Addressing the Impact of Business Process Changes on Software User Interfaces

Kênia Sousa, Hildeberto Mendonça, Jean Vanderdonckt
Université catholique de Louvain, Louvain School of Management (LSM), Information Systems Unit (ISYS)
Louvain-La-Neuve, Belgium
kenia.sousa@student.uclouvain.be, {hildeberto.mendonca, jean.vanderdonckt}@uclouvain.be

Abstract—**This paper defines an approach to maintain the work of business process analysts aligned with the work of UI designers. With this approach, models are derived from each other and aligned in order to more efficiently propagate changes when needed. In this way, each model modification could be adequately propagated in the rest of the chain. By applying this model-driven approach, the user interfaces of the information systems are abstracted from models and their interrelationships, thus directly meeting the requirements of the business processes. This approach has been validated on a case study in a large bank-insurance organization and with the implementation of a traceability tool.**

Business process modeling; model driven engineering; usability, user interface extensible markup language

I. Introduction

A revolution is occurring in the Information Technology (IT) industry through service-oriented applications. All facilities to compose functionalities, integrate different platforms and legacy systems, using the Internet potential, transformed Service-Oriented Architecture (SOA) in a success factor for organizations compromised with the time to market.

As it is possible to notice in a variety of available solutions, process automation has been the main focus of this revolution. Processes are modeled using notations and technologies compatible with SOA, which, when associated with services, might be used afterwards to be deployed in application servers and immediately become available for use. Once deployed and adopted, the business processes receive pressures from the market and even from the workers to be improved. Ralph Smith emphasizes the importance of process improvement as a way to identify and correct weaknesses and alerts that this initiative has become more critical and frequent in the twenty first century [22].

However, process automation through SOA overshadows an important conquest of the user through the years: to work with applications easy to use, to learn, and to remember in order to be more productive and comfortable when performing their activities. This problem becomes evident when the user interface usually is proposed by the own infrastructure in a generation mechanism, which does not take into account profound evolutions of the Human-Computer Interaction area, where the context of use represents a set of essential requirements to achieve the user's needs.

At the same time, business process modeling exposes how the organization visualizes its own work and how it might be different from the workers' point of view. Features in evidence on current SOA solutions reoriented the decision about the application behavior to business process analysts. Before that, system analysts naturally entered into an agreement between various stakeholders, including the end-user, to do a parallel between the business needs, the user tasks, the appropriate User Interface (UI) to accomplish these tasks and limitations of the available technologies. The work of UI designers was also minimized to the creation of templates and widgets.

Considering the scenario that values the importance of user interaction in systems, our proposal is to align business process models with UI models and explain how the link between these models are useful to identify which changes on business processes affect the user interaction and vice-versa.

During the next sections we will explore a case study where these issues are evident and propose a traceability solution to maintain the work of business process analysts aligned with the work of UI designers. In this context, the term “alignment” means to develop the user interaction according to the business process, customizing this interaction to respect the context of use; thus, a business-UI alignment. The traceability from the business process until the user interface helps business analysts in predicting the impact of process changes on the user interaction and propose changes in the processes when the user interaction is improved. This strategy is according to the traceability definition, which is “the ability to describe and follow the life of a requirement in both a forward and backward direction” [8], providing a critical support for software maintenance.

Section II presents current state of the art on the relationship of business and IT in comparison with our proposal. Section III describes the organization in which the case study took place, its methodology and issues. Sections IV and V outline our proposal and detail its application. Section VI and VII demonstrate an example for UI development based on business process using transformation rules, tracking changes and how this is automated. To finalize, section VIII presents a qualitative evaluation of the proposal and conclude with final remarks and future work.

II. Related Works

As Business-driven IT management (BDIM) refers to a new research area that focuses on making IT help the business [21], it is evident that the importance of maintaining business process-IT alignment has increased. Considering the three-layer BDIM model to derive IT-business linkage mapping functions [15], these layers are: business layer, which models business entities; business process layer, which consists of processes dependant on IT services; and IT service layer, which considers IT infrastructure, its services and behaviors of users of such services. These layers are linked to investigate cause-and-effect relationships. Similarly to our approach, we also connect business process with IT in a way to enable models to be analyzed from any layer to the next, thus, helping in the investigation of associations and impact of changes.

The Model Information Flow used at SAP [4] follows a model-driven approach that uses models and transformations between them for defining, deploying and managing SAP applications. It is noted that the models in the chain, after the business process model is customized for a project, are mostly related to IT infrastructure, for instance models to describe internal structure of the software, demand for resources by components, resource infrastructure configurations, etc.

Other recent works propose technological support for traceability between business processes and the supporting software systems, such as [2] and [24]. Many of them are founded on Organizational Engineering (OE), which focuses "on understanding the relationships and dependencies between business strategy, business processes and the supporting information systems" [25]. Analyzing strategies to align business processes and software systems, a framework was defined to do the traceability between business strategy, business processes and Information Systems (IS) using UML [27]. This framework proposes a diagram called Goal/Process/System (GPS) diagram, which represents the association between these three areas and allows identifying dependencies when changes are performed in any of these areas. The GPS diagram has a high level definition that does not demonstrate how it can express dependencies when the requested changes are very specific and within components. For instance, when users request viewing detailed payment values, where can we find the information that specifies which components are affected for a retail store case study? Is it in all of the Purchasing components; including Purchase Order and Invoice component or only in one of them?

To detect misalignment when changes are executed, Aversano applies two different techniques [2]. The first technique considers attributes of objects in order to identify misalignment. Such objects are either process activities or system components (classes and methods). The second one uses impact analysis to identify objects affected by a performed change. The attributes of objects considered in this work are "technological coverage" and "technological adequacy". These parameters advocate the point of view of the organization's managers and the business process' executors. The impact analysis considers the dependencies between the objects, identifies the types of modifications (modify activity/software component) and the propagation rule that needs to be executed to propagate a change to the connected objects.

Concerning alignment, this proposal presented traceability techniques (e.g. alignment identification and impact analysis) considering the system components, which is commonly found in other approaches. It goes further in details than the previously mentioned work, but it still lacks the support for a user perspective since it is focused on technological and managerial aspects.

