Abstract
RUP is a process that emphasizes the development of very complex projects that includes a great volume of activities and artifacts implying software developers do not consider usability aspects. In that way, there is a possibility of enriching this process by taking into account the benefits of Human-Computer Interaction (HCI). The main goal of this paper is to describe a strategy that provides usability to the business applications within the Rational Unified Process (RUP).

1. Introduction

Successful and usable interactive system requires cooperation as software engineering as human computer interaction (HCI). But these disciplines consider the user interfaces from different point of view. Therefore to join the activities that take place in these two disciplines have been complex [1]. The functionality provided to the user is very difficult to reach if we do not consider the business model in the software subsystems integration process [2]. The business model defines how real people interact between them and with resources to reach a common objective. Therefore, a business model should be a human oriented one permitting the understanding of the organization domain knowledge and its business [3]. It is named how problem domain model which should be independent of solution model.

The Human Computer Interaction area is currently a field of great interest in systems that require high interaction with users [4]. There are many implemented systems that are not user centered and although they satisfy technical requirements and work correctly, they do not allow users to make use of all their capacities. The standard ISO-13407 provides some solutions to this kind of problem. These solutions must include multidisciplinary design, active users participation, distribution of functions, etc [5].

The user centered design (UCD) refers to the process that focuses on the usability during the whole project development. However, these methodologies do not guarantee usable developments [6], and although it tends to focus on the user to describe him/her and on the technology that will support the activities in order to reach objectives in a specific context, it does not take into consideration the understanding of organization macro goals and in many cases ignore great amount of useful information at the business level -using architectural terms-. The intended position of a company is expressed through the business objectives. These realize business goals to generate value for the company. This value should consider the customer value. This is defined by the user in both levels: the business and the software system. The Intended value helps us to focus on aspects of context that can guide us towards a systematic approach to HCI [7].

Business model have been not considered as an essential issue in the software development process. The first version of RUP did not contain a concrete approach in what concerns the specification of business systems [2]. The fuzzy specification of the boundaries between business system and software system has not permitted the independence between these systems models and their related activities. Therefore, it is necessary to have independence between business and system model and the effective allocation of usability information in those models like it was mentioned before. Although RUP is not a process that considers the usability as a crucial point, its structure presented as templates permits its customization, and it is common to find out some integration proposals between usability and RUP [8]. This paper considers some of these proposals into the business model (business architecture); in addition, we have studied a standard congress
management system (CMS), this problem has been a common example in HCI literature where always it has been analyzed from the software perspective. The CMS offers services, such as: Register, Paper Evaluation, Paper Assignment, Paper Submission, etc. Although we have modeled completely this problem we only are going to present two views of this model in order to illustrate the manner we have included the information of usability within Business Model. Next section will describe related works, and then we will explain our proposal. A brief discussion will be depicted in the section 4, and finally some conclusions and further work will be presented.

2. Related works

Engineers dedicated to the software development have seen the necessity to have process models to apply them to the development of “their products” in the same way how it has been made in other engineering areas. As a result of these initiatives have arisen several processes models which have perfectly been documented in books and papers [9]. The Software engineering is a discipline focused mainly in the functionality, efficiency and reliability of the system in execution. The current processes of development promote a fast prototypes generation of the software systems key requirements. Just as in the case of the software engineering several authors have proposed valid process models for the design of user interface [1].

This proposal allows the developers to implement their applications under usability parameters. Some of these proposals are focused to the work of the heuristic evaluation [10], others to the contextual design [11], to the essential use cases [12], to the development of scenarios [13], etc. Granollers proposes the Model of Process of the Engineering of the Usability (MPlu+a) that has its foundations as the Software Engineering as in the HCI discipline following UCD approaches [9]. Ferre et al [14] have related the activities into a user-centred process with the common activities undertaken in a generic software development process, as has been understood from the software engineering perspective in order to make this scheme understandable by software developers. Göransson et al [15] add a new discipline to the RUP, it is called “Usability Design”, in order to make this process more user centered. Philips et al [16] incorporate two artifacts that support prototyping and modeling of interfaces, creating a bridge between them. Anderson et al [17] integrate usability techniques into RUP in a project in the medical area. They state that usability is necessary in order to improve user’s satisfaction to obtain a better position in the market. Souza et al [18] propose the adaption of the RUP towards having HCI aspects integrated into its main workflows, called RUPi. Lozano [19] propose IDEAS, it is an interface development framework. It studies the software engineering and usability gap.

