity of the whole artifact.

2. THE DESIGN OF INFORMATION SYSTEMS (IS)

Information and knowledge are becoming the main assets of enterprises and their value is going up the value of property, capital and labor. The strategic resources management of the company is associated with the development of conditions for vertical (involving business, design and engineering) and horizontal (involving material, suppliers and manufacturing) integration, the involvement of professionals do deal with information and knowledge, and a mastering of information technologies to facilitate the management of the information flow through the business process (BP).

According to Stair and Reynolds (2010, p. 7) a system is "a set of elements that interact to accomplish goals". The operation of a system is determined by its elements and their relationships, containing entries, processing mechanisms, outputs and feedback. An IS is also a interrelated set of elements or components that collects, retrieves, processes, stores and disseminates data and information in order to support strategic decisions, solve problems of information flow and coordinate the overall system or organization (Stair and Reynolds, 2010) (Laudon and Laudon, 2009).

Data is a sequence of basic facts and parameters. The information that could be derived from these data can be presented in a format meaningful and useful to monitor systems and organizations. The information value is linked to how this information helps professionals to make decisions that make possible to achieve systemic and organizational goals. According to Stair and Reynolds (2010) the characteristics that make information valuable are: accessibility, accuracy, completeness, economy, flexibility, relevance, reliability, security, simplicity, availability and consistency.

IS's are constantly changing the way companies conduct their business and are essential to extract the full potential and effectiveness of business processes (Stair and Reynolds, 2010). An IS is not only computers and programs, and in a broader perspective also involves human resources, manuals, procedures and databases. Therefore IS development covers activities designed to create new systems or modifying existing systems, including all aspects that goes from identifying the problem to be solved or opportunity to be seized, to maintenance and review phases. Accordiglgy, the value of an IS is proportional to its ability to help the company to achieve its goals (Stair and Reynolds, 2010).

However, according to the CHAOS Report (The Standish Group International, 2009) in 2009 only 32% of software development projects, including IS, were successful as the assumptions of time and costs, 24% were not successful and the remaining 44% were only recovered after modifications. Thus, in spite of the strategic importance of IS development, we are still far from solving this problem.

During the development of IS is usually the initial steps that generate the most problems, even when the consequences are detected in other stages of development. Paradoxically, the initial stages are less costly because they do not require large purchases of goods and services. These steps include the engineering phase of requirements acknowledged as one of the most important in the IS development process (Kotonya and Sommerville, 1998). According to Carr (2000), the cost to repair a problem of requirements when the system is already in production can be up to 500 times greater than if the problem was detected and treated during the requirements phase.

According to /Kotonya and Sommerville (1998), the goal of requirements engineering is the validation of requirements by stakeholders (individuals or organizations that are benefited by the system), with a long activity involving a heterogeneous group of people seeking problems, omissions and ambiguities in the requirements document, thereby generating a final version which is called a specification. Despite all this, the main problem in most market designs is precisely the absence of a reference document to be used as a basis for validating requirements - giving therefore a certain degree of quality to these specifications.

The use of a method of management of requirements based on BP as reference documented can provide a basis to substantiate the validation and acceptance of requirements, especially in IS. The adoption of a formal, accurate and standardized modeling language, can offer to the method the possibility of eliminating possible ambiguities and inaccuracies in the models. The use of formal language always creates a barrier to the implementation of these methods on the market.

For Marshall (2000) BP determines how an organization achieves its goals. According to Davenport (1993), BP is defined as a specific ordering of activities in time and space, with beginning and end, clearly identifying the inputs and outputs (and thus a process commonly referred to as planning, in the classical texts of Engineering and Science Computing). Finally, according Aalst and Hee (2002), the word "workflow" can be used as a synonym for BP, since the main objective of a workflow is to ensure that appropriate activities are performed in the correct order, by the right people and exact time (Aalst, 1998).

Thus, a problem situation or demand of a BP is an Extended Planning Problem (EPP), represented by:

$$\wp=(s_i, s_o, \Re, Y).$$

EPP consist of specifying an initial state s_i, an final state s_o, a set of resources $\Re$, e a set of outputs Y. BP is a sequence activities A (or business activities) that lead from the initial state to the state final, using addressed resources, resulting in Y. Therefore, we assume first that the BP can be formally represented or modeled. Second BP can be interpreted, and thus generate constraints to be used in the elicitation and requirements analysis. Constraints can be represented in a specific language such as Object Constraint Language - OCL (http://www.omg.org/spec/OCL/2.0/) (Takemura and Tamai, 2006), associated with the Unified Modeling Language - UML (http://www.uml.org).

The process modeling techniques (Anderson et al., 2005) (Barros, 2007) (Akhavan et al., 2006) (Rukanova et al., 2005) (Rukanova, 2005) (Shaw et al., 2007) seek to offer flexible solutions to support the maintenance of these processes, through the facilitation of change and new processes. However, a more complex analysis of the models is limited by the lack of a formal definition as to the syntax and semantics of these techniques. Through a graphical representation, easy to learn and representation, Petri nets offer great potential as a communication language among professionals from various areas of the organization, and provide the appropriate mathematical formalism for methods of analysis (Aalst, 1998). Petri nets are also used to model and analyze problems like planning even in the classic model used in Artificial Intelligence (Vaquero et al., 2007).