Rosenkranz & Holten [20] use the cybernetic concept of variety as a measure for the complexity of systems (e.g. organizations, ISs), that is, the possible states or patterns of behavior of a system. Following a language-based approach to ISs, they argue that conceptual models can be used to communicate the possible states of a system. They used the MetaMIS approach, an ontology-driven method to bridge communication gaps between the business process and IT departments. It has been found in various researches that such communication gap is very common because one department finds it difficult to understand reports coming from the other department. To deal with this issue, we agree that conceptual models can express the shared understanding of the concepts of the organization.

Aiming at solving communication issues, one of their examples is related to appointing employees with IT background to work in the business unit. However, in our analysis, it represents a short-term solution that is strongly dependent on the knowledge of specific individuals and, consequently, a misuse of resources, as they have agreed upon in their concluding remarks. Such solutions are often applied to address increasing time pressures, common in the banking business.

Even though many contributions support the alignment of business processes and IS, it still lacks concerns on the UI. Results from investigations [7] have demonstrated that for IT to positively affect the organization, ISs must be appropriately used (e.g. use of certain functionality for its intended purpose). This corroborates with the idea that user interaction has influences on the outcome that IT can bring to organizations. But little attention has been devoted to the user interaction of developed systems.

One of the few researches on the enhancement of the UI design practice with business process modeling is the experience reported by the IBM T.J. Watson Research Center [23]. In their work, they point out the importance of model-based UI design automation in scenarios with intensive business process that give rise to systems with lots of data entering and display, and role-specific UIs. In this work, even though they do not detail how they apply it in their projects, they specify the need to align UI design with business process models. But, it does not mention whether or not UI designers receive business process models as input for their work and the difficulties they face when required to understand process models or to enhance them with human aspects, such as user tasks.

Their work was then further detailed in [24], in which they explained their approach to leverage business process models as a starting point of the UI design process. They mention that business process models are not representative enough to communicate with customers or users; therefore they use low-fidelity mock-ups to share the understanding of the process. In their approach, they argue that the information in business processes share some characteristics with the task model and use the business process model as a starting point for UI design. We do not intend to use business process models as communication instruments with customers or users; we agree that they are a good starting point for UI design and can be used by system analysts to conceive the task model.

To address model traceability, we consider some of the existing works that focus on mapping models to generate UIs. Paternò & Santoro [17] specify the relationships between task model and AUI, and between the AUI and its implementation. Vanderdonckt [26] defines a mapping model that contains the mappings between the models and elements of models. There are Model-Driven User Interface Development Environments that map concepts from different models to guide in designing UIs, such as TEALLACH [9], TERESA [14], and UI Pilot [19]. It is not in the scope of this work to detail the different techniques, nor to compare and assess them since they do not consider business process models, but we consider the contributions above as a support for model mapping and the basis for the traceability between the models.

Summarizing, many approaches are either more technical or more managerial and lack the concern with the user perspective. Recent approaches that integrate business perspective with UI design are more concerned with specific solutions and devices through the use of rapid prototyping, for instance. With a pragmatic point of view, our approach is envisioned as more flexible because of the use of conceptual models to facilitate communication and knowledge sharing between departments and interoperability of solutions when facing change requests.

In the following sections, we demonstrate how our proposal is suitable for the context of a large bank-insurance organization, which aims at aligning their business processes with IT, but is also concerned with the user experience.

III. The Organization

Our research team along with a consulting company analyzed the current situation of the organization in order to propose solutions for their main issues concerning the alignment of business processes with UIs. The organization under study is decomposed in two main sub-divisions: the insurance and the bank. The insurance is responsible for process engineering, management and implementation of products in the back-end. The bank is responsible for designing screens, usability evaluation, management and implementation of products in the front-end. We conducted interviews with business analysts from the re-engineering department; system analysts, software architects and programmers from the IS department; and UI designers and human factors experts from the UI design department.

The re-engineering team expected the other departments to follow a UI design method that maintains alignment with the business processes. Since that was not currently the case, they prepared different spreadsheets in the attempt to align and correlate business process layers with UIs elements, which can start with screens until basic objects (e.g. fields).

The IS, UI design and human factors departments found it very complicated to handle the business process documents. They argued that such documents required them to have a deep understanding of the organizational products. Besides, maintaining them required extra time to keep all the spreadsheets updated and consistent, which is not their reality since they are under-staffed to attend various demands, especially for UI design and usability evaluations. As a consequence, business processes were not aligned with UIs and it was difficult to identify the impact that changes on a business process had on UIs and vice-versa.

Aiming at decreasing such discrepancies between business and UIs, they spent a considerable amount of time on meetings to explain the business processes, to make suggestions on UI design, and on quality assurance checks. However, these meetings were not enough to make UIs be in accordance with the business processes before sending them to development.

The scarceness of product-knowledge and human resources in the UI design departments led to two possible scenarios: (i) UI designers did their work in an ad-hoc manner, considering previous designs and insights from professionals more experienced with the products; or (ii) the IS department, with more staff and product knowledge, took the lead and implemented the system before UI design. In the first case, the business process was seldom followed. In the latter, it was difficult for human factors experts to make sure the outcome of usability evaluation was put into practice.

In summary, their issues were related to: (i) Difficulties in understanding business process documents; (ii) Product knowledge owned mostly by business analysts; (iii) Lack of correlation between business process and UI design; (iv) Difficulties in doing impact analysis after changes; and (v) Difficulties to understand, to find, and to keep updated information spread in many different artifacts.

To address these issues, we proposed an approach to align their business processes with UI design, trying to maintain both an organizational point of view, to be compliant with decisions from top-management; and an end-user perspective, to develop systems that are usable.

In order to make our proposal more feasible, we first considered their current working methodology: Re-engineering team models business processes; UI design team design UIs based on templates and style guides; Human factors experts evaluate UI usability and coherence with organizational standards; Re-engineering team reviews UIs and accordance to processes; IS team implements UIs.

This methodology is executed in an iterative manner and when there are problems detected by human factors experts or when business analysts detect some needs for improvements or corrections, UIs need to be re-designed or processes need to be updated.

Our proposed methodology is explained in details in the next two sections, but it is important to point out that it does not have a great impact on their current way of working. The new approach was largely based upon the methodology they currently use. The way of working is largely the same, except that instead of working with a lot of different artifacts, often not even aligned with each other and with information duplicities, they will work with mapped conceptual models. New models with well delimited scope help them to consider the user perspective and allow them to better communicate with other departments. Therefore, it is possible to have a smooth transition from a business focus into a hybrid perspective.

