Deriving User Requirements from Business Process Models for Automation: A Case Study

Banu Aysolmaz, Onur Demirörs
Informatics Institute
Middle East Technical University
Ankara, Turkey
banu@aysolmaz.com, demirors@metu.edu.tr

***Abstract*— The knowledge captured in business process models is essential to identify user requirements of the process aware information system to execute the processes. Frequently, relations between the models and requirements are either not established or only partially available. This results in increased effort, broken traceability, completeness and consistency problems for user requirements. We proposed a unified business process modeling method, UPROM, to analyze and develop models for business processes and user requirements. These models are utilized to automatically generate textual user requirements. We present a multiple case study to evaluate UPROM in two e-government projects initiated for developing workflow management systems. We applied UPROM to identify most of the user requirements and prepare the technical contracts. The multiple case study shows that by applying UPROM, user requirements and business processes can be analyzed in a unified way and textual requirements can be generated automatically. The results proved to be helpful in real life settings.**

***Index Terms*— business process modeling, user requirements analysis, process aware information system, workflow management system**

I. INTRODUCTION

Business processes are among the organization's most valuable assets as they are reflections of the organizational culture [1]. Business process modeling (BPMod) is a common method to analyze, understand and define business processes. BPMod has become a tool for various practices including "pure organizational purposes" such as process improvement and business process reengineering, and many others such as workflow specification, project management, human resource planning, knowledge management, certification and process automation [2]. We name the models used to analyze, understand and communicate the processes in the business domain as descriptive business process models.

In the last decade, many organizations started projects for developing process aware information systems (PAIS) to execute their operational processes [3]. Most commonly, PAIS is developed as a workflow management (WFM) system or business process management (BPM) system that execute operational processes based on formal models [4]. These models are part of the technological side of BPM. Still, there is a need for analysis of business processes and requirements for PAIS from the user perspective [4].

Descriptive business process models are used in many ways to gather requirements in software development [4]–[7]. Descriptive business process models are the primary input to identify requirements of PAIS [8], [9]. However, frequently, software engineers do not utilize descriptive business process models systematically during process enactment. Sometimes the phase of business process analysis is even totally skipped.

Therefore we observe three major improvement areas to enhance the utilization of descriptive business process models in requirements analysis of PAIS.

1. To ensure that PAIS meets user requirements, the knowledge captured in descriptive business process models needs to be transferred to the technological BPM domain in a structured way. That is, the models shall be utilized similar to the way requirements are exploited in traditional software development approaches. Current BPMod languages and tools do not directly support an integrated approach for requirements analysis and other software development activities [10], [11].

2. Business process models serve as the main source of requirements, but still they are not sufficient to analyze and define how the automated system is supposed to behave to execute the business processes [12]. This analysis needs to be completed in the business domain and provided as an input to the technological domain, where it will be exploited to conduct detailed requirements specification and design of PAIS.

3. In many PAIS projects, organizations require a set of numbered and textual requirements statements. These statements may be needed for managing the scope in project management, conducting reviews and getting approvals of different stakeholders, and preparation of contract documents [13]. Although there is value in generating textual requirements from models, approaches that form requirements sentences from business process models are limited [12], [13].

To systematically transfer the business process knowledge captured in the business domain to the technological domain and identify the requirements of PAIS based on business processes from the user perspective, we developed a unified BPMod method, UPROM. UPROM is utilized to conduct BPMod and user requirements analysis in an integrated way and generate textual requirements sentences by using the models developed. UPROM covers notation, process, guidelines and artifact generation principles. A BPMod tool,

UPROM tool is developed as a prototype that supports modeling activities in conformance with the notation and automatically generates the user requirements document [14].

UPROM is utilized to integrate other practices and automatically generate the artifacts of functional software size estimation, process definition document, business glossary and process metrics list in addition to the user requirements document. These aspects of the method are left out of scope of this paper and can be found in other studies [15], [16].

In this paper, we describe the application of UPROM on a multiple case study for two workflow management systems in an e-government program: Company and Trademark Central Registration Systems. Initial phases of both projects included: (1) analysis of existing business processes, (2) definition of the to-be processes that will be supported by PAIS, (3) analysis of requirements and development of the technical contract document to be used in tender for PAIS. UPROM is followed as the methodology for the integrated business process and requirements analysis and the requirements generated by the tool constituted most of the technical contract.

The rest of the paper is organized as follows. Section 2 presents the UPROM. Section 3 provides the related work. Section 4 describes multiple case study design and section 5 covers the implementation. Section 6 concludes the study.

II. UPROM BACKGROUND

Associating artifacts with each other and tracing their relation throughout the phases is a common practice in software development [17]. Various methods including model driven approaches are suggested to systematize these relations. We observe that a similar approach to corroborate relations between business process models and user requirements can provide benefits such as effort efficiency, completeness, consistency and maintainability for the requirements. As an attempt to achieve these benefits, UPROM is developed to integrate analysis for business processes and user requirements.

Before going into details, we need to clarify how we use the term user requirement within this study. Requirements focus on identifying the "goals" of the system from a user perspective [18]. While a requirement expresses the desired property within the environment, a specification derived from a requirement identifies the behavior of the system to achieve the requirement [19]. In this study, our focus is on establishing user requirements from a user point of view by utilizing business process information. Though UPROM models aim to capture information on how the system will work, more analysis is required to turn them into detailed specifications.

UPROM modeling notation contains six diagram types based on a common metamodel. Core diagram of the notation is Event Driven Process Chain (EPC), which is widely used for process analysis in the business domain and known with ARIS Framework [20]. UPROM tool, a BPMod tool based on bflow* Toolbox [21], is developed to support UPROM. The tool enables modeling of six diagram types conforming to the metamodel. The tool can automatically generate user requirements document from models as defined by the method.

UPROM process consists of three main activities:

- Develop core BPMod diagrams for descriptive business process analysis,
- Develop analysis diagrams associated with BPMod diagrams to determine user requirements of each function to be automated,
- Generate user requirements document.