Petri nets have been widely studied (Murata, 1989) (Rozenberg and Engelfriet, 1998) (Reisig, 1985) (Desel and Reisig, 2004) and applied in modeling and analyzing discrete event systems, where activity sequences, parallel and competitors, are assembled to make processes.

The BP modeling and analysis using Petri net is an active area of research with several papers addressing the topic (Verbbek et al., 2007) (Or-Lin and Yang, 2007) (Guan et al., 2006) (Zhang, 2006) (Aalst et al., 2003) (Sivaraman. and Kamath, 2002) (Aalst et al., 2000) (Salimifard and Wright, 2001) (Aalst, 1998) (Adam et al., 1998).

There are several studies (Korthaus, 1998) (Marshall, 2000) (Noran, 2000) (Baker, 2001) (Jackowski, 2003), using the UML version 1.0 as graphical language for business modeling , dealing with this issue comprehensively , describing the steps and resources to be used, and stating the problems, difficulties, and inconsistencies found. Several of these difficulties have been resolved in later versions of UML (2.0 and 2.1) and certainly new supplies will come in future versions (Kobryn, 2004).

On the other hand requirements engineering (Kotonya and Sommerville, 1998) (Pressman, 2006), understanding the phases of elicitation, analysis and verification is developing rapidly, driven by the experience acquired in more technical fields of engineering such as product development and control systems. However, when development occurs in IS a much greater dependence of agents as privileged users and funders stands as a decisive element for the success of the venture (Smith et al., 2005) (Silva and Santos, 2004).

According Kotonya and Sommerville (1998) there is no ideal method for requirements engineering, because few or none of the known methods have all the necessary attributes. The main methods and techniques of requirements engineering are: data flow modeling, semantic modeling, object-oriented modeling, formal modeling, SADT - Structured Analysis and Design Technique (Ross and Schoman, 1977), CORE - Controlled Requirement Expression (Mullery, 1979), VOSE - Viewpoint-oriented System Engineering (Finkelstein et al., 1992) and VORD - Viewpoint-oriented System Definition (Kotonya and Sommerville, 1998).

Since its introduction in 1997, UML has become a standard representation for modeling object-oriented software development projects, including the stage of requirements specification. UML version 2, presented by the Object Management Group - OMG (http://www.omg.org/) in 2004, included 13 different modeling notations, ranging from high-level diagrams such as use-case diagram, which describes the interactions and relationships between actors and the basic business functions, even low-level diagrams, such as objects diagrams, which capture individual object data instances (Russell et al., 2006).

UML 2 introduces a number of differences from the earlier versions, as: greater precision in the language definition; execution semantics of activity diagram based on Petri net (Murata, 1989) (Rozenberg and Engelfriet, 1998); highly modular architecture language; new composition and decomposition recursive elements of of hierarchical modeling; and improved support for domain-specific languages (Russell, 2006) (Selic, 2006) (Kobryn, 2004). UML 2 is a language independent of tool or method for developing systems. Can be used XML Metadata Interchange - XMI (http://www.w3.org/XML/ and http://www.omg.org/ spec / XMI /) and Metaobject Specification Facility - MOF (http://www.omg.org/mof/) for the transfer of UML models between different tools.

According Störrle and Hausmann (2005), UML 2 was designed to enhance the semantic precision, which is evident in the representation of the activity diagram with new features based on Petri nets. For the basic constructs, the formalism of elementary Petri nets is maintained, however for the more advanced constructs such as exceptions, traverse-to-completion and streaming, semantic changes are required to maintain the semantics of Petri nets. According to Russell et al. (2006), this new version of the UML activity diagram 2 brings new advances, such as modeling notation for BP.

The UML is constantly evolving (Kobryn, 2004), and version 2.3 (May/2010) the latest official release of UMG. One of the key evolution of UML 2 is the System Modeling Language - SysML (Kobryn, 2004), an extension developed for modeling systems and product architectures, as well as their behaviors and features (Balmelli, 2006).

SysML is the result of an initiative shared between OMG and The International Council on Systems Engineering - INCOSE (http://www.incose.org/) to create a unified modeling language, formal and semantically reasoned that meets the needs of engineering systems. SysML is a graphical language that supports the specification, analysis, design, verification and validation of a large set of complex and heterogeneous systems, not necessarily based on software, integrating hardware, software, data, people, procedures, processes and infrastructure (Vanderperren, 2005).

Figure 1 shows graphically the relationship between SysML and UML 2. The State Machine Diagram, the Sequence Diagram and Use Case Diagram of UML 2 are reused without modification in SysML. The Activity Diagram and Block Diagram of UML 2 are reused and extended in SysML. Finally, the new Parameters Diagram and the new Requirements Diagram are only available in SysML (Balmelli, 2006).