IV. Unifying Business and User Perspective

Business Process Modeling Notation (BPMN) [16] is useful to represent the sequence of work and the decomposition of the organizational complexity, as well as most business process notations. However, business processes are often not representative of the flexibility necessary for user interaction, but its structure is important to decide how to make the relationship between business processes and UI explicit in order to address the following goals: (i) **traceability**: support organizations in maintaining their systems aligned with constantly evolving business processes; and (ii) **user-centered design**: provide usability in information systems used by professionals in their every day work.

Conceptual models have been valued on the exchange of knowledge and clear communication between different departments, which has been proved to be an important factor to the success of both short-term and long-term alignment [28]. Considering the importance of user interaction to make IT add value for organizations, we focus on the alignment of business processes and ISs through a UI conceptual model: the task model, which represents tasks performed by users when interacting with a system. Tasks can be decomposed in other sub-tasks and these sub-tasks can be decomposed as well, forming a tree representation, where a new level is created whenever there is the need for better description of what to do.

Figure 1 depicts, on the left, how the re-engineering department has defined their methodology to design business process models, decomposed in these six layers. Business domain represents the core and supporting processes. Processes present the generic flows per business domain. Process in a logical channel shows the process flow taking into account a specific logical channel (e.g. home banking, branches, back office). Sub-process presents a process for a logical channel in a flow. Activities represent details of a sub-process in a flow. Tasks give further detail of an activity in a flow.

In our proposal, these levels of the business process are associated with the hierarchical levels of a task model, on the middle of the figure. The association starts with the process layer because the business domain represents the overview of the process architecture. In their turn, on the right of the figure, the second level of the task model is associated with levels of screens, which are: (i) screen group, a group of closely related screens; (ii) screen, a state of the user interface when executing a task or part of a task; (iii) screen fragment, a container of related elements; and (iv) screen element, the most atomic component. This association starts in the second level of task models because the first level is often an abstract task useful for grouping, not representative for screen organization.

Both the second and third levels of task models can be represented through screen groups, since they can be decomposed in screen groups. In cases when these two levels are screen groups, the fourth level is represented by a screen, not a fragment. However, the last level of task models is always represented by screen elements; therefore, screen fragments are optional when there are many levels of screen groups. These are examples of verification rules, which can be applied to verify if system analysts associate tasks with screens in an appropriate manner, which are more detailed in section V.

Focusing on user-centered design, task models foster a strong representation for UI design because they contain decomposition in a hierarchical structure, which provides an

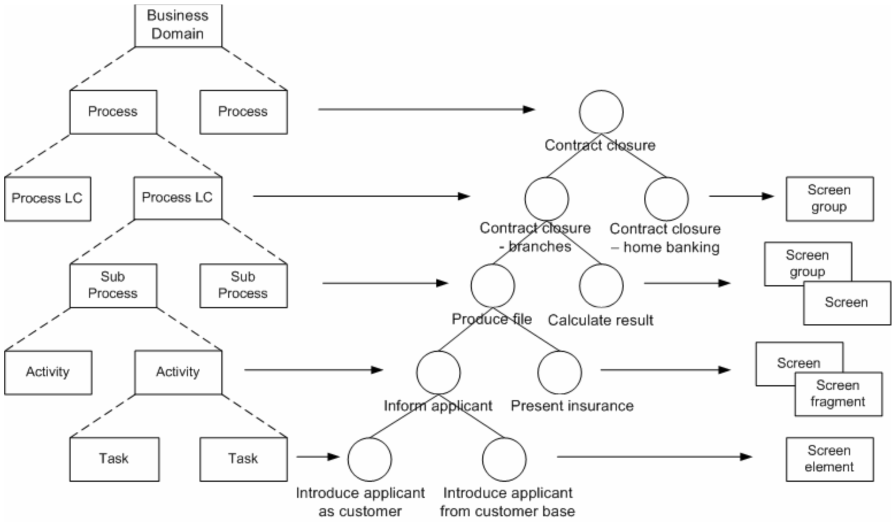


Figure 1. Association of business process with user interfaces.

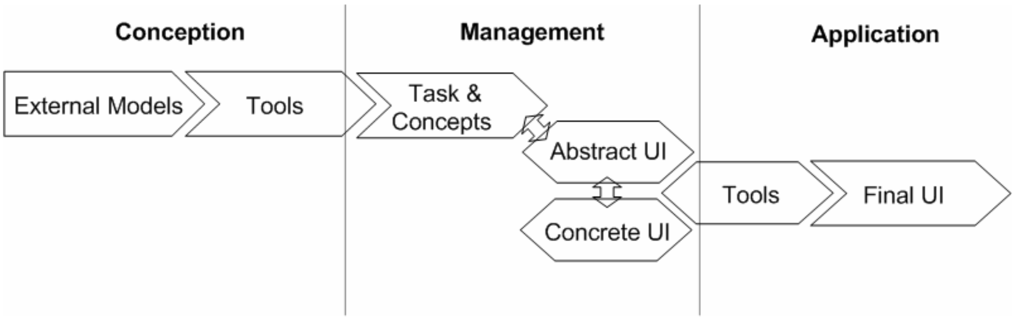


Figure 2. Phases for Model-Driven User Interface Development.

overview of the user interaction useful for decision-making on UI design (e.g. which tasks should be grouped in the same screen); and a variety of temporal ordering of tasks, which delivers the flexibility and representational power that users need while interacting with ISs.

Task models can be used as an efficient conduit between business processes and UI design because of their representativeness for UI design. Business processes are often an abstract representation of the business tasks, different from task models, which are concrete set of tasks that help UI designers visualize what could happen during user interaction. There are certain characteristics of business processes that make them a limited representation in terms of information needed for UI design: 1) the business process concepts do not consider automation in itself. The one responsible for applying the process decides what should be automated. Therefore, business processes ignore how the activity is accomplished, focusing only on what should be accomplished. 2) Business processes do not encompass all tasks that are intrinsic of user interaction, such as cancel, save temporarily, undo, etc. 3) In most cases, the business process is not detailed enough to describe individual behavior and even when it is present, the sequence of activities may not represent the user behavior, which has strong influence from the context of use. As a result, business process models can be used as requirements, not directly to UI prototyping. However, in our point of view, it is useful for the creation of task models, before UI design. These two models represent two different schools of thought; one is business-oriented, while the other is user-oriented. It is important to recognize that we need to unite both worlds to design UIs of ISs.