Each activity are briefly described in the following sections.

A. Develop Core Business Process Modeling Diagrams

To analyze functional and behavioral aspects of the business processes, Value Chain (VC), Function Tree (FT) and Event Driven Process Chain (EPC) and to represent organizational perspective, Organization Chart (OC) diagram types are defined in the notation. VC diagram shows a high level view of value added activities to obtain products and services. It can have VC or EPC type sub-diagrams referenced from value chain symbols. FT diagram is used to organize a number of related activities that are not part of a control flow. Each function can have FT or EPC sub-diagram. EPC is a common notation to model control flow based on events and functions [22]. An EPC diagram can have an EPC, FT or Function Allocation (FA) type sub-diagram for a function.

B. Develop Analysis Diagrams Associated with Business Process Modeling Diagrams

To move on to requirements analysis based on business processes, each leaf function in EPC diagrams is analyzed for automation in PAIS. If the function is to be automated, an FA diagram is created as a sub-diagram. FA diagram serves the purpose of analyzing how the function will be executed by identifying the responsibilities, related entities, operations on entities and constraints. FA diagram includes object types of: Function, Entity, Application, Organizational Elements (position, organizational unit, external person), Constraint.

During development of FA diagrams, conceptual definitions of entities are revealed. Entity Relationship (ER) diagram is modeled to grasp a full view of these entities with general, aggregation and generalization relations between them.

- The Guidelines for Organizing the Diagrams

Diagrams of all types are organized in an interrelated structure of *modeling project*. A modeling project needs to conform to a particular folder structure coherent with its hierarchy. A process map diagram of type VC, EPC or FT is placed at the highest level. The rest of the VC, EPC and FT diagrams are associated with sub-diagram relations and placed in an appropriate subfolder. Objects with the same name are assigned to be unique, attributes updated simultaneously. Logically similar types, like document and entity objects, are also retained as unique objects to establish cross-relations between objects of different diagram types. The diagrams and objects are enriched with attributes so that information required to generate the artifacts are embedded inside process models.

C. Generate User Requirements Document

Three types of requirements sentences are generated by using FA diagrams. EPCs are utilized to organize the requirements as a document. The details of requirements generation are provided in case study implementation section.

III. Related Work

As the popularity of BPM rose, the number of diverse areas and aims for which BPM is utilized also increased [2]. Number of studies focusing on structured use of business process models for such aims is relatively restricted. Requirements engineering is among the most important areas in which BPMod can be of major use. Business process models are used as a method for requirements elicitation, and "documenting and codifying" business process requirements [8], [23]. BPMod is used as a common tool to collect requirements during software development [3], [5], [7] and seen as an essential part of those activities [24], [25]. A study to assess the expressiveness of business process models for software requirements concludes that the notations are yet not expressive enough [23].

Although we can find studies that utilize UML models to analyze requirements [9], [26], studies based on business process models are limited. URN and goal modeling approaches are used for such an integration in early phases of system analysis [18], [27]. Data centric approaches exist to identify requirements by modeling executable processes [28]. A literature study concludes that although there is value in generating textual requirements from models, approaches do not exist to form sentences from business process models [12]. Another problem is the lack of synchronization between the requirements generated and the related models. Other examples integrating two fields in a structured way are the study of Mayr, Kop and Esberger for integrated requirements modeling and BPMod [7]; and problem frames approach to identify software requirements from business process models by Cox et.al. [11]. Enterprise modeling languages and frameworks such as ARIS [20] and ArchiMate [29] provide a layered approach and cover different views. However, they have many diagram types to be used for a wide variety of purposes and in a higher level.

We observe a gap to utilize business process models for a thorough analysis of user requirements between early system analysis and detailed software analysis levels. Our research group previously analyzed how business process models can be used in requirements analysis [30], [31], which utilized BPMod for requirements analysis in an unstructured way.

BPMN, used for both process analysis and execution [32], can be an alternative to EPC. BPMN, being the most complete notation, have complexity and lack of clarity problems for business users [33]. To overcome the problems usually a restricted set of constructs is used for business analysis [34]. BPMN with the restricted set and EPC have similar expressive power for descriptive modeling. It is shown that EPC models can be converted to BPMN without significant loss, and transformation is supported with the tools [35], [36].

IV. Multiple Case Study Design

Case study research is used in information systems to investigate a phenomena in its natural setting [37]. Case studies provide a deeper understanding of the topic, investigate the effect of many variables and enable intensive examination of the study in a qualitative setting [38], [39]. Thus, we decided case study research to evaluate UPROM. Multiple case study is applied to collect more data and deal with validity problems.

A. Description of the Case

Company (eCompany) and Trademark (eTrademark) Central Registration projects were launched to automate and monitor the lifecycle processes for the companies and trademarks as part of an e-government program. An online WFM system was required on which citizens can apply, officials can review and maintain applications, confirm or reject operations; information can be obtained from external services; and the system can manage state changes and inform citizens, officials and external stakeholders.

High level processes of eCompany include establishment of a new company, update of company information, approval of company applications, listing and closure of companies. eTrademark has high level processes of registration of a new trademark, regular update of a registered trademark, update of the trademark owner, update of trademark information, update of franchising for a trademark, update of trademark deputy, listing and cancellation of trademarks.

The initial phase of the projects covered the analysis of the as-is business processes, definition of the to-be processes and preparation of the technical contract document. The organization planned to subcontract the development of the WFM system by using the outputs of the initial phase. For both projects, UPROM was followed as the method to analyze business processes and user requirements, and develop the user requirements document to prepare the technical contract.

B. Research Questions

eCompany and eTrademark projects provided the opportunity to evaluate UPROM for the following research questions.

Question 1: Can we use UPROM to analyze and model business processes and user requirements in a unified way and utilize the generated user requirements as an input to PAIS development?

Question 2: Are user requirements better than the ones developed manually and separately in terms of completeness, consistency, maintainability and traceability?