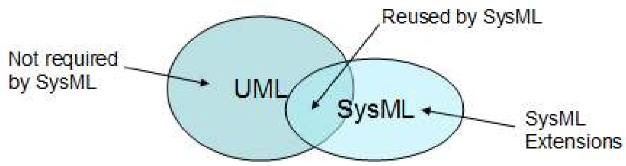

Figure 1 - Comparison between UML 2.0 and SysML (adapted from http://www.omgsysml.org/)

2.1. Matching between Requirements and BP

Figure 2 shows a method for requirements management directed the development of IS, composed of management processes and requirements applicable by the business analysts and requirements engineers, seeking to expedite the process of acceptance of the specifications of the IS from stakeholders. This method was not result of statistical sampling or testing of cases have developed, but a compilation of best practices, gained from the authors' experience, combined with formal methods and viewpoints (Leite, 1996) (Silva et al., 2005) (Silva and Santos, 2004) (Silva et al., 2006).

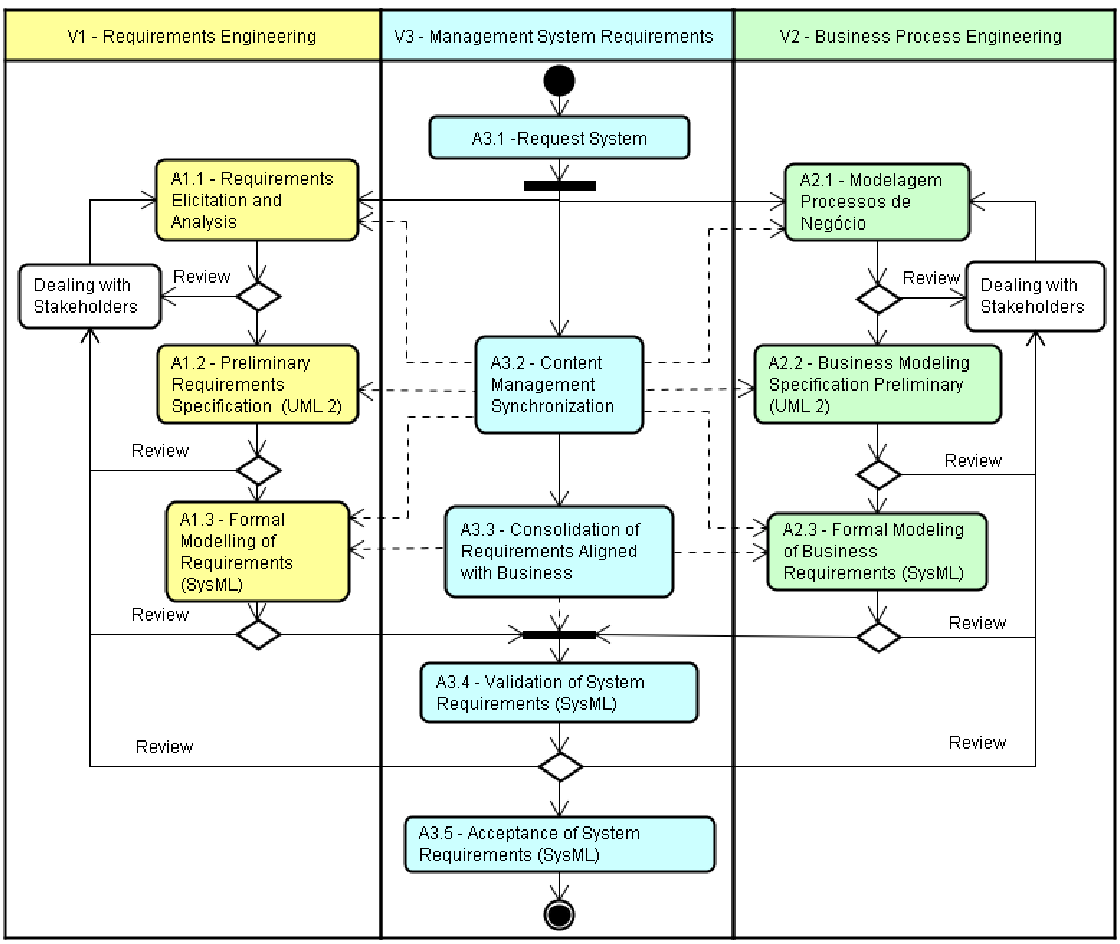

Figure 2 - Proposed method for requirements management (Source: author)

The method of Figure 2 proposes processes that allow the use of object-oriented modeling and UML 2, both in engineering requirements vision, such as engineering BP view. These processes are carried out, preferably by independent teams and possibly in different time windows. In both, this paper proposes management processes that enable the transformation of the UML 2 modeling for SysML. The method for requirements management wishes to obtain a formal modeling of requirements, using checkpoints to perform the matching between requirements and BP.

Figure 2 shows the processes of elicitation, analysis and modeling of requirements to be parallel and independent processes of business modeling, coordinated by a management scheme that is based on matching between the visions and business requirements (functional and nonfunctional).

The left column shows the activities associated with Vision Requirements Engineering (V1), whose goal is the elicitation and specification of requirements, both held by the team of requirements engineers. The right column shows the activities associated with Vision BP Engineering (V2), whose goal is the modeling of business linked to the IS request, conducted by the team of business analysts. The middle column includes the activities of management and control requirements (V3), whose management team has been active since the request from the system until final acceptance of the requirements, directing the synchronization of all activities.