In other works, task models have been used to support requirement analysis, design of ISs and usability evaluation [13], and even to model user roles with different responsibilities and rights in large applications [18], but it has not yet been used as a source for traceability. Furthermore, we demonstrate how our approach can address traceability issues. We propose to correlate business process elements (e.g. sub-process, activity, task) with tasks; and tasks with UI elements (e.g. screens, objects) to identify the impact of changes in any of these models.

Changing from a business perspective to a user-centered perspective, it is necessary to focus on certain aspects of user interaction, such as naming of tasks in the task model must follow a consistent taxonomy. The relationships between tasks are very detailed in terms of understanding user behavior. Following, we present some examples of task operators used in transformations in section VI and in **Figure 4**: (i) enabling (>>) means that a task T1 has to be finished in order to initiate a task T2; (ii) deterministic choice ([]) means that once one task is initiated the other cannot be accomplished anymore; (iii) disabling means that a task T2 may interrupt T1 before the termination of T1, but T1 cannot be resumed after T2 has terminated; and (iv) concurrency means that T1 is parallel to T2.

V. MODEL-DRIVEN APPROACH FROM BUSINESS PROCESS

By adopting a model-driven approach for UI design, we take advantage of the importance of models for communicating and exchanging knowledge. Some of these approaches use User Interface Description Language (UIDL) to specify models and exchange these specifications between IT professionals; which can be exemplified by UsiXML [10], UIML [1], etc. But a UIDL needs to be grounded on a methodology in order to specify the steps to handle the models according to certain requirements, such as possessing a systematic approach and following well-founded guidelines in a repeatable manner [5]. The Cameleon Reference Framework [6] is a flexible approach towards model-driven UI design, in which models are created and mapped using UsiXML. This framework is composed of four development steps: create conceptual models (e.g. task model, data model, user model), create Abstract UI (AUI), create Concrete UI (CUI), and create Final UI (FUI). The Cameleon Reference Framework was selected because it is not coupled with specific software to be applied in an organization, which advocates some adaptation of the method. The UsiXML language will be used to exemplify our proposal since it provides the necessary support to represent models in a structured form and it supports the flexibility necessary for defining model-driven UID methods.

Adapting the Cameleon Reference Framework and using UsiXML models in the context of large organizations, we present a business–driven approach organized in three phases, as depicted in Figure 2. The models are mapped among each other in the proposed sequence, which preserves continuity and supports traceability. The main difference from the Cameleon Reference framework is the organization of the models in phases and the addition of the conception phase, in which there is an interest on business process modeling. Another aspect is the communication between the phases through tools, which facilitates its application in the industry.

A. *Business Process Modeling*

The conception phase is when business analysts model business processes that serve as requirements for UI design. It is currently a model external to UsiXML and it can be created using any available process modelling tool. These tools are able to export their models into XML format, which is appropriate to interchange information with other tools or systems that communicate with UI models.

B. *Task Modeling*

Changing the perspective from business orientation into user orientation is the focal point of the management phase. Business analysts can help system analysts to conceive the task model through meetings where business analysts explain the most crucial parts of the processes to be considered for UID.

Another option is to use a tool that uses the exported XML of business processes and transforms it into a UsiXML representation of the task model. In cases when task models are automatically generated from business processes, human factors experts should review them in order to make sure that the user perspective was taken into consideration.

The review of task models is done by creating relationships between tasks that are not considered during business analysis, but are pertaining to the user interaction, such as interruption of tasks that can be undertaken afterwards, disabling other alternative tasks when one task with higher priority is finished (e.g. form submission), optionality of tasks, etc.

C. *AUI Modeling*

Task models and other conceptual models such as the data model are used by human factors experts and UI designers to conceive the AUI. The AUI model visually specifies which screens are needed and which elements (e.g. text, field, command) are within each of them to execute a set of tasks. In the context being studied, the organization uses a standard that well resembles the AUI structure. It is comprised of broad components that are detailed until an atomic level.

Once the AUI is modeled and organized in different levels of screens, the system analyst associates each screen level with specific nodes of the task model. Considering the example used for the organization presented in Figure 1, the contract closure for the bank branches is associated to a screen group; produce file is associated to a screen, inform applicant is associated to one screen fragment of the 'produce file' screen; and introduce applicant as customer is associated to a screen element. After the association of tasks with screens, a validation engine needs to check if the association is coherent with pre-defined verification rules:

- If the root node is a 'screen group', then the node directly below it is represented by 'screen group' or 'screen';
- If the node is a 'screen group', then the node directly below it is represented by 'screen';
- If the node is a 'screen', then the node directly below it is represented by 'screen fragment';
- If the node is a 'screen' and is the second to last level, then the node directly below it is represented by 'screen element';
- If the node is a 'screen fragment', then the node directly below it is represented by 'screen element';
- If there are three or more levels below 'screen' level, then the node directly below it is 'screen fragment group'.

D. *CUI Modeling*

UI designers use AUI models, the organizational style guide, and platform restrictions to create the CUI model. Each identified screen on the AUI model will be better described in a CUI, which visually resembles the real user experience.

With an existing work on model transformations using UsiXML [11], it is possible to execute transformation of a high-level requirement into low-level analysis or design, which

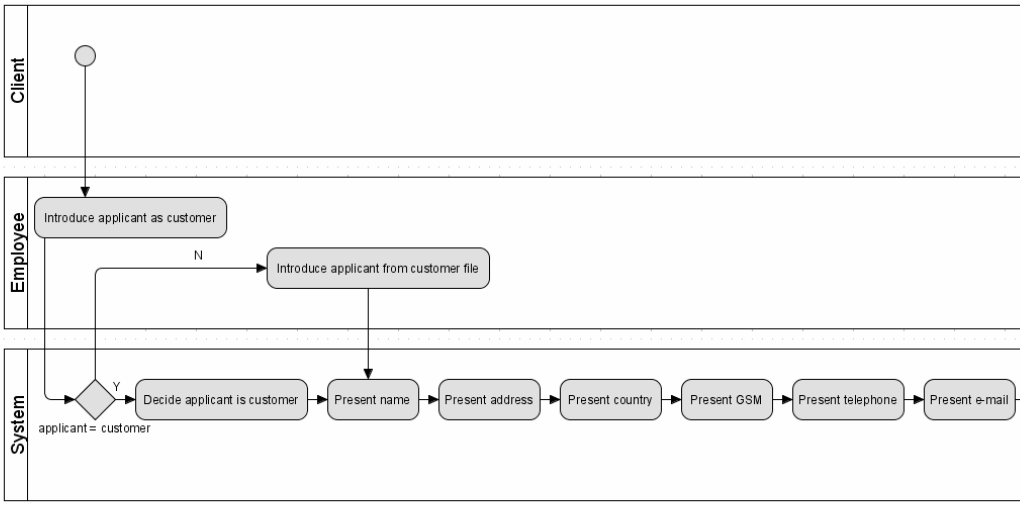


Figure 3. Business process.