C. Case Site Selection and Case Study Plan

The objective of this multiple case study was to apply UPROM and evaluate if the claimed benefits can be obtained in a real life setting. The selection criteria for the case site was identified as follows:

- The system under study shall have detailed processes to benefit from analysis by BPMod.
- There shall be a need to develop PAIS to automate those processes by means of a BPM or WFM system.
- The organization shall be willing to analyze the user requirements to develop PAIS in house or to subcontract it.

eCompany and eTrademark projects conformed to the above conditions. A case study protocol is defined including the activities planned for the conduct of the case study, preparation of project deliverables and collection of sources of evidence. 3 analysts from the contractor company, 3 analysts from the contracting organization and 2 domain experts

participated in the studies. First author of this paper worked as an analyst. Therefore this study has characteristics of an action research. To overcome threats due to direct involvement of the researcher, all of the analysts were introduced to UPROM method before the start of the project with a training and the researcher did not lead the analysis activities. Also, multiple sources of evidences was used to evaluate the results.

UPROM tool was utilized to enable rapid development of models conforming to the notation and generate the requirements automatically. Case study plan included the following activities.

- Conduct workshops with end users to understand as-is processes and their needs for improvement.
- Develop business process models (EPC, FT and OC diagrams) for to-be processes. Follow UPROM process and guidelines to analyze and model the processes.
- Organize regular workshops to present to-be process models to the end users, obtain feedbacks and update the models accordingly.
- Design a high level architecture for the system modules so that requirements can be associated with the related module during the analysis.
- Analyze user requirements on models. Develop FA diagrams for each activity to be automated and in parallel, model all entities on ER diagram.
- Generate user requirements document.
- Deliver user requirements document to the end users and obtain feedbacks. Analyze effects of requested changes and update the models. Regenerate and redeliver the document.
- Analyze all requirement for completeness and reveal additional requirements to be added to the technical contract document. Form the technical contract.
- Interview with analysts and end users to evaluate if UPROM supported the development of the technical contract in a complete and consistent way, to be maintainable and traceable.
- Collect metrics and observations on the application.
- Utilize metrics, observations and interview results to answer the research questions.

Multiple source of evidence was used. Metrics to be collected and interview questions are defined and reviewed. Direct and participant observations were other evidences [37].

V. Case Study Implementation

The case study and initial phase of the projects were conducted following the plan described above. In this section we present the conduct of the activities, the collected metrics, observations and interview findings; and discuss the results. Originally in the projects, activities were conducted in Turkish. The examples in this paper are translated to English.

A. Develop Core Business Process Modeling Diagrams

Initially, FT, EPC and OC diagrams were modeled for the projects. The folder structure of the modeling projects was organized such that each sub-diagram of type FT and EPC is placed inside a subfolder. An FT diagram was placed at the highest level as the process map. At lower levels of the hierarchy, all EPC, VC, FT and FA diagrams were referenced as sub-diagrams initiated from the process map. ER and OC diagrams were placed in the highest level folder, as they provide general information on the modeling project. The structure was validated automatically by UPROM tool to check the conformance to these aspects.

Metrics on business process models of eCompany and eTrademark projects are provided in Table 1. eCompany processes have 110 function objects in total, where eTrademark has 40. Even though the two projects are different in size, they are similar in characteristics as average values are close. EPC diagrams conform to the suggested process modeling guidelines [40] on limiting the number of control flow nodes (function, connector and process interface), connectors (And, Or, Xor), functions and control flow arcs. Although usage of multiple start and end events makes the models unsound [40]; as UPROM's aim for BPMod is to support communication and agreement with end users, usage of multiple start and end events to enhance expressiveness is reasonable [2]. By means of UPROM tool's support for unique objects, the flow across the processes by means of unique start and end events could be analyzed and observed on the models.

TABLE I. Metrics for business process models of projects

Metric	eCompany	eTrademark	Metric	eCompany	eTrademark
Deepest level	3	3	# EPC diagrams	15	6
# FT diagrams	3	3	# ER diagrams	2	1
# functions on EPCs	110	40	# functions per EPC	7,3	6,7
# FA diagrams	82	36	# FA per EPC	5,5	6
# unique control flow nodes	183	79	Average # control flow per EPC	12,2	13,2
Average connectors per EPC	4	4,7	Average # arcs per EPC	24,7	21,8

B. Develop Analysis Diagrams Associated with Business Process Modeling Diagrams

Considering the business process decomposition, a high level architectural design was conducted by decomposing each system to software modules. During requirements analysis, requirements were assigned to these modules, though this is not an implication for the development of PAIS. It rather helped to assign the requirements to high level user functionalities. The modules identified for eCompany were: eCompany Application, Approval, Listing and Closure Modules. For eTrademark, the following modules were identified: eTrademark Application, Update, Approval and Renewal Modules. In addition to these, external applications with which eCompany and eTrademark systems have interfaces with were identified. Some of those are Payments Web Service, Electronic Document Management System (EDMS), Legal Entity System and Central Civil Registration System.

In the second step, detailed user requirements analysis was initiated by evaluating each function in EPCs. Each leaf function, function with no sub-diagram assigned, was analyzed to determine if the user requires the automation of that function in PAIS. For each function to be automated, an FA diagram was created and assigned as a sub-diagram of that function.

User requirements for each function was analyzed in detail by means of FA diagram. In this way, requirements were analyzed based on business processes. Thus, models for business process and requirements analysis were developed in association with each other based on the functions.

Functional requirements explain the functionality and capability expected from a solution in terms of behaviors and operations [41]. They include functionalities for transfer, transformation, storage and retrieval of data [42]. In line with these definitions, FA diagram of UPROM was designed to analyze functional user requirements by identifying responsibilities, entities, operations on entities and constraints to conduct each function.

An example FA diagram is provided in Fig. 1. The rationale for using each object and relation type in user requirements analysis is described below.