The proposed method in Figure 2 has advantages with respect to a formal traditional sequence, ie, V2 is performed before V1 or in the extreme case that V2 does not exist. In the first case we lose the chance to review the impact of integration and automation that can be obtained from the insertion of IS, with the implication that V1 already contains the unification of diverse viewpoints. The V1 and V2 processes should be integrated by the Vision Management Requirements (V3), which synthesises the specifications, but based on the harmonization of the two earlier views, and therefore the efficiency of IS in achieving business results (or simplification these same administrative processes).

The benefit is the anticipation of the formalization of requirements, result of the analysis carried out separately for the BP and the requirements associated with a phase matching (made in SysML) followed by integration. Another outcome is a considerable increase in harmony with the stakeholders and thus to obtain a more convergent negotiation leading to a final specification consistent.

Usually the amount of work on the project management process (translated in V3) increases considerably, however it is noted that the possibility of successful implementation of the system increases in proportion. It should be noted that the actual work of the Project Management evolves to the extent that the timing and steps to advance the views (viewpoints) are very well delineated. For example, recommended that the project is to begin by the A2.1 stage, ie the BP Modeling. This phase can be performed in parallel with the initial requirements elicitation. However, the hope is that obtained a consistent version of the BP immediately be initiated the compatibilization with the requirements already collected and analyzed, even though this process is not completely finished.

Not performing the activity A2.1, indicates the management that this activity should be passed on to the requirements team, even increasing the cost of the project and reducing the chances of implementation success. The option to completely overlook the lack of business modeling is not considering.

Moving from the first level of synchronization (solid line between the activities A1.1 and A2.1), it follows the process of representation (UML) and where the synchronism analysis is indirect (dotted lines) and done by the management team. Team training and familiarity with the development of viewpoints is essential in this process. Note that the revisions that align processes are internal to the teams requirements and business modeling, respectively.

The acceleration of formalization happens exactly at this stage, with the introduction of SysML, which contains part of the UML representation, thus facilitating transfer semantics between them. Thus one arrives at a formal version of both the business modeling and requirements, which are now free of contradictions, many of the omissions, and have included the design constraints. So this is the time to make a second synchronization (solid lines), now with the participation of the management team. The busbar in Figure 2 represents this convergence. Once consolidated, these requirements can be validated by stakeholders. The management staff should monitor the requirements acceptance process, which may be the basis for a future contract for development and deployment.

3. SCIENCE SERVICE STATE OF ART

For Vargo and Lusch (2006) service is the application of resources for the benefit of others. Considering specifically the dominant logic of service, we have the definition of service as the application of skills, including knowledge and proficiency, for the benefit of others. The term "services" in the plural indicates the traditional treatment services, where resources are static and that create value must be operated, since the use of the term "service " in the singular indicates the new view, where resources are active, dynamic, intangible and have the ability to create value.

The services can be differentiated from goods through the following characteristics (Lovelock and Wright, 2002):

- Intangibility, services can not be tasted, felt, heard or smelled before purchase;
- Inseparability: services are delivered and consumed simultaneously. The inseparability has as one of its consequences, the presence of greatly increasing customer concern about the immediate gratification of it;
- Variability: the same service, provided by different people and / or for different customers varies as to its outcome;
- Perishability: services can not be stocked.

According to Lovelock and Wright (2002), products, goods or services, have search attributes, experience and confidence. The search attributes allow the client to evaluate the product through the perceived characteristics before its acquisition. The background attributes of a product can be evaluated after the purchase because the customer must be experienced to be appreciated. The attributes of trust are associated with the characteristics of the products, even after purchase and consumption, are difficult to assess by customers. The goods have basically search attributes, but may also present the same experience. The services are characterized by attributes of trust and experience.

The services have a higher degree of intangibility and variability, implying a higher risk exposure to customers. These risks can be classified as: functional, indicating the possibility of unsatisfactory performance results; financial, resulting in monetary loss or increased costs; temporal, with the possibility of delay; physical, incurring personal injury or material damage; psychological, due personal fears and emotions, social, including concerns about how people react, sensory, undesirable impacts on the five senses. According to Lovelock and Wright (2002), after the client consuming a service it compares with the expected quality received, evaluating it as higher quality if service performance is above their levels, as appropriate, if the delivery service is within the tolerance zone, and as inadequate if below.

Service systems are characterized by dynamic configuration of people, technologies, organizations, and also shared information between these elements, which allows vendors to create and to deliver value to its customers (Maglio et al., 2009) (Spohrer et al., 2007) (Cambridge, 2007) (Chesbrough and Spohrer, 2006).

Several systems can be considered as service systems, such as families, corporations, foundations, NGOs, government agencies, departments in corporations, cities and nations. The key concept associated with service systems is that they interact to co-create value (Spohrer et al., 2007).

The principle of co-creation of value is emphasized the joint creation of value where an entity applies his skills and the other integrates these skills with other resources through co-creation (Maglio et al., 2009).

The design of a service system depends on decisions made about the target audience, the objectives to be achieved, the strategies to be used and the intended market positioning. The service system includes three overlapping subsystems: operation, when the inputs are processed and the elements of the service product are created; delivery, which occurs when the final meeting of the elements of the service and it is delivered to the client; marketing, which involves all points of customer contact, such as advertising, sales and market research (Lovelock and Wright, 2002).