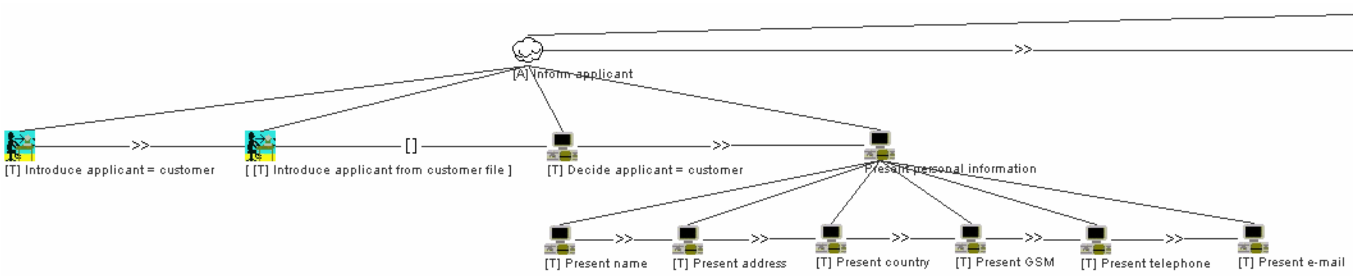


Figure 4. Task model.

is the case of transforming AUI into CUI and even CUI into code. Transformations are possible because mappings are established from any source model to any target model.

E. FUI Development

Since the conceived CUI is specified in a mark-up language, it can be processed by a tool that transforms it into a FUI. Because a CUI can represent structures for different devices, each one will be derived into a FUI, when developers consider aspects such as architecture, programming languages and infra-structure.

VI. UI Development and Tracking Changes

Model-driven development provides the traceability opportunity facilitating the creation and maintenance of relationships between models [3]. These models can be managed and traced in order to provide forward engineering to develop the system; reverse engineering to create conceptual models of existing systems; and translation of the same model for a different platform. According to the Query/Views/Transformations from the Object Management Group (OMG), traceability relationships between model elements involved in a transformation are created implicitly, hence, changes to a source model may be propagated to a target model by re-executing the transformation [12].

A. Business-Driven UI Development

We present an example of a sub-process to produce file for insurance contracts in the organization. Business analysts had meetings with system analysts and human factor experts to review some business processes and their related systems. In the review process, they noticed that decisions taken by business analysts on the order of certain activities for the 'produce file' sub-process did not consider the future user interaction with the system. As a consequence, there was some related information that was requested separately.

The 'produce file' sub-process under analysis was used to generate the complete task model. An extract of the business process is depicted in Figure 3, which shows the tasks for the activity 'inform applicant'. It specifies that the client, who contacts an employee in the bank, starts the flow. The employee informs the applicant of the insurance by either selecting the name of the applicant from a list of customers or by automatically accepting the customer from log-in information. Once the customer is identified, his or her information is retrieved and shown on the screen.

A task model was generated for this business process and **Figure 4** shows an extract of this task model for the activity 'inform applicant'. This extract is represented with interactive tasks (represented by user and computer) and system tasks (computer) that are grouped by abstract tasks (cloud). In this model, it is possible to see the application of some transformation rules from business process into task model:

- **Sequential Flow**: Sequential flows in business processes are transformed into enabling operators in task models;
- **Decisions:** Gateways in business processes are transformed into deterministic choice operators in task models.

There are also other substitutions that can be considered in this transformation, such as:

- **Cancel**: Cancel events in business processes are transformed into disabling operators in task models;
- **Parallel**: Parallel gateways in business processes are transformed into concurrency operators in task models;
- **Data Object**: Data objects associated with flows in business processes are transformed into an operator that has information passing in task models (e.g. enabling + information passing).

The grouping of the six activities related to presenting personal information of the applicant is an example of a review that is manually done by system analysts and human factors experts aiming at helping in UI design.

From the task model, the task 'inform applicant' is associated with the screen fragment 'personal data'; the task 'introduce applicant = customer' is associated to the screen element applicant (the label applicant and the combo box); the task 'introduce applicant from customer file' is associated to the screen element Look up (the button); the task 'present name' is associated to the screen element applicant name; etc. With further mapping of model components, it is possible to identify that this screen fragment is part of the screen 'insurance contract' and of the screen group 'closure of insurance contract'. The tasks that are performed by the system are not shown on the UI. The organization of the different levels of screen is shown in **Figure 5**.

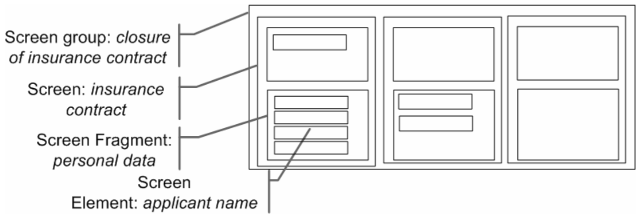

Screen group: *closure of insurance contract*
Screen: *insurance contract*
Screen Fragment: *personal data*
Screen Element: *applicant name*

Figure 5. Abstract UI.

With the AUI, it is already possible to know which UIs are impacted by changes made in the process. From this point on, UI designers are concerned with improvements on the user experience, such as platform characteristics and style guide, considered on the CUI as depicted in Figure 6.

B. Tracking Changes

This section presents how changes in business processes are reflected on the UI using a list of rules for tracking changes.

In the 'produce file' sub-process, its activity 'inform applicant' is composed of tasks related to personal information of the applicant. But, system analysts and human factor experts noticed that the subsequent activity 'present insurance' was composed of tasks also related to personal information of the applicant. Thus, leading to a situation in which related information were placed in different screens. This fact made it difficult for bank agents to interact with customers since they expected to be asked personal questions in the beginning of their meeting; then when talking about options of insurances, they did not want to start answering personal questions once again. Therefore, they decided that the business process had to be changed to address such demands from system analysts. Some tasks from the activity 'present insurance' were moved to the activity 'inform applicant'.