1) Organizational Element – Function: This connection specifies the responsibility(s) involved to conduct the function in PAIS. Organizational element object types include organizational unit, group, position and external person. Connection type between an organizational element and the function can be assigned as: "*carries out, approves, supports, contributes to, must be informed on completion*".

If more than one organizational element is connected, it means that all connected organizational elements need to conduct the specified responsibility to complete the function. If any of the organizational elements can conduct the related responsibility individually on the function, and the conduct of one responsibility of the same type is enough for the function to be completed; the connections are labeled with a number. An example for this can be seen in Fig. 1. Here, a position and an external person are connected to the function and connections are labeled with a number, meaning either of them can carry out the function. If a number label wasn't used, both of them needed to be involved to carry out and complete this function.

2) Function – Entity – Application: These connections identify entities involved in the execution of the function and the systems on which the entities reside. Connection types between function and entity are designed based on CRUDL functions: "*uses, views, creates, changes, reads, deletes, lists*". Entities used and manipulated by the function and high level operations on these entities to automate the related function are analyzed using these connections.

Due to the level and purpose of user requirements analysis in UPROM, entities placed on FA diagrams are definition of concepts in the system in logical level. They cannot be directly utilized to generate a database schema, but can be utilized as input for it. Each connection type between function and entity serves the purpose of expressing data transfer, transformation, storage or retrieval requirements as explained below.

a) ***Create***: The connection expresses that an instance of the connected entity is created and stored on the connected application system during the execution of the function. When the connected application system is an external application, the connection indicates that the information on the entity is sent to the external system.

b) ***Change***: It is used when the attributes of an instance of the connected entity are updated during the execution of the function. It needs to be considered during analysis that a requirement for listing or querying the related entity exists frequently together with the change operation.

c) ***Delete***: The connection is utilized when an instance of the entity is to be deleted on the connected application system during the function.

d) ***View***: When the attributes of an entity need to be retrieved and displayed to the user during the operation of the function, entity is connected to the function using this connection type. Frequently further operations are performed on this entity or other entities during the function, after the entity is viewed by the roles.

e) ***List***: When one or more attributes of all instances of an entity, or a limited set of instances as a result of a query are to be displayed, this connection type is used. It is also used when a list of the entity needs to be populated (such as in a drop-down list) so that a selected instance can be utilized to perform operations on other entities.

f) ***Read***: This connection type is established when the attributes of an entity instance need to be obtained to be used in other operations, but not necessarily shown to the user. Read operation can never exist alone for a function, it must be combined with other operations; as the attributes are retrieved for the purpose of using in other operations.

g) ***Use***: It is only used together with list or view operations to specify the entity values required for those operations. For example, an entity connected with use operation represents that, that entity is utilized for querying other entity connected with list operation during the conduct of the function.

In the case studies, all operations that are to be enabled by the PAIS during the execution of the function were modeled on FA diagrams. This means that all operations are not necessarily performed in all executions of the function. For example, if create and change operations are applied on the same entity, the user may be creating an instance of the entity in one function execution, and only changing it in another. The analysts focused on the life cycle of entities in the whole modeling project. If an entity was created in some processes, it was considered that it possibly needs to be updated, deleted, listed and viewed in the same process or others.

In the example in Fig. 1, it is indicated that on Legal Entity System, legal entity no is used and legal entity credentials are viewed. On Central Civil Registration System, personal credentials are viewed by using id no. These are external systems and entity values obtained from them by view operations are utilized in the control of the constraints modeled in the lower part of the diagram. On eCompany Application Module, which is a module to be developed as part of eCompany PAIS, company capital info is read and stockholder info can be created and changed based on the company capital info during the function.

3) Application – Constraint: Constraints to be considered during the execution of the function, either in design or runtime, are modeled in natural language. Constraints can be on change of values for specific attributes, general limitations, state changes and time limitations. The business rules connected to the function on EPC are clues to identify the constraints. However, constraints are only applicable for the application system and restrict how the system operates. The constraints are identified as a full sentence that indicates how the system shall behave, so that the expression can be utilized directly to generate a requirements sentence.

Entities revealed during development of FA diagrams were placed on an ER diagram created for the modeling project. Relations of type aggregation, generalization and generic relationship were modeled among the entities. Considering that the analysis was performed at the user level, entities were defined conceptually. However, in the following phases this conceptual ER model will be the starting point to conduct software requirements analysis and to design the ER database model. Therefore, it helps to derive software requirements in the technological side from the user requirements in the business side.

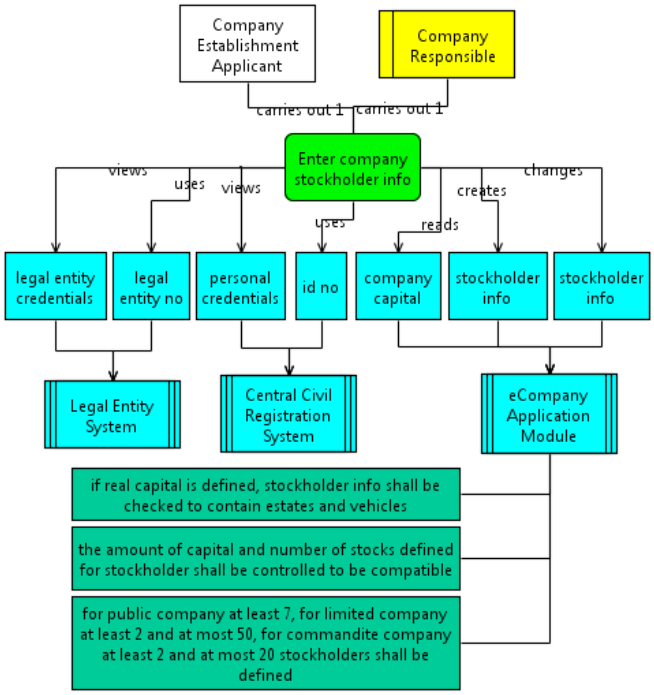

Fig. 1. An FA diagram for a function in "Enter Company Establishment Info" process

C. Generate User Requirements Document

Three types of requirements sentences are generated in UPROM from FA diagrams. Type 1 sentences specify the responsibilities to conduct the related function. Type 2 sentences express operations conducted on entities, in this way, identifies the data needs and manipulations for the execution of the function. Type 3 sentences define the constraints to be considered. Notice that functions are labeled in imperative form. The templates for generating the sentences and generated sentences for the FA diagram in Fig. 1 are provided below.