In S-D Logic (Service Dominant Logic) specialized skills are applied to the benefit of the customer, allowing the creation of value through collaboration between all parties including the supplier and customer. The goods remain important, but now treated as vehicles for the transmission of resources within the processes that make up the business (Merz et al., 2009). According Lusch et al. (2008), the S-D Logic is focused on the following conceptual framework:

- Operant resources, resources produce results through transformation of other resources, and are often intangible, such as knowledge and skills;
- Resourcing, is to create value through the transformation of a potential resource in specific benefits, relying on three key aspects: creation of resources, integration of resources and removal of resistance among them;
- Servicing and experiencing, focus on the customer experience drives the service so that your needs are met;
- Value proposing, the client is seen as an integrator of several features aimed at creating value for your organization;
- Dialog, which involves developing an effective communication based on trust, joint learning and adaptability;
- Value-creation networks, that requires a redefinition of the supply chain searching to get a network service systems;
- Exchange as learning, emphasizes the appreciation of the usefulness of the service and creating joint;
- Collaborative marketing, the customer is treated as a collaborative partner in the marketing process.

The service science or SSME (Service Science, Management and Engineering) is associated with the study of service systems (Lusch et al., 2008) (Spohrer el al, 2008) (Cambridge, 2007). As a discipline in training (Ng and Maull, 2009) (Zhao and Perros, 2009) (Li et al., 2007), the service science requires a thorough and systematic study that makes possible the creation of a framework of support, grounded theory and appropriate to be applied in several areas of knowledge.

In search of a theoretical framework for the service science, the use of S-D Logic (Lusch et al., 2008), even incipient, provides a basis for theorizing, confirmation and refinement of theoretical foundation of the science of service (Vargo and Akaka, 2009) (Maglio et al., 2009) (Lusch et al., 2008).

In the study by Barile and Polese (2010) the search for a formalization of Science Service is through the field of general systems theory (Bertalanffy, 1950) (Boulding, 1956), exploring the relationship between the proposed approach based on VSA - Viable System Approach proposed by Beer (1984) and recent advances in the area of Services Science, called Smart Service Systems. VSA is an interdisciplinary theory applied to observation of complex phenomena, based on systems theory, focusing on the relationships between socio-economic entities, searching viable conditions for interaction between them. Smart Service Systems is a proposal for advances in Information Technology and Communication Technology (ICT) aimed at a planet more intelligent. The main result is that the VSA provides valuable information for design and management systems for intelligent services, especially regarding harmonization, governance systems and processes of co-creation successful.

Bardhan et al. (2010), propose a framework to evaluate the the main research lines in SSME, with a multidisciplinary analysis , including IS, computer science, economics, finance, marketing, operations management and supply chain management. As a result of the analysis the authors obtain comprehensive coverage, with interpretation of the main issues raised; initial theoretical perspectives for further research; applications to better understand the innovation-oriented services, and service science as a key area for new research in IS.

The work of Ostrom et al. (2010), which was attended by academics working in related disciplines and institutions around the globe, and with the cooperation of executives connected to at least 1000 small, medium and large companies, helped to arrange a list of the ten largest research priorities for the development of Science Service, grouped in three areas of business:

- Strategic Priorities:
  - Foster the dissemination and growth of services.
  - Improving the quality of life through services.
  - Create and maintain a culture of service.
- Development Priorities:
  - Stimulate innovation in services.
  - Improving the design of services.
  - Improve the service networks and value chains.
- Execution Priorities:
  - To make effective the brands associated with the sale of services.
  - Improving the service experience through joint creation with the customer.
  - Measure and improve the service value.
- Permeating all priorities, using technology to leverage the service.

4. THE NEW INFORMATION SYSTEM DESIGN

The product-oriented approach, emphasizing the production of physical goods, very suitable for the manufacture, has been replaced by a service-oriented approach. At the same time, the mode of production and systems integration are undergoing changes, going from a product-oriented model to a model-driven and process-based IS. The growth of service-oriented innovations offer opportunities for new research in the area of IS are made regarding behavioral issues, economic, technical and organizational (Bardhan et al., 2010).

The services have a higher degree of intangibility and variability, implying that its production is more complex and greater exposure to risks of customers, because when the customer uses the service compares with the expected quality to perceived quality and can fit it in varying levels of acceptance. Service systems enable the production of services and are characterized by dynamic configuration of people, technologies, organizations, and also shared information between these elements, which allows vendors to create and deliver value to its customers. IS that support services system should support these requirements.

The new IS must meet more sophisticated requirements that allow the integration of the various elements that compose the company, becoming an integrator element, with much more elaborate functions, requiring a perfect framing to the needs of service systems, collecting information on various parts of BP, always with a customer focus, and delivering the information in the right places and the people who need the information.

In the development of IS, the purpose of requirements engineering is the validation of requirements by stakeholders, being a long activity involving a heterogeneous group of people seeking problems, omissions and ambiguities in the requirements document and without a reference document to be used as base on the requirements validation .