By comparing the updated business processes with previous task models, the change on the screen components may vary, as listed below: (i) Add screen / screen fragment – when there is a new sub-process or activity in the business process that is not yet in the task model; (ii) Delete screen / screen fragment – when there is a task in the task model, but a sub-process or activity was taken out of the business process; (iii) Add screen element – when there is a new task in the business process that is not yet in the task model; (iv) Delete screen element – when there is a task in the task model, but it was taken out of the business process; (v) Change order of screen fragments – when the order of activities in the business process and their correlated tasks in the task model were changed; (vi) Change order of screen elements – when the order of tasks in the business process and their correlated tasks in the task model were changed; and (vii) Simple review – when the changes were related to description, or rules, not on the business process or task model structure.

Based on the application of these rules and the association of tasks with screens, it was possible to detect that the screen fragment 'personal data' (for activity 'inform applicant') had to add the screen elements 'marital status' and 'birth date'. This was possible by applying rule (iii) since there were new tasks in the business process for presenting 'marital status' and 'birth date' that were not yet in the task model. It was also necessary to delete the screen elements 'marital status' and 'birth date' from the screen fragment 'insurance data' (for activity 'present insurance') by applying rule (iv) since they were in the task model, but they had been taken out of the updated business process.

VII. AUTOMATION

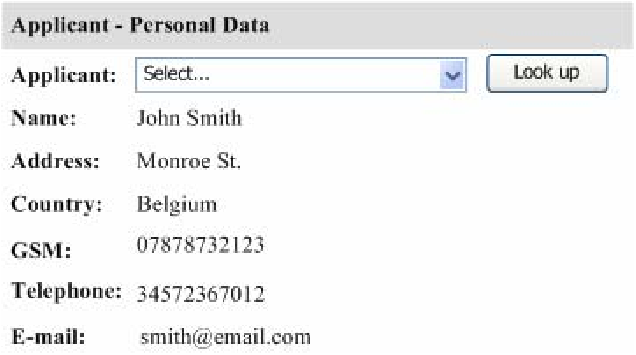
Any business process modeling tool can communicate with any external tool via a XML of processes. We have developed a tool for mapping business models, task models and screens in order to allow analyzing the impact of changes. This tool is called Usi4Biz (User Interface for Business). For the communication between these different tools to happen smoothly, we propose the following strategy.

For newly created processes, the steps are: (i) business analysts export the XML of business processes, import the XML into the traceability tool; (ii) system analysts request the generation of task model based on the XML structure; (iii) UI designers and human factors experts refine the generated task models by updating the tasks and relationships since new tasks can be created to address user interaction; (iv) UI designers and human factors experts list the screen components (screen group, screen, screen fragment, and screen element); and (iii) UI designers and human factors experts associate screen components with tasks from the task models.

The task model refinement can certainly be done with the participation of business analysts in order to allow consistency and alignment of task models with the business processes.

Whenever the business processes are updated, there is a need to synchronize the business processes with the related task models. For that, the tool needs to compare business process with task model and detect what needs to be changed on screens using the strategy explained in section VI. It is not the goal of the tool to automatically carry out changes on the models, it is rather to indicate the changes that need to be considered and analyzed by the professionals to make decisions grounded on the mappings between the models.

Figure 7 depicts this tool, in which on the left of the tool there is the business process extracted from the XML shown in a tree structure. On the middle, there is the task model generated from this business process. On the right, there are the screen components, which are useful to generate both AUI and CUI models.



Applicant - Personal Data

Applicant:	Select...
Name:	John Smith
Address:	Monroe St.
Country:	Belgium
GSM:	07878732123
Telephone:	34572367012
E-mail:	smith@email.com

Figure 6. Concrete UI.