In addition to related work mentioned before, currently there are RUP customizations that consider usability, such as: The user experience Plug-in that is focused basically on the websites development but it does not cover all user-centred design aspects [2]. The “Usability Engineering Road Map” [20] focuses on the customization of disciplines taking into account the effort and application of usability techniques. Although this illustrates a table with usability related content by artifact on Business Modeling, it does not show how the usability is achieved in practice. The “J2EE Developer Road Map” introduces new artifacts and activities around the users interaction specification [21].

As we can observe there are some approximations in order to include usability aspects in the software development processes, however they are not integrated in the business modeling. This research is important because it incorporates advances carried out until now in RUP.

3. Our proposal

A more effective form to organize the information of the users in the models of the RUP is proposed by using previous works. The way that is suggested is to improve the usability by capturing a more significant amount of useful information in the business model. In this paper a meta-model is presented to guide the best location of user information on the models. Some models of the case of study and the traceability between the RUP models are illustrated. The case of study is a Congress Management System. Although many researchers and developers agree on describing the RUP as a software developmental process, this one goes a step beyond of the software specification permitting the specification of the whole organization [2]. The RUP uses the
Unified Modeling Language (UML) applied for system engineering as well as business engineering [2]. HCI researchers have studied the UML applicability in HCI and they propose a good way to relate the two design spaces of HCI and software engineering by applying UML. “This could only be achieved by modifications such as: (1) extending the basic UML notations, (2) modifying existing notations, or (3) somehow linking the UML diagrams to separate external task models” [22]. The last version of RUP includes the business modeling discipline. Although it does not present a detailed business architecture, it defines some views to represent systems to be developed, such as: [2] marketing view, business process view, organization view, human resources view, domain view, geographic view and communication view.

3.1. Business Architecture

The business architecture specified by RUP has a very high abstraction level. There are three differentiated levels, the business system, the software system and the technical system. Usability Engineering and user-centred design normally begin their activities at software architecture level. However, useful user information available in business system should be captured. Butler states the users are involved in a conceptual model of three layers when they use interactive systems to support their work [23]. In the RUP, the software architecture is guided by 4+1 views. However, the same should be done in business modeling using zachman views [24]. Zachman specifies more complete and less abstract business architecture [25].

We propose to combine usability aspects in the RUP business architecture using the Zachman approach. The RUP views are: Marketing view: it provides marketing information and consumer profiles. Thus, it is useful in order to understand potential costumers and define usability and accessibility strategies. Business process view: it describes the objectives of business system. It also gives information about user activities that will be automated. Organization view: it describes the important roles and responsibilities to define the personalized interfaces. Human resource view: it contains fundamental cultural characteristics and profiles of competence and formation. It also contains essential information to define interfaces. Domain view: it describes essential concepts to the business. This is fundamental at data presentation level that will be more helpful for users. Geographic view: it describes geographic business distribution. This is essential to improve the usability and accessibility - internationalization of interfaces -. In addition, the specification of these views must be guided by the best developmental practices. Göransson et al [15] proposes 12 additional practices. However, there are only some practices that are considered new: User Focus, Active User Involvement, Usability Champion, Holistic Design and A User-centered Attitude.

3.2. Business Modeling

The business model, a human being oriented model, specifies the interactions between the people and the technology [3]. Therefore it must include usability specifications. In particular, we must design for real humans for whom it is necessary to build software systems. The goal of business modeling is to define the strategy to provide value to the clients. Value-Centred HCI is an approach oriented to the value generation [7]. Gould et al [26] suggest that developers should decide what users to involve in the process in the first opportunity they have. The artifacts describing end users must be written only after substantial, firsthand contact with users [27]. The artifacts of the business modeling must be complemented as they will be described in the next section.

3.2.1. Artifacts

Business Vision: This includes people that interact with business they can be either business workers or business actors. Business Vision captures a description of the business's working environment and stakeholder characteristics, including their expertise, technical background, responsibilities, success criteria, deliverables, etc. [20]. The business Vision artifact could be improved by the kinds of activities Gould describes[26], such as, observe users working, videotape users working, learn about work organization, and try it yourself get users to think aloud while working. In the Business Vision
Guideline is necessary to include a guide to explain a new critical success factor like is "the value". Developers don’t define value. They should talk to people about it [7]. Value can be defined using essential use case capturing the user intention and system responsibility.

**Business goals**: Business goals are mainly envisioned to obtain money. However, it is important to keep clearly defined the user’s goals. User’s goals define business design mechanisms that are realized in the business process by scenarios. Business-oriented measures such as revenue and cost-per-click, to customer satisfaction-oriented metrics like page views, conversion and bok (or exit) rate could be appropriate for measuring success [28].

**Business Use Case Model**: It describes the tasks (business use cases) that human business actors perform. It captures similarities and relationships between Business Actors, using generalization relationships [20]. Effective work allocation to business actors and workers in association their responsibilities into the current process should be achieved.