The use of requirements management method (proposed in Figure 2) based on BP as a reference document for requirements engineering can provide a base to substantiate the validation and acceptance requirements, which together with the adoption of a formal modeling language, accurate and standardized, such as SysML, may foster the elimination of possible ambiguities and inaccuracies in the models, offering the process of developing the IS possibility of matching between BP and the system requirements, and also a reference to the formal design steps. The use of formal language always creates a barrier to the implementation of these methods on the market because of the difficulty of using the formalism for both requirement and BP team.

The electric power sector has remained practically unchanged for almost one hundred years, even considering the advances in telecommunications, computing and electronics, in this period. The actual electric grid has not keeping pace with modern challenges, featuring several systemic problems: security threats to energy suppliers, including cyber attacks; limited sources of alternative energy, lack of solutions for the conservative use of energy; failure to ensure a constant supply of energy; and lack of management of the distribution network. Thus, one of the greatest current technological challenges is to transform the current electricity network in a dynamic, flexible and adaptable.

A recent movement called Smart Grid, has been promoting upgrades in the electrical sector, through the merger of several technological advances, both in developed and developing countries, including Brazil.

The European Technology Platform SmartGrids (www.smartgrids.eu) defines Smart Grid as “electricity networks that can intelligently integrate the behaviour and actions of all users connected to it - generators, consumers and those that do both – in order to efficiently deliver sustainable, economic and secure electricity supplies”.

The Smart Grid will allow consumers of electricity may have a better control their consumption, integrating the new generation of appliances, including hybrid vehicles, and integrating different sources of power generation in order to increase efficiency and better use of renewables.

According to NIST (2010) Smart Grid must provide applications and services for:

- Improve the reliability and power quality.
- Optimize the use of facilities and avoid the construction of plants for treatment of peak consumption.
- Improve the capacity and efficiency of existing power grids.
- Improve resistance to breakage.
- Enable predictive maintenance and automated responses to system disturbances.
- To facilitate the expansion in the use of renewable energy sources
- Accommodate the distributed energy sources.
- Automate the maintenance and operation.
- Reduce emissions of greenhouse gases, allowing electric vehicles and new energy sources.
- Reduce oil consumption, reducing the need for inefficient production during peak periods.
- Present new opportunities to improve network security.
- Enabling the transition to the coupling of electric vehicles and new options for energy storage.
- Increasing consumer choice.

In this context, Smart Grid is a service system, where we have many "clients" being met by the system as consumer units that require the availability of energy prices ever lower, utilities that must meet the wishes of shareholders by providing new services with differential competitive and the government interested in regulating the use of energy planning and energy balance.

5. CONCLUSION

In this work was presented the current approach to IS design considering their problems, failures and the possibility of not meeting the requirements, or not getting the proper matching between requirements and BP. An alternative wre shown were it was used of a formal modeling using SysML for requirements and BP mathinc, based on the state of SSME art, which resulting artifact were called SIS.

SIS design must meet the new directions of SSME, where the focus on service requires great flexibility and dynamism in BP, resulting in requirements for complex systems that integrate a solution of SIS. Thus, the SIS development process requires an even stronger formalism and a method which facilitates the matching between BP and requirements.

Considering the objective of this work to obtain a connection between IS and service-oriented approach, it was concluded that there is a need for a broader design process based on formal theories of systems, applied to a case study based on a design "alive", ie, a project still in progress which allows the application matching between BP and requirements.

6. REFERENCES

Aalst, W. 1998. “The Aplication of petri nets to workflow management”, The Journal of Circuits, Systems and Computers, v. 8, n. 1, p. 1-53.

Aalst, W.; Barros, A.; Hofstede, A.; Kiepuszewski, B. 2000. “Advanced Workflow Patterns”, Lecture Notes in Computer Science, vol. 1901.

Aalst, W. ; Hee, V. 2002. “Workflow management: models, methods and systems”, Cambridge: MIT Press, 2002.

Aalst, W.; Hofstede, A.; Kiepuszewski, B.; Barros, A. 2003. “Workflow Patterns”, Distributed and Parallel Databases, 14(3), pages 5-51, July.

Adam, N.; Atluri, V.; Huang, W. 1998. “Modeling and Analisys of Workflows Using Petri Nets”, Journal of Intelligent Information Systems, 10, 131–158.

Akhavan, P.; Jafari, M.; Ali-Ahmadi, A. 2006. "Exploring the interdependency between reengineering and information technology by developing a conceptual model", Business Process Management Journal, Vol. 12 No. 4.

Andersson, B.; Bider, I.; Johannesson, P.; Perjons, E. 2005. "Towards a formal definition of goal-oriented business process patterns", Business Process Management Journal, Vol. 11 No. 6.

Baker, B. 2001. "Business Modeling with UML: The Light at the End of the Tunnel", Rational University.

Balmelli, L. 2006. "An Overview of the Systems Modeling Language for Products and Systems Development", IBM Technical Report.

Bardhan, I. ; Demirkan, H.; Kannan, P.; Kauffman, R.; Sougstad, R., 2010. “An Interdisciplinary Perspective on IT Services Management and Service Science”. Journal of Management Information Systems, v. 26, n. 4, p. 13-64.

Barile, S.; Polese, F. 2010. “Smart Service Systems and Viable Service Systems: Applying Systems Theory to Service Science”. Service Science Journal, v. 2, n. 1/2, p. 21 - 40.