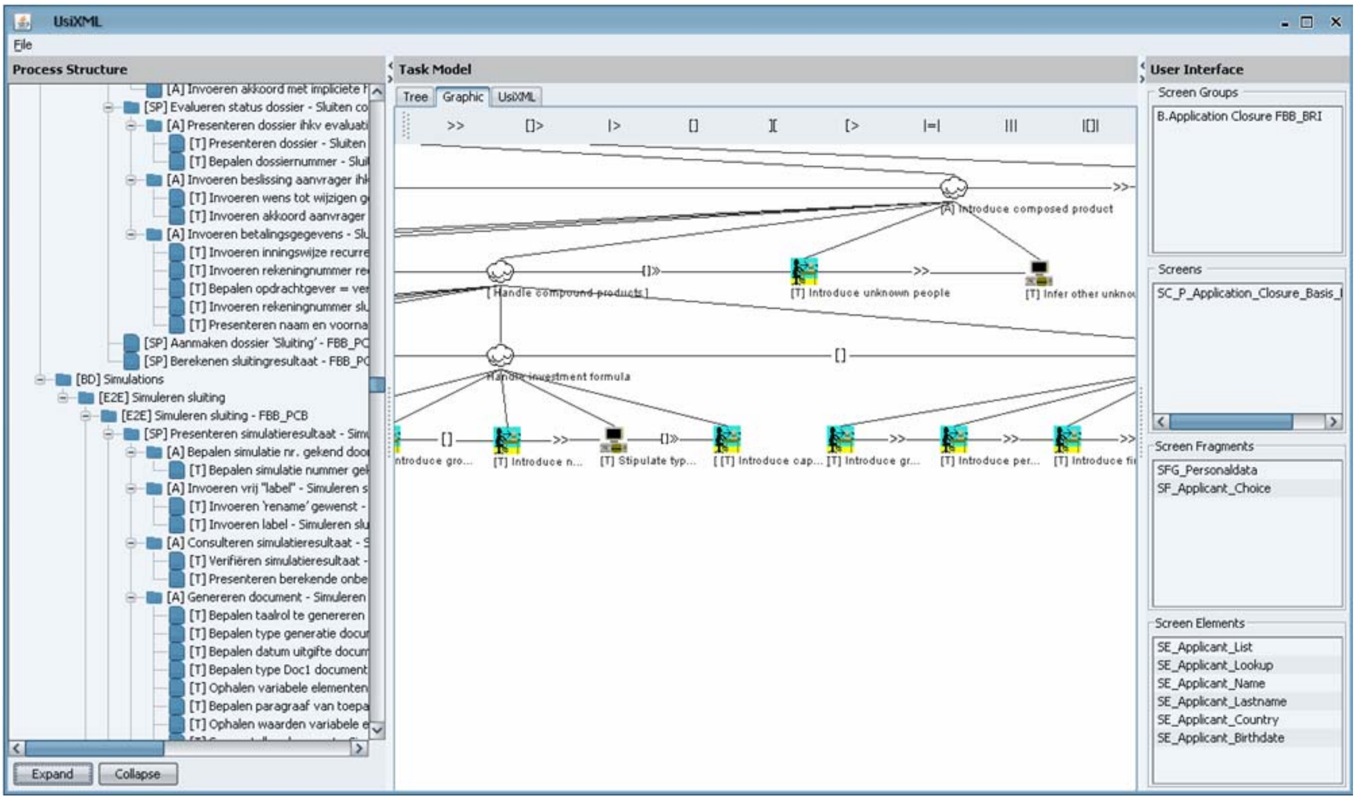

UsiXML
File
Process Structure
[A] Invoeren akkoord met impliciete f
[SP] Evalueren status dossier - Sluiten co
[A] Presenteren dossier ihkv evaluati
[T] Presenteren dossier - Sluiten
[T] Bepalen dossiernummer - Slui
[A] Invoeren beslissing aanvrager ihk
[T] Invoeren wens tot wijzigen g
[T] Invoeren akkoord aanvrager
[A] Invoeren betalingsgegevens - Slu
[T] Invoeren inningswijze recurre
[T] Invoeren rekeningnummer re
[T] Bepalen opdrachtgever = ve
[T] Invoeren rekeningnummer slu
[T] Presenteren naam en voorna
[SP] Aanmaken dossier 'Sluiting' - FBB_PC
[SP] Berekenen sluitingresultaat - FBB_PC
[BD] Simulations
[E2E] Simuleren sluiting
[E2E] Simuleren sluiting - FBB_PCB
[SP] Presenteren simulatieresultaat - Sim
[A] Bepalen simulatie nr. gekend doo
[T] Bepalen simulatie nummer gel
[A] Invoeren vrij "label" - Simuleren s
[T] Invoeren 'rename' gewenst -
[T] Invoeren label - Simuleren slu
[A] Consulteren simulatieresultaat - S
[T] Verifiëren simulatieresultaat -
[T] Presenteren berekende onbe
[A] Genereren document - Simuleren
[T] Bepalen taalrol te genereren
[T] Bepalen type generatie docu
[T] Bepalen datum uitgifte docum
[T] Bepalen type Doc1 document
[T] Ophalen variabele elementen
[T] Bepalen paragraaf van toepa
[T] Ophalen waarden variabele e
Expand
Collapse
Task Model
Tree
Graphic
UsiXML
>>
[]>
|>
[]
|[
[>
|=|
|||
|[]|
[A] Introduce composed product
[Handle compound products]
[T] Introduce unknown people
[T] Infer other unkno
Handle investment formula
[]»
>>
[]
ntroduce gro...
[T] Introduce n...
[T] Stipulate typ...
[[T] Introduce cap...
[T] Introduce gr...
[T] Introduce per...
[T] Introduce fi
User Interface
Screen Groups
B.Application Closure FBB_BRI
Screens
SC_P_Application_Closure_Basis_I
Screen Fragments
SFG_Personaldata
SF_Applicant_Choice
Screen Elements
SE_Applicant_List
SE_Applicant_Lookup
SE_Applicant_Name
SE_Applicant_Lastname
SE_Applicant_Country
SE_Applicant_Birthdate

Figure 7. Traceability Tool

VIII. Discussion

The organization was aiming for a short-term solution that did not require much investment. Therefore, we acknowledged that there is an underlying cost in our proposal that characterizes it as a long-term proposal. First, there is the cost needed to deploy our tool to manage the various models. Existing tools for business process modeling and task modeling can be used, but they lack functionalities that are present in our tool, which maps the models and tracks the changes. This tool is crucial for allowing the correct alignment between the business process and the UI models, and also to ensure traceability and synchronization. In addition to that, there is also the cost of training the professionals on task modeling and the cost with time to actually create and refine the task models.

The effort of refining tasks models is justified in contrast to the time spent on updating long spreadsheets, highly error prone, and managing artifacts spread in different tools and servers. In addition, besides serving for traceability purposes, task models are useful for user interaction analysis.

Even though, there is the time and budget restriction, we call attention for crucial factors that make this proposal feasible. Flexibility is provided for UI designers since the task model represents a structure with vast space for decision making of what represents screen groups, screens, screen fragments and screen elements. Traceability comes inherent with the adoption of model mapping and thus, makes the application of this approach more appealing for a large organization with extensive business processes.

Maintainability is simplified when synchronization is built into a tool that tracks model changes. Mapping between the different models allows an easy impact analysis when changes must occur in one of the models. The acceptance of this approach was conditioned by the ease of professionals learning the task model, which was not common in their working abilities. But once faced with the benefits brought by reducing the number of artifacts, improved communication, better consideration of user perspective and traceability; it made the concern on the learning process not a main issue for acceptance.

IX. Conclusion

This work presented a model-driven approach to link the business process with the final user interface and explored the advantages of this link. With this approach, models are derived from each other and aligned in order to more efficiently propagate changes when needed. In addition, the user experience is considered in alignment with business needs.

If applied appropriately, this approach can bring more benefits proportionally to the size of the system. The application on small systems could be problematic because it will add more complexity instead of keeping the work simpler, as preached by agile practices. On the other hand, big systems can benefit from it because of the activity volume and its respective screens, abstracted by models and their interrelationships.

This approach started to become concrete with the implementation of the tool described in section VII. Inspired by the case study and consolidated by recent researches, explored throughout this text, the tool still needs to be validated and improved in future case studies considering different business profiles.

We also emphasized the importance of the model-driven approach to address the issues discussed. Since we recognize those models, mainly UI models, as not widely known in the business environment, this motivates us to study the cognition aspects of the models applied on the business context. The next steps to continue this approach are related to the improvement of: transformation rules from business process into task model; verification rules; and rules for tracking changes through another concrete example from a prospective case study. In addition, metrics can be specified to quantitatively demonstrate the impact of mapping models and keeping track of changes.

ACKNOWLEDGMENT

Our special thanks for the organization and the consulting company for allowing us to share information of our joint project. We gratefully acknowledge the support of the Program Alban, the European Union Program of High Level Scholarships for Latin America, under scholarship number E06D103843BR.