**Business Actor/Worker**: Although all they do not become users of automated systems, an understanding of their characteristics provides a better understanding of the user environment [20]. Important usability relative information such as, the social and physical environment, number of user represented, computer experience and social implications should be acquired.

**Business Analysis model**: Business modeling discipline includes business analysis and design tasks. It captures the objects which are important to the domain and to users [20]. Within the RUP [27], a domain model should ensure consistence in all terminology and concepts of the user interface. Therefore the design of the user interfaces and their abstract GUI (Graphical User Interface) components must be progressive and consistent with the evolution of this domain model.

**Storyboard**: Although StoryBoard is not actually an artifact of business model [2] we propose it as a mechanism to the business level that should be captured early at any project. Business modeling artifacts specify activities and work products independent of the technology that can be realized by people. Each interaction between a person and a domain entity will require the specification of a prototype of its associate view. Each view defines the form how information is visualized or introduced, for the case of automatic activities; the views become interfaces of the software system. Storyboards are necessary to define the initial business scenarios, these become by means of prototypes of the views.

![Figure 1. Vision view of the business metamodel.](image)

Göransson et al [15] propose a new "Usability Design" discipline. This discipline is located in RUP under the requirements discipline. We propose to include in the business modeling some activities of this discipline, for example: to incorporate activities from of the workflow detail, such as "Conduct User Studies" [15]. This one focuses on the understanding of potential system users in order to create a profile of each business actor, even though there is no guarantee that these actors will transcend to the software system level. Other workflows details that are carried out in business level are: Perform Competitor Analysis, Usability Design and Develop User Assistance.

### 3.3. Arranging the Business Model

The relative information to the usability that is captured in the business model must be available in the design of the software system. In the software engineering it is so necessary to raise the requirements as to control the traceability. Our proposal is to use a metamodel not only to locate the requirements but to control the traceability in a more effective manner.

The metamodel proposed is based on Eriksson et al [29] and RUP metamodel [2]- See figure 1.
Examples of requirements are: Situations, problems, goals, business process, rules, etc. The metamodel describes how the goals are obtained from the problem domain. The problems are the difference between the real situations and states wished within the system, this is done by the stakeholders of the business. An example of this approach is presented by Remus et al [28] by means of 3 scenarios of use on “Shopzilla” which is a comparison-shopping Web site. The metamodel illustrates relations such as process-activity diagram, actor-artifact, and activity-technology support. We are presenting two figures -figure 2 and figure 4- where the arrows indicate the elements of the model associated with the relations mentioned before. They all contain user's information that must be captured with emphasis on the usability.

Figure 2. Business process “Paper Evaluation” of the case of study.

Each business process is associated to an activity diagram which tries to group the subset of responsibilities -activities- associated to it. Each actor that participates in the process is the owner of a group of activities -its responsibilities - instead of the activity diagram. This subset provides a definition of the context of use of possible software functionalities within an activity diagram -See figure 4-. Each business process is realized by a business use case realization -See figure 2-. The realization specifies what business domain objects intervene in the execution of each business process. The domain object life cycle is modeled by a state machine. The state machine defines actions which could be related with actor interactions, such as: create, destroy or change state actions. These actions may need software interfaces. In addition, the state machine make possible the whole business processes coordination and it is a control mechanism of their internal interaction. The actor is an abstract concept and is used for the specification of attributes of the people who is in charge of the different interactions that will be carried out with each one of the domain objects. If the domain object is a physical object then the actor is a business worker, whereas if is a logical one the actor is a user of the system. The functionalities that are derived from activities of an actor acquire its namespace, therefore it is not necessary in the future to add the actor’s name to each one of actor's interfaces on the model as it happens with some proposals, such as: IDEAS [19].

Figure 3. Activity view of business metamodel.

The functionalities should be represented using prototypes to capture scenarios and usability information. If either a business actor or a business worker requires technology support in order to execute his/her activities then he/she could be a software system actor. We should not forget that technology is not exclusively represented by the software system to be developed. It can be represented by any type of technological device. As a result, it is necessary to document this aspect very well -See figure 3-. At this point, functionality of software system is defined. It will require a set of interfaces for each user. It is important to remember that business workers or actors' realizations provide a very well-defined context for the specification of interaction between users and software. Butler [23] states
that to improve the usability, the system's functionality should correspond clearly with the user’s task requirements within a coherent traceability. The allocation of functions mentioned before is a key aspect when we design the functionality. Another view of metamodel is illustrated in figure 3. This provides a best detailed guideline for the specification of the support of the technology to the activities of the business. The specification of relationships activity-artifact, activity-step, step-tool, tool-tool mentor and actor-artifact in the metamodel captures the initial requirements of the software system. The business analysis model provides important information for the specification of software system requirements and the definitions of essential concepts in order to analyze it.