Barros, O. 2007. "Business process patterns and frameworks", Business Process Management Journal, Vol. 13 No. 1.

Beer, S. 1984. “The Viable System Model: Its Provenance, Development, Methodology and Pathology”. The Journal of the Operational Research Society, v. 35, n. 1, p. 7-25.
Bertalanffy, L. 1950. “An outline of general system theory”. The British Journal for the Philosophy of science, v. 1, n. 2, p. 134. Br Soc Philosophy Sci.
Bitner, M.; Brown, S. 2008. “The service imperative”. Business Horizons, v. 51, n. 1, p. 39-46.
Boulding, K. 1956. “General systems theory-the skeleton of science”. Management science, v. 6, n. 1, p. 127-139.
Cambridge 2007. "Succeeding through Service Innovation: a service perspective for education, research, business and government", Cambridge Service Science, Management and Engineering Symposium, 14-15 July. 29 Jun 2011. <http://www.ifm.eng.cam.ac.uk/ssme/default.html>
Carr, J. 2000. "Requirements engineering and management: the key to designing quality complex systems", The TQM Magazine, Volume 12 . Number 6.
Chesbrough, H.; Spohrer, J. 2006. “A research manifesto for services science”, Communications of the ACM, 49(7), July, 35-40.
Davenport, T. 1993. " Process innovation : reengineering work through information technology ", Boston: Harvard Business School Press.
Desel, J., Reisig, W. 1998. “PlaceTransition Nets”, Lectures Notes in Computer Science, vol. 1491.
Finkelstein, A.; Kramer, J.; Nuseibeh, B.; Finkelstein, L. & Goedicke, M. 1992. “Viewpoints: a framework for integrating multiple perspectives in system development”, International Journal of Software Engineering and Knowledge Engineering, v.2, pp31-58.
Guan, F.; Hernandez, F.; Bangalore, P.; Gray, J.; Skjellum, A.; Velusamy, V.; Liu, Y. 2006. "Grid-Flow: a Grid-enabled scientific workflow system with a Petri-net-based interface", Concurrency and Computation: Practice and Experience, v. 18, n. 10, p. 1115-1140.
IBM 2004. “Service Science: A new Academic Discipline?”. IBM Research. 05 Jan. 2011, <http://www.almaden.ibm.com/asr/resources/facsummit.pdf>
Jackowski, Z. 2003. "Business Modeling with UML: A Business Process Centred Architecture". 04 Jul. 2007 <http://www.agilealliance.org/show/1202>.
Kobryn, C.; 2004. “UML 3.0 and the future of modeling”, Software System Modeluing, no. 3, pp 4-8.
Korthaus, A. 1998. "Using UML for Business Object Based Systems Modeling", The Unified Modeling Language --- Technical Aspects and Applications, páginas 220--237. Physica-Verlag, Heidelberg, Germany.
Kotonya, G.; Sommerville, I. 1998. "Requirements Engineering: Processes and Techniques", John Wiley & Sons Ltd.
Laudon, K.; Laudon, J. 2009. Management Information Systems. 11th Edition. Prentice Hall.
Leffingwell, D.; Widrig, D. 2000. "Managing Software Requirements: a unified approach", Addison-Wesley.
Leite, J. 1996. “Viewpoints on Viewpoints”, ACM Workshop SIGSOFT, San Francisco, CA.
Li, H.; Wang, J. Yu, H.; Yang, D. 2007. “Services Science, Management, and Engineering: A Literature Review in the Perspective of Management Science”. IEEE Computer, 1438-1441.
Lovelock, C.; Wright, L. 2002. “Serviços: marketing e gestão”, São Paulo: Saraiva.
Lusch, R.; Vargo, S.; Wessels G. 2008. "Toward a conceptual foundation for service science: Contributions from service-dominant logic", IBM Systems Journal, v47, no1, pg 5-14, Jan/Mar.
Maglio, P.; Vargo, S.; Caswell, N.; Spohrer, J 2009. "The service system is the basic abstraction of the service science", Information Systems and E Business Management.
Marshall, C. 2000. "Enterprise Modeling with UML", Addison-Wesley.
Merz, M.; He, Y.; Vargo, S. 2009. "The evolving brand logic: a service-dominant logic perspective", Journal of the Academy of Marketing Science.
Mullery, G. 1979. “CORE - a method for controlled requirement specification”, ICSE '79 Proceedings of the 4th international conference on Software engineering.
Murata, T. 1989. “Petri Nets: Properties, Analysis and Applications”, Proceedings of the IEEE, no. 4, Vol. 77, April.
Ng, I.; Maull, R. 2009. “Embedding the new Discipline of Service Science”. In Demirkan, H, J.H. Spohrer and V. Krishna Eds, “The Science of Service Systems (Vol. 44)”. Springer.
NIST. 2010. Framework and Roadmap for Smart Grid Interoperability Standards. Nist Special Publication. 20 Dec 2010. <http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf>
Noran, O. 2000. "Business Modeling: UML vs. IDEF", School of Computing and Information Technology, Griffith University.
Ou-Yang, C.; Lin, Y. 2007. "BPMN-based business process model feasibility analysis: a petri net approach", International Journal of Production Research. v. 45, n. 12.
Ostrom, A.; Bitner, M.; Brown, S.; Burkhard, K.; Goul, M.; Smith-Daniels, V.; Demirkan, H.; Rabinovich, E. 2010. "Moving Forward and Making a Difference: Research Priorities for the Science of Service", Journal of Service Research, vol. 13: pp. 4 - 36.
Pressman, R. 2009. " Software Engineering: A Practitioner's Approach", 7th edition, McGraw-Hill.
Reisig, W. 1985. “Petri nets: an introduction”, volume 4 of Monographs in theoretical computer science: an EATCS series. Springer-Verlag, Berlin.