1) Type 1 Sentence:

Template: Role N1, Role N2, …, and/or Role N shall carry out; Role M1, Role M2, … and/or Role M shall approve/support/contribute to/must be informed on completion of the operation of Function1.

Fig. 1: Company Establishment Applicant or Company Responsible shall carry out the operation of entering company stockholder info.

2) Type 2 Sentence:

Template: During Function1, by using Entity N1, EntityN2, … and Entity N Entity M1, Entity M2,… and Entity M shall be viewed, Entity K1, Entity K2,… and Entity K shall be created, Entity L1, Entity L2,… and Entity L shall be updated, Entity P1, Entity P2,… and Entity P shall be read, Entity Q1, Entity Q2,… and Entity Q shall be deleted and Entity R1, Entity R2, … and Entity R shall be listed on/from Application 1.

Note: This template is adapted for the entities and operations on FA diagram. The sentence is partially formed separately for each application system on an FA diagram and connected to make a full sentence, as seen in the example.

Fig. 1: During entering company stockholder info, by using id no personal credentials shall be viewed from Central Civil Registration System; by using legal entity no legal entity credentials shall be viewed from Legal Entity System, stockholder info shall be created and updated and company capital shall be read on eCompany Application Module.

3) Type 3 Sentence:

Template: During Function1 on Application 1, constraint 1.

Fig. 1:

- During entering company stockholder info on eCompany Application Module, if real capital is defined, stockholder info shall be checked to contain estates and vehicles.
- During entering company stockholder info on eCompany Application Module, the amount of capital and number of stocks defined for stockholder shall be controlled to be compatible.
- During entering company stockholder info on eCompany Application Module, for public company at least 2 and at most 50, for limited company at least 2 and at most 50 stockholders shall be defined.

These sentences were automatically generated by UPROM tool in the projects and a user requirements document was prepared as the output. The document is organized so that each requirement sentence is placed under a heading of its related EPC diagram. The headings are leveled in conformance with the process hierarchy. Each statement is assigned a unique number. Differently in eCompany and eTrademark projects, requirement sentences were organized under the headings for applications, rather than EPC diagrams.

Analysts and domain experts examined the generated requirements for completeness. They manually added some requirements for general aspects, organization and web services of the systems. In total, 540 requirements were generated

which constituted about 90% of total requirements in the technical contract as shown in Table 2.

TABLE II. METRICS FOR USER REQUIREMENTS OF PROJECTS

Metric	eCompany	eTrademark
Total # generated requirements	363	177
Total # type 3 requirements generated	176	98
# requirements added manually	35	19
Total # of requirements	398	196
% of generated requirements	91%	90%
# software modules	6	5
# requirements per module	66	39
# requirements per EPC diagram	26,5	32,7

VI. FINDINGS AND DISCUSSION

To evaluate the results of the case study, metrics, observations and interview results were utilized. Metrics on business process models and user requirements are provided in Table 1 and Table 2. Observations were described in Section 5 where applicable, and further elaborated in this section. We also provide information on the interviews in this section.

UPROM was evaluated by analysts and end users through semi-structured interviews. Interviews were conducted with 7 different stakeholders; 2 end users, 3 analysts from the contracting organization and 2 analysts from the contractor company. The duration of interviews for each stakeholder was around 30 minutes. We asked two questions to learn the background of the interviewees on BPMod and requirements analysis. All analysts were experienced in analyzing and defining business process models both with natural language and BPMod notations. End users had experience in defining processes in natural language, and they were literate for simple notations like flow charts. End users were not directly involved in modeling in the case studies. Therefore, they were only asked questions to evaluate the outputs. A complete set of interview questions together with summary and highlights of the answers can be found in [16].

A. Evaluation of Core BPMod Aspects

We directed 5 questions to the analysts and 2 questions to the end users to evaluate descriptive business process analysis aspects of UPROM. The questions aimed to evaluate UPROM guidelines such as the establishment of a modeling project, usage of common meta-model and unique objects.

Analysts stated that folder structure of the modeling project supported them to organize, analyze and access the diagrams. Analysts spent little extra effort to reorganize the folder structure yet benefited from this practice while tracking the process hierarchy in UPROM tool.

All analysts agreed that unique object definitions were helpful for analysis and maintenance. When a modeler made changes on one instance of an object, all instances were updated. This way, it was assured that instances representing the same object have consistent and unique attribute values.

We observed that business process models provided an effective communication environment to discuss the processes with end users and find out erroneous and missing aspects. Answers of the analysts and end users supported this finding.

B. Evaluation of Integrated BPMod and User Requirements Analysis and Generated Requirements

9 questions were asked to evaluate the benefits of conducting integrated analysis activities for business processes and user requirements; and its effect on obtaining complete, consistent, traceable and maintainable user requirements. Thus, the answers obtained in this part of the interview together with observations and metrics provided the necessary evidences to answer the research questions.

Analysts from the contracting organization and end users utilized the user requirements document (not the models) to review the requirements, which showed that textual requirement statements were preferable for reviews by those stakeholders. Textual requirements were required in the projects also for the preparation of the technical contract.

All of the analysts stated that by means of UPROM notation and process, business processes steered the user requirements analysis. As part of the analysis conducted on FA diagrams, modeling of responsibilities, entities and operations on entities provided a structured representation of the user requirements. On the other hand, constraint objects were used to add “free-hand” requirements, as they contained sentences without any specific structure. One can think that there is no additional value in placing constraints on models and same benefit could be achieved by just writing them on a document. To clarify this, we asked analysts their ideas on the benefits of using constraints. They mentioned that, by means of the guidance provided by EPC and FA diagrams, they were forced to think about different aspects of the process and reveal the constraints considering the function and entities. They stated that they would not be able to uncover those constraints systematically and thus would end up with incomplete requirements if they just noted those in a document.