REFERENCES

[1] Ali, M.F., Pérez-Quiñones, M.A. and Abrams M., Building Multi-Platform User Interfaces with UIML. in Multiple User Interfaces: Engineering and Application Framework. John Wiley and Sons, New York, 2003.

[2] Aversano, L. Bodhuin, T., and Tortorella, M. Assessment and impact analysis for aligning business processes and software systems. In Proc. of SAC'2005, ACM Press, 2005, pp. 1338–1343.

[3] Aizenbud-Reshef, N., Nolan, B. T., Rubin, J., and Shaham-Gafni, Y. Model traceability. IBM Systems Journal 45,3, 2006) pp. 515–526.

[4] Belrose, G., Brand, K., Edwards, N., Graupner, S. Business-driven IT for SAP – The Model Information Flow. Proceedings of the Second IEEE/IFIP International Workshop on Business-Driven IT Management (BDIM '07) (2007).

[5] Braun, C., Wortmann, F., Hafner, M., and Winter, R. Method construction - a core approach to organizational engineering. In Proc. of SAC'2005, ACM Press, new York, 2005, pp. 1295–1299.

[6] Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., and Vanderdonckt, J. A Unifying Reference Framework for Multi-Target User Interfaces. Interacting with Computers 15, 3, June 2003, pp. 289–308.

[7] Devaraj, S. and Kohli, R. Performance Impacts of Information Technology: Is Actual Usage the Missing Link? Management Science 49, 3, 2003, pp. 273–289.

[8] Gotel, O.C.Z. and Finkelstein, A.C.W. An Analysis of the Requirements Traceability Problem. In Proc. of ICRE'94, IEEE Computer Society Press, Los Alamitos, 1994, pp. 94–101.

[9] Griffiths T., Barclay, P.J., Paton, N.W., McKirdy, J., Kennedy, J.B., Gray, P.D., Cooper, R., Goble, C.A., and Silva, P.P. Teallach: A Model-based user interface development environment for object databases. Interacting with Computers 14, 1, 2001, pp. 31–68.

[10] Limbourg, Q., Vanderdonckt, J.: UsiXML: A User Interface Description Language Sup-Porting Multiple Levels of Independence. In: Matera, M., Comai, S. (eds.): Engineering Advanced Web Applications. Rinton Press, Paramus, 2004, pp. 325–338.

[11] Limbourg, Q. and Vanderdonckt, J., Multi-Path Transformational Development of User Interfaces with Graph Transformations, Chapter 8, In Human-Centered Software Engineering: Software Architectures and Model-Driven Integration", Vol. II, Springer HCI Series, Springer-Verlag, 2007.

[12] MOF 2.0 Query/Views/Transformations RFP, OMG Document ptc/2005-11-01, Object Management Group, Inc., revised on November 01, 2005.

[13] Mori, G., Paternò, F., and Santoro, C. CTTE: Support for Developing and Analyzing Task Models for Interactive System Design. IEEE Transactions on Software Engineering 28, 8, 2002, pp. 797-813

[14] Mori G., Paternò F., Santoro C. Design and development of multidevice user interfaces through multiple logical descriptions. IEEE Transactions on Software Engineering 30, 8, 2004, pp. 507–520.

[15] Moura, A., Sauve, J., Bartolini, C. Research Challenges of Business-Driven IT Management. Proceedings of the Second IEEE/IFIP International Workshop on Business-Driven IT Management (BDIM '07) (2007), pp. 19–28.

[16] OMG, Business Process Modeling Notation Specification, 1.0, February, 2006.

[17] Paternò, F., Santoro, C. A unified method for designing interactive systems adaptable to mobile and stationary platforms. Interacting with Computers 15, 3, 2003, pp. 349–366.

[18] Penichet, V. M. R., Paternò, F., Gallud, J.A., Lozano, M.D. Collaborative Social Structures and Task Modeling Integration. In Proc. of DSVIS' 2006, Springer-Verlag, Berlin, 2007, pp. 67–80.

[19] Puerta, A., Micheletti, M., and Mak, A. The UI Pilot: A Model-Based Tool to Guide Early Interface Design. In Proc. of IUI'2005, ACM Press, New York, 2005, pp. 215–222.

[20] Rosenkranz, C. and Holten, R. Combining Cybernetics and Conceptual Modeling – The Concept of Variety in Organizational Engineering. In Proc. of SAC'2007, ACM Press, New York, 2007, pp. 1228–1233.

[21] Sauve, J., Moura, A., Sampaio, M., Jornada, J., Radziuk, E. An Introductory Overview and Survey of Business-Driven IT Management. Proceedings of the First IEEE/IFIP International Workshop on Business-Driven IT Management (BDIM '06) (2006), pp. 1–10.

[22] Smith, Ralph. Business Process Management and the Balanced Scorecard. John Wiley & Sons, New Jersey, 2007.

[23] Sukaviriya, N., Kumaran, S., Nandi, P., and Heath, T. Integrate Model-driven UI with Business Transformations: Shifting Focus of Model-driven UI. In: Proc. of MDDAUI'05 (Montego Bay, Jamaica), CEUR Workshop Series, Vol. 159, 2005.

[24] Sukaviriya, N., VibhaSinha, Ramachandra, T., Mani, S., and Stolze, M. User-centered Design & Business Process Modeling: Cross Road in Rapid Prototyping Tools. In Proc. of Interact'07, Springer-Verlag, Berlin, 2007, pp. 165–178.

[25] Tribolet, J., Sousa, P. Editorial Message: Special Track on Organizational Engineering. In Proc. of SAC'2004, ACM Press, New York, 2004, pp. 1337–1338.

[26] Vanderdonckt, J. A MDA-Compliant Environment for Developing User Interfaces of Information Systems. In Proc. of CAiSE'05, Springer-Verlag, Berlin, 2005, pp. 16–31.

[27] Vasconcelos, A., Caetano, A., Neves, J., Sinogas, P., Mendes, R., and Tribolet, J. A Framework for Modeling Strategy, Business Processes and Information Systems. In Proc. of IEEE International Enterprise Distributed Object Conference, IEEE Computer Society Press, Los Alamitos, 2001.

[28] Wagner, H-T., Beimborn, D., Franke, J., and Weitzel, T. IT Business Alignment and IT Usage in Operational Processes: A Retail Banking Case. In Proc of HICSS'2006, IEEE Computer Society Press, Los Alamitos, 2006, pp. 172–173.