Ross, D.; Schoman, K. 1977. "Structured Analysis for Requirements Definition", IEEE Transactions on Software Engineering, vol. 3, no. 1, pp. 6-15.

Rozenberg G.; Engelfriet J. 1998. “Elementary Net Systems”, Lecture Notes in Computer Science, vol 1491, 1998.

Rukanova, B. 2005. "Business Transactions and Standards", Tese (PhD) - University of Twente.

Rukanova, B.; Slooten, K.; Stegwee, R. 2005. "Business Process Requirements, Modeling Technique, and Standard: How to identify interoperability gaps on a process level", Interopesa.

Russell, N.; van der Aalst, W.; ter Hofstede, A.; Wohed, P. 2006. "On the Suitability of UML 2.0 Activity Diagrams for Business Process Modelling", In Third Asia-Pacific Conference on Conceptual Modelling (APCCM2006), volume 53 of CRPIT, páginas 95–104.

Salimifard, K; Wright, M. 2001. “Petri net-based modeling of workflow systems: An overview”, European Journal of Operational Research 134, 664-676.

Selic, B. 2006. "Tutorial: an overview of UML 2", 28th International Conference on Software Engineering.

Shaw, D.; Holland, C.; Kawalek, P.; Snowdon, B; Warboys, B. 2007. "Elements of a business process management system: theory and practice", Business Process Management Journal, Vol. 13 No. 1.

Silva, J.; Drira, K.; Rosário, J. 2006. “A Proposal to Model Collaborative Work: the Case Study of a Research Network“, WSEAS Transaction on Information Science and Applications, vol. 3, no. 1, pg. 29-34, January.

Silva, J.; Santos, E.; Vaquero, T. 2005. “Specification and Analysis for Automated Flexible Manufacturing, in 18th. Congress of Mechanical Engineering“, Ouro Preto.

Silva, J.; Santos, E. 2004. “Applying Petri Nets to Requirements Validation”, in 11th. IFAC Symposium on Information Control Problems in Manufacturing, Salvador.

Sivaraman. E; Kamath, M. 2002. “On The Use of Petri Nets for Business Process Modeling”, Proceeding of the 11th Annual Industrial Engineering Research Conference.

Spohrer, J.; Maglio, P.; Bailey, J.; Gruhl, D. 2007. "Steps Toward a Science of Service Systems”, IEEE Computer 40, No. 1, 71-77.

Spohrer, J.; Vargo, S.; Caswell, N.; Maglio, P. 2008. "The service system is the basic abstraction of service science", Proceedings of the 41st Hawaii International Conference on System Sciences.

Stair, R. ; Reynolds, G. 2010. “Information Systems”, 9th ed., Course Technology.

Störrle, H.; Hausmann, J.; 2005. "Towards a Formal Semantics of UML 2.0 Activities", In Proceedings German Software Engineering Conference, V P-64, P 117-128.

Takemura, T.; Tamai, T. 2006. "Business Process Modeling with OCL", OCLApps2006 - MoDELS/UML Conference, Genova, Italy.

The Standish Group International 2009. “The CHAOS Report”, Dennis, MA.

Vargo, S.; Akaka, M. 2009. "Service-Dominant Logic as a Foundation for Service Science: Clarifications", Service Science Journal. 1(1), 32-41.

Vargo, S.; Lusch, R. 2006. “Service-dominant logic: what it is, what it is not, what it might be”. In S. L. Vargo & R. F. Lusch, “The service-dominant logic of marketing: dialog, debate, and directions“ (p. 43–56). Armonk: M.E. Sharpe.

Vaquero, T.; Romero, V.; Tonidandel, F.; Silva, J. 2007. “itSIMPLE 2.0: An integrated tool for designing planning domains”. In The International Conference on Automated Planning and Scheduling (ICAPS07).

Vanderperren, Y.; Dehaene,W. 2005. "SysML and Systems Engineering Applied to UML-Based SoC Design", Design Automation Conference (DAC), UML for SoC Design Workshop.

Verbbek H.; Aalst, W.; Hofstede, A. 2007. "Verifying Workflows with Cancellation Regions and OR-joins: An Approach Based on Relaxed Soundness and Invariants", The Computer Journal, 50(3), 294-314.

Zhao, G.; Perros, H. 2009. “How service science management and engineering (SSME) can evolve to an academic discipline?”. International Journal of Web Engineering, v. Volume 5, n. Issue 4.

Zhang, L. 2006. "Research on Workflow Patterns based on Petri nets", IEEE Conference on Cybernetics & Intelligent Systems (CIS) Robotics, Automation and Mechatronics (RAM), 2006, Bangkok, p. 1-6.

7. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.