This is a vital point concerning the specification of interfaces according to the profiles and necessities of users. This aspect is also important in order to define the scenarios and dialogs of the system. Figure 4 illustrates a realization of the PC_Chair into the context of the business process “Paper Evaluation” for the activity “Select Paper”. Each activity in the whole system could generate diverse functionalities - use cases, see Figure 4-. Each one is possibly generated in a different context. Consequently, the functionalities related to each actor are naturally different. However, shared functionalities can exist. This factor is very useful in order to define the interfaces of each actor. This traceability is achieved by identifying what the automation points are inside the activity diagrams. The activity "Select Paper" contains seven steps from which six are automated. They are represented into parentheses -See Figure 4-. Thus, we define the software system use cases associated to each system actor and its orchestration is defined in the activity diagrams.

4. Discussion

It is necessary to clarify that the Congress Management System is not only a software system -See Figure 5, right- it is a business system –See Figure 5, left-. A problem arises because the models are the same. Consequently, many questions rise concerning this aspect: Could the same diagrams (Figure 5) represent business and software? Do more people intervene in this system? Do available software applications that offer these services exist? The answer of the first question is that same diagram could not represent both models because the abstraction level should be decreased from business to software. However, similar use cases models to this have been proposed in literature. Currently, there are companies that offer these services, such as: Zakon group its software is OpenConf. However, people who have had experience in this area will feel that it is natural to have other actors involved in the process. Those actors are not taken into...
account in previous analyses. As a result, those programs are not related to research group characteristics. For example, reviewers and other actors share their passwords with auxiliary technical personnel because these users were not taken into account at the beginning. If we analyze this example from the software perspective many questions should be answered: What would happen to the congress if the software system fails for a certain period of time? Would it be canceled? Or would alternative tools be used to carry out the congress? Would the software system be enough to carry out the processes with inexperienced authors and reviewers? Would the system documentation manual include software and business aspects in the same text? Those entire questions are important because they are a business subject instead of a software subject. When businesses are modeled, it is necessary to think about their functioning without technology. The fact of not taking this aspect into account can cause failures in applications that could not work appropriately. This deficiency is reflected in the lack of understanding by employees in what concerns system functioning when a series of utilities with no clear objectives is offered. As a result, there is lack of knowledge about responsibilities concerning the development of their activities. A problem appears when the users must exchange between their conceptual models about the activities that they make and mapping with the user interfaces that support them which are modeled with the conceptual models of the programmers. The problem is increased when the users must make this periodically to obtain a personal goal or one of the organization [23]. Solutions on usability are made by providing alternative business models for HCI, such as, risk management, design management, new product development, standards and quality assurance, and service agreements [30]. Cockton [7] states that "Quality in use and fit to context matter because they can both degrade, or even destroy, the customers intended value". He proposes to include usability goals of customer in the business. Although those kinds of solutions improve business productivity we suggest including those in the development process himself to accept and maintain a focus on value throughout their application. We have conducted some experimentation but until now we do not have analyzed the results. However, we have some comments based on the observations on the experiments: this strategy has been used in students’ final projects with successful improving the communication between users, developers and the project manager. The mentioned strategy focuses on maintaining models that are more structured and rich concerning users’ knowledge and their interaction in clear and defined contexts that make both developers and users understand the system to be developed. Workers and users or business actors will have the perception that the complete system and the applications that support it, seen from the process perspective, are useful. The use of meta-model that guides the models development, their organization, and their documentation is very useful mainly when so complex processes, as RUP, are used.

5. Conclusions and further work

Nowadays there exist many and even better tools used to define user centered projects. This aspect is considered a benefit for the development team in order to reach objectives in what concerns quality, more specifically from the usability perspective. Comparing and personalizing methodologies are time-consuming tasks due to the amount of artifacts and activities that they provide. Consequently, it is necessary to consider different perspectives in order to cover a significant amount of these processes. The mentioned customizations are very useful for the specification of human-computer interaction. RUP reinforces the developmental process in order to transform it in a more effective tool in what concerns the product quality. One advantage of RUP is that it permits customizations by using the meta-model. The models of dialog interaction are very interesting when interfaces evolution of different application services have to be modeled, mainly when the interface design is separated from the system design. A further study will be centered in the customization of RUP that integrates the technique presented in this study. This technique will update RUP activities and artifacts. This customization must focus on the traceability that has to be developed concerning the creation of models that guarantee information consistence. This information should progress in each model in order to obtain an excellent
isolation level in each architectural level. Finally, it is necessary to perform practical experiments to illustrate our ideas which should be demonstrated in practice.

6. References