Performing Early Feasibility Studies of Software Development Projects Using Business Process Models

Ayman A. Issa, Faisal A. Abu Rub

***ABSTRACT*—A new approach to perform feasibility studies using business process models is proposed. The utilization of the improved Role Activity Diagram notations has led to the extraction of the number of candidate use cases in software systems which naturally led to software cost estimation much before the availability of detailed system requirements. This enables project managers to perform their what-if analysis to inform the applicability and profitability of the anticipated software system. The process of cancer registration in Jordan is used to demonstrate the proposed approach and showing its ease and timeliness. Finally, further work is planned to validate and evaluate the proposed approach using a number of current software projects.**

***Index Terms*—Business Process Model, Feasibility Study, Software Cost Estimation, Use Case Model.**

I. INTRODUCTION

Several estimates of the software cost are required throughout the early and late stages of the software development life cycle for different purposes [1]. The main purpose of software cost estimation at the early stages is to investigate the economic feasibility of the software project [1]. In contrast, the purpose of advanced stage software cost estimation is to build up project plans and schedules.

The inherent problem with cost estimation is that small projects can be easily estimated, but the required accuracy may not be very important. On the other hand, large projects are very difficult to estimate, but the required accuracy is greater than what is normally achieved [2]. Different factors contribute to the inaccuracy of software cost estimation, such as the difficulty of coping with imprecise requirements and changing technologies

Unfortunately, these are the main common characteristics of the early phases of software development projects and contribute to the inaccuracy of the estimates that are generated. However, the accuracy of an estimate can be enhanced throughout the software development cycle as more stable requirements become available. For instance, Boehm et al. [1] reported that the COnstructive COst MOdel II (COCOMO II) uncertainty varies from ± 400% in the early feasibility phase down to 0% in the delivery phase. This paper aims to investigate and develop software cost estimation methods that are applicable in the early stages of software development much before the availability of detailed system requirements and models.

Ayman A. Issa, Software Engineering Department, Faculty of Information Technology, Philadelphia University, Jordan, aissa@philadelphia.edu.jo

Faisal A. Abu Rub, Management Information Systems Department, Faculty of Business and Financial Sciences, University of Petra, faburub@uop.edu.jo

Business Process Modelling (BPM) can be defined as the representation of one or more of the process perspectives to understand, analyse, and/or improve automated and/or non-automated business processes. Hence, the availability of business process models in any organisation is not tied to any corresponding software system. Rather, they may exist much before the automation of the business of any organisation. However, business process modelling can be used to contribute positively in the software development process. One of the contributions was suggested by Odeh and Kamm [3], they illustrated the derivation of use cases from a process model.

On the other hand, the software cost estimation literature [4], [5] described numerous early stages based software cost estimation models, more interestingly, those that are use case model based. Karner [5] developed a use case points method that utilizes the identified actors and use cases to size the software project and consequently estimate the predicted effort and time required to deliver an operational system. In addition, Issa et al. [4] developed three use case based software estimation methods regardless of the use cases' levels of detail. These are: use-case rough estimation, use-case patterns catalogue estimation, and object points extraction using the anticipated system's use-case model. The initial investigation for the results of these new methods showed promising signs on the applicability of employing use case models for software cost estimation purposes [4[, [5].

Thus, the early availability of business process which can be used to derive use cases, and the promising results of the use case based software cost estimation models raise two research questions in which this paper aims to investigate: what is the reliability of the business process based use cases? To what extent can the generated use cases be utilised to drive subsequent software development phases, in particular, early stages software cost estimation (i.e. feasibility studies)?

The related literature of bridging the gap between business process models and system models is introduced in the next section followed by the proposed new approach to early feasibility studies of software systems in section 3. The proposed approach is demonstrated by example in section 4. Finally, the conclusion and future work are discussed in section 5.

II. BUSINESS PROCESS MODELS AND SYSTEMS MODELS

Role Activity Diagramming is the process modelling language that is adopted in this research. Role Activity Diagrams (RADs) are diagrammatic notations to represent

and model coordinated behaviour and interactions within a process.

According to Ould [6], RAD represents the roles that perform a part in a process, and their elements (activities and interactions). RAD presents the task(s) of roles in the process and how they collaborate. Because RAD adopts the role as the primary unit for analysis in process models, it is suitable for organisational contexts, since it partitions the organisational behaviour of a process into roles [7]. A role involves a sequence of activities, which are carried out together as a particular responsibility. According to Saven [8], roles can be identified as abstract notations of behaviour representing a desired behaviour within the organisation. They can include software systems, customers and suppliers. RADs provide a visual representation of the different aspects of a process, which makes them useful in supporting communication, since they are easy to read and understand. RADs can also be used to demonstrate how processes interact [8].

In RAD models, each role is represented as a separate shaded area with activities represented as black boxes within a role. Interactions are represented as horizontal lines joining roles, and a vertical line within a role represents states of the role; the state of any part in the role can be labelled by using small loop around the state line. A role can be constructed to depict concurrent or parallel behaviour, using the point-up refinement triangle. Choice, or "case-refinement" [6], is shown by a point-down triangle. The main RAD notation is shown in Figure 1.

Jacobson [9] defines a use case as "a behaviourally related sequence of transactions in a dialogue with the system". A more recent definition in the context of Rational's Unified Process (RUP) shows little real change, in which they defined a use case as [10] "a description of a set or sequence of actions, including variants, that a system performs that yields an observable result of value to a particular actor". It is stated that the general idea of use case modelling is to represent intended sequences of interactions between a system and the world outside that system represented by actors. An actor is a role that a human or non-human plays to interact with the planned system to send information, receive information, or both [10].

This indicates that there is a relationship between business process and use case models. In 2003, Odeh and Kamm [3] proposed a method to explore this relationship. Their method has led for the derivation of use case system model from a process model particularly Role Activity Diagram (RAD). Although they managed to develop use cases from transactions and states derived from RAD models, some difficulties in deriving system actors were highlighted. This was attributed to the unavailability of simple mapping of roles in process models onto actors in use case diagrams.

According to Aburub et al. [11], RAD models can be used to pinpoint where actions (activities and interactions) could be automated, either partly or fully. This suggests a real improvement to Odeh and Kamm's approach and paves the way to the key question being investigated in this research with respect to the extent to which business process based use-cases can be reliable to drive subsequent software development phases.

III. MODEL LED APPROACH TO CONDUCT EARLY FEASIBILITY STUDIES

Odeh and Kamm's approach to derive use cases from business processes is based on grouping of states and transactions into subsets. The generated subsets may vary from one analyst to another according to her/his understanding of organizational contexts. Hence, the same business process may lead to different number of use cases with different responsibilities in the different contexts. Also, Odeh and Kamm's approach pays no attention to whether the allocated activities and interactions to use cases are automated or not. This means that the derived use cases will be biased with some non-automated activities that explicitly affect their utilization in subsequent software development phases.