Considering the number of Type 3 requirements provided in Table 2, it is observed that about 50% of the requirements sentences were generated from the constraints. However, it needs to be considered that a type 2 requirement sentence is always counted as one, no matter how many operations are conducted within that FA diagram. Each operation carries an important knowledge of the system and shall be tested. But all operations are collected in one sentence for the sake of integrity and readability. If we were to count each of these operations as a single requirement, the number and percentage of Type 2 requirements would be considerably higher. Also, to enhance the readability, long Type 2 sentences can be broken down to separate requirements for each application; depending on the preference of the project stakeholders.

End users indicated that they reached a detailed understanding of the projected system by reviewing business process models and user requirements document, and no major inconsistencies were identified. All analysts stated that the analysis of user requirements guided by business process models enabled them to achieve a complete and consistent requirements set. All analysts indicated that, if they used traditional methods to analyze the requirements, the amount of effort they would spend would be similar. But they would not be able to end up with such a set if they were to write

requirements manually. Project manager, an analyst from the contracting company, stated that the technical contract defined the system so well that it would leave no room for ambiguity in bidding, and he could not achieve this in his previous projects.

All generated requirements were used and they constituted 90% of the uniquely numbered statements in the technical contract. Additional statements were added for general characteristics of the system and data interfaces between external systems.

In UPROM, the source of information for both business processes and user requirements is the model set. User requirements are analyzed for each function of the business processes. Thus, the requirements are traceable to the business processes. UPROM, as the method and with the tool that supports it, provided an environment to analyze the effects of the change and make necessary updates on the single source of models. Then, user requirements were regenerated in under a minute by means of the tool functionality. The analysts stated that although the initial requirements analysis activities required similar effort with their previous experiences, UPROM facilitated the activities of analyzing the effects of changes and updating the requirements document. Thus, in the long run, they confirmed that considerable effort savings can be reached for updates, thus maintainability is enhanced.

A few additional needs emerged for the analysis of the requirements. In Type 2 and Type 3 requirements sentences, it is stated that all of the operations conducted on the entities and constraints are valid "during" the function. However, for a limited number of functions we observed that they are applicable "on the completion" of the function. UPROM notation can be enhanced to represent operations that are valid "during" and "after" the execution of a function, and the requirements can be generated accordingly. Another point is that although multiple operations are conducted during a function, no order is specified for these operations. The operations are not always conducted in an order, but if they are, specifying the order can help analysts to better organize the needs of the function and form a better textual statement. For both improvement opportunities, we keep in mind that the requirements are analyzed from the user perspective. Thus we need to evaluate if the complexity introduced is worth the increase in expressiveness power to implement those aspects.

In addition to the business process models and technical contract, two additional needs emerged to further analyze the system in this phase of the projects: One is "Data Table" which described detailed attributes of each entity placed on ER diagrams. This information could be provided in ER diagram, but it would result in a cumbersome model due to the high number of attributes. The other is "Data Interface Table" listing the data to be provided to other systems, and the data that needs to be obtained from external systems. The data listed in this table also came from the models, but further analysis of this data was required to determine interfaces between the systems.

C. Threats to Validity

To overcome the construct validity threats, we prepared a case study plan, identified metrics to be collected and interview questions; and multiple external experts reviewed these [37].

Internal validity was our concern as we tried to make conclusions on identification of user requirements by means of applying UPROM. Implementation of multiple case study helped to enhance the causal relations. The outputs to be prepared for the case study were already required as part of real life projects. To evaluate the results from the multiple perspectives, a chain of evidence was maintained while conducting the study, and multiple sources of evidences were utilized to evaluate the results. The details of the evidences created during the case studies are provided in [16]. The background on the cases, the objectives of the related projects and how those studies utilized the case study outputs in real life setting were clearly defined.

External validity deals with the concern of generalizability of the results [37]. To overcome this threat, the conditions for applicability of UPROM were specified and case studies were described so that replication logic can be applied. UPROM tool facilitates the replication of the cases. All of these activities also enhanced the reliability of the study. The results were reviewed and conformance to the method were approved by external experts. We need to consider that the involvement of experienced analysts helped to achieve successful results, which is the case in all BPMod studies. Although we cannot differentiate the effect of experienced analysts, considering the evidences we deduce that UPROM was also found usable and beneficial for less experienced analysts and domain experts. Also, the threats due to the involvement of the researcher in the projects was dealt by training all analysts for UPROM method before the project, an analyst different than the researcher leading the project and usage of multiple sources of evidences collected from all stakeholders in addition to the researcher.

Still the results of the case study may be affected not only by the application of UPROM, but by also various conditions such as project stakeholders, previous experiences. We can never be fully sure that user requirements would be less complete and consistent if the analysis were conducted with conventional methods. However, the activities conducted to overcome the validity threats enhanced our confidence on the effect of UPROM. Moreover, we planned more case studies with different properties to obtain more data. For example, a retrospective case study was conducted to reanalyze business processes and user requirements in a completed project and compare the results with previous outcomes [16].

VII. CONCLUSION

This paper presents the conduct and results of a multiple case study for a unified BPMod method, UPROM. UPROM provides the notation, rules and guidelines to conduct integrated analysis of business processes and user requirements. As a result of UPROM application, all information is embedded in a modeling project formed with 6 interrelated diagram types. These models are then utilized to automatically generate user requirements document with textual statements. UPROM tool is utilized to develop models and generate requirements as specified by the method.

UPROM was applied in the initial phase of two WFM system projects before the bidding: eCompany and eTrademark

which are part of an e-Government program. Case study research was conducted to evaluate UPROM. User requirements document generated by UPROM were utilized to prepare the technical contract.

UPROM ensures that business process knowledge is transferred to requirements analysis phase by analyzing the functions to be automated based on business process models. In this way, the repetition of effort to gather the process knowledge is prevented, descriptive and formal operational processes remain aligned, traceability between the business and technological artifacts are kept and artifacts are easily maintained. A unified approach also brings other benefits such as providing a good communication environment between customers and developers, ensuring that process owners and software engineers are on the same terms, allowing process knowledge to be used in detailed requirements analysis phase [43], revealing relations between models and requirements, exposing information systems integration points within business process models and in these ways, improving completeness and traceability of requirements [12].

We claim that we achieved various benefits identified in the literature by identifying requirements based on business process models [12]. We summarize the benefits achieved and the answers to the research questions below.

- UPROM was applied in two projects to analyze and model processes and user requirements in an integrated way, and automatically generate user requirements document. Business process models and the technical contracts prepared by using the generated requirements document were utilized as project deliverables. Interview results revealed that stakeholders are content with the results. Considering these aspects, we conclude that UPROM can be used as an integrated method to analyze business processes and user requirements and the generated requirements can be used as input to PAIS development. To make a thorough evaluation of the adequacy of generated user requirements for the implementation of PAIS, we need to observe their usage through the development of PAIS. The development activities just started in the projects and we planned to collect data to reevaluate the results upon completion.
- Considering the interview responses, we conclude that models ensured better analysis of both business processes and requirements for these projects. Business process models and user requirements document together enabled the end users to easily understand and validate processes and user requirements. As a result, guided by business processes, analysts concluded that the level of completeness was hard to be achieved by traditional methods, or would require too much effort.
- Artifacts were developed in a more consistent way. We observed that business processes enabled identification of separate functions in the system, and prevented duplication. Reviews of analysts from the contracting organization and end users also supported the findings, as none of their findings were for overlapping and inconsistent information in the processes or requirements.
- To sum up, requirements fitted better to the criteria identified by standards [44] to be unambiguous, complete, consistent and verifiable.
- All artifacts were traceable to the models as they were generated based on the models. The effort to maintain the artifacts were decreased, as the updates were conducted and their effects were analyzed easily by means of models. The artifacts were generated automatically with almost no extra effort by means of automated generation functionality of the tool.

For future work, we plan to enhance requirements analysis method and investigate further utilization of business process knowledge for the development of PAIS. As an example, we plan to integrate analysis activities of requirements on general properties of the system and interfaces with other systems and utilize the developed models for testing. Considering the widespread usage of BPMN, we also plan to integrate BPMN as the alternative notation to EPC.

ACKNOWLEDGMENT

The projects are conducted as part of the e-Government program for the Turkish Republic of Northern Cyprus under the management of Türksat A.Ş.

REFERENCES

[1] O. Turetken and O. Demirors, “Process modeling by process owners: A decentralized approach,” *Softw. Process Improv. Pract.*, vol. 13, no. 1, pp. 75–87, 2008.

[2] J. Becker, M. Rosemann, and C. Von Uthmann, “Guidelines of Business Process Modeling,” in *Business Process Management (2000)*, 2000, pp. 30–49.

[3] M. Dumas, W. Van der Aalst, and A. ter Hofstede, *Process-aware information systems : bridging people and software through process technology*. New Jersey: John Wiley & Sons, 2005, p. 426.

[4] W. M. P. Van Der Aalst, “Process-Aware Information Systems: Lessons to Be Learned from Process Mining,” in *Transactions on Petri Nets and Other Models of Concurrency II*, K. Jensen and W. M. Aalst, Eds. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 1–26.

[5] M. Indulska, P. Green, J. Recker, and M. Rosemann, “Business Process Modeling: Perceived Benefits,” in *Conceptual Modeling - ER 2009 SE - 34*, vol. 5829, A. F. Laender, S. Castano, U. Dayal, F. Casati, and J. Oliveira, Eds. Springer Berlin Heidelberg, 2009, pp. 458–471.

[6] P. Jamshidi and C. Pahl, “Business process and software architecture model co-evolution patterns,” in *Modeling in Software Engineering (MISE), 2012 ICSE Workshop on*, 2012, pp. 91–97.

[7] H. C. Mayr, C. Kop, and D. Esberger, “Business Process Modeling and Requirements Modeling,” in *Digital Society, 2007. ICDS ’07. First International Conference on*, 2007, p. 8.

[8] J. Vom Brocke and M. Rosemann, *Handbook on business process management 1*. Springer-Verlag, 2010, p. 632.

[9] B. Berenbach, D. Paulish, J. Kazmeier, and A. Rudorfer, *Software & Systems Requirements Engineering: In Practice*, 1st ed. New York, NY, USA: McGraw-Hill, Inc., 2009.

[10] T. Specht, J. Drawehn, M. Thränert, and S. Kühne, “Modeling Cooperative Business Processes and Transformation to a Service Oriented Architecture,” 2005.

[11] K. Cox, K. T. Phalp, S. J. Bleistein, and J. M. Verner, “Deriving requirements from process models via the problem frames approach,” *Inf. Softw. Technol.*, vol. 47, no. 5, pp. 319–337, 2005.

[12] J. Nicolás and A. Toval, “On the generation of requirements specifications from software engineering models: A systematic literature review,” *Inf. Softw. Technol.*, vol. 51, no. 9, pp. 1291–1307, 2009.

[13] N. A. M. Maiden, S. Manning, S. Jones, and J. Greenwood, “Generating requirements from systems models using patterns: a case study,” *Requir. Eng.*, vol. 10, no. 4, pp. 276–288, 2005.

[14] B. Aysolmaz and O. Demirörs, “UPROM Tool : A Unified Business Process Modeling Tool for Generating Software Life Cycle Artifacts,” in *Proceedings of the CAiSE Forum*, 2014.

[15] B. Aysolmaz and O. Demirörs, “Modeling Business Processes to Generate Artifacts for Software Development : A Methodology,” in *Modeling in Software Engineering (MISE), 2014 ICSE Workshop on*, 2014.

[16] B. Aysolmaz, “Unified Business Process Modeling Methodology (UPROM) Application: Case Study Results METU/II-TR-2014-32,” Ankara, Turkey, 2014. http://expertjudgment.com/publications/METU_II_TR_2014_32.pdf

[17] “CMMI® for Development, Version 1.3 CMMI-DEV, V1.3,” 2010.

[18] A. Borgida, F. Dalpiaz, J. Horkoff, and J. Mylopoulos, “Requirements models for design- and runtime: A position paper,” in *Modeling in Software Engineering (MiSE), 2013 5th International Workshop on*, 2013, pp. 62–68.

[19] M. Jackson and P. Zave, “Deriving Specifications from Requirements: An Example,” in *Proceedings of the 17th International Conference on Software Engineering*, 1995, pp. 15–24.

[20] A.-W. W. Scheer, *Aris-Business Process Frameworks*, 2nd ed. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1998.

[21] C. Böhme, J. Hartmann, H. Kern, and S. Kühne, “bflow* Toolbox-an Open-Source Modeling Tool,” in *Business Process Management Demonstration Track 2010*, 2010, pp. 1–6.

[22] P. Fettke, P. Loos, and J. Zwicker, “Business process reference models: Survey and classification,” *Bus. Process Manag. Work.*, pp. 469–483, 2006.

[23] C. Monsalve, A. April, and A. Abran, “On the expressiveness of business process modeling notations for software requirements elicitation,” in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, 2012, pp. 3132–3137.

[24] J. Vara, M. Fortuna, J. Sánchez, C. L. Werner, and M. S. Borges, “A Requirements Engineering Approach for Data Modelling of Process-Aware Information Systems,” in *Business Information Systems SE - 12*, vol. 21, W. Abramowicz, Ed. Springer Berlin Heidelberg, 2009, pp. 133–144.

[25] A. Gross and J. Doerr, “EPC vs. UML Activity Diagram - Two Experiments Examining their Usefulness for Requirements Engineering,” in *Requirements Engineering Conference, 2009. RE ’09. 17th IEEE International*, 2009, pp. 47–56.

[26] J. Daniels and T. Bahill, “The hybrid process that combines traditional requirements and use cases,” *Syst. Eng.*, vol. 7, no. 4, pp. 303–319, 2004.

[27] A. Pourshahid, D. Amyot, L. Peyton, S. Ghanavati, P. Chen, M. Weiss, and A. J. Forster, “Toward an Integrated User Requirements Notation Framework and Tool for Business Process Management,” in *e-Technologies, 2008 International MCETECH Conference on*, 2008, pp. 3–15.

[28] C. M. Chiao, V. Kunzle, and M. Reichert, “Integrated modeling of process- and data-centric software systems with PHILharmonicFlows,” in *Communicating Business Process and Software Models Quality, Understandability, and Maintainability (CPSM), 2013 IEEE 1st International Workshop on*, 2013, pp. 1–10.

[29] The Open Group, “ArchiMate.” [Online]. Available: http://www.archimate.nl/en/home/. [Accessed: 07-Jul-2014].

[30] A. Coskuncay, B. Aysolmaz, O. Demirors, O. Bilen, and I. Dogan, “Bridging The Gap Between Business Process Modeling And Software Requirements Analysis: A Case Study,” *MCIS 2010 Proc.*, p. Paper 20, 2010.

[31] O. Demirors, Ç. Gencel, and A. Tarhan, “Utilizing business process models for requirements elicitation,” in *Euromicro Conference, 2003*, 2003, pp. 1–4.

[32] “Business Process Model and Notation (BPMN) formal/2011-01-03,” 2011.

[33] J. Recker, M. Rosemann, M. Indulska, and P. Green, “Business process modeling: a comparative analysis,” *J. Assoc. Inf. Syst.*, vol. 10, no. 4, pp. 333–363, 2009.

[34] J. Recker, “Opportunities and constraints: the current struggle with BPMN,” *Bus. Process Manag. J.*, vol. 16, no. 1, pp. 181–201, 2010.

[35] O. Levina, “Assessing Information Loss in EPC to BPMN Business Process Model Transformation,” in *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2012 IEEE 16th International*, 2012, pp. 51–55.

[36] W. Tscheschner, “Transformation from EPC to BPMN,” in *Business Process Technology*, 2006.

[37] R. K. Yin, *Case Study Research: Design and Methods, 3rd Edition (Applied Social Research Methods, Vol. 5)*. SAGE Publications, Inc, 2002, p. 200.

[38] I. Benbasat, D. K. Goldstein, and M. Mead, “The Case Research Strategy in Studies of Information Systems,” *MIS Q.*, vol. 11, no. 3, pp. 369–386, Sep. 1987.

[39] P. Runeson and M. Höst, “Guidelines for conducting and reporting case study research in software engineering,” *Empir. Softw. Eng.*, vol. 14, no. 2, pp. 131–164, 2009.

[40] J. Mendling, H. A. Reijers, and W. M. P. van der Aalst, “Seven process modeling guidelines (7PMG),” *Inf. Softw. Technol.*, vol. 52, no. 2, pp. 127–136, Feb. 2010.

[41] K. Brennan, *A guide to the Business Analysis Body of Knowledge (BABOK guide), version 2.0*, 2nd Editio. IIBA International Institute of Business Analysis, 2009.

[42] COSMIC, “The COSMIC Functional Size Measurement Method Version 4.0 Measurement Manual,” 2014.

[43] M. Niazi, K. Cox, and J. Verner, “A measurement framework for assessing the maturity of requirements engineering process,” *Softw. Qual. J.*, vol. 16, no. 2, pp. 213–235, 2008.

[44] IEEE, “IEEE Recommended Practice for Software Requirements Specications - IEEE Std 830-1998,” New York, NY, USA, 1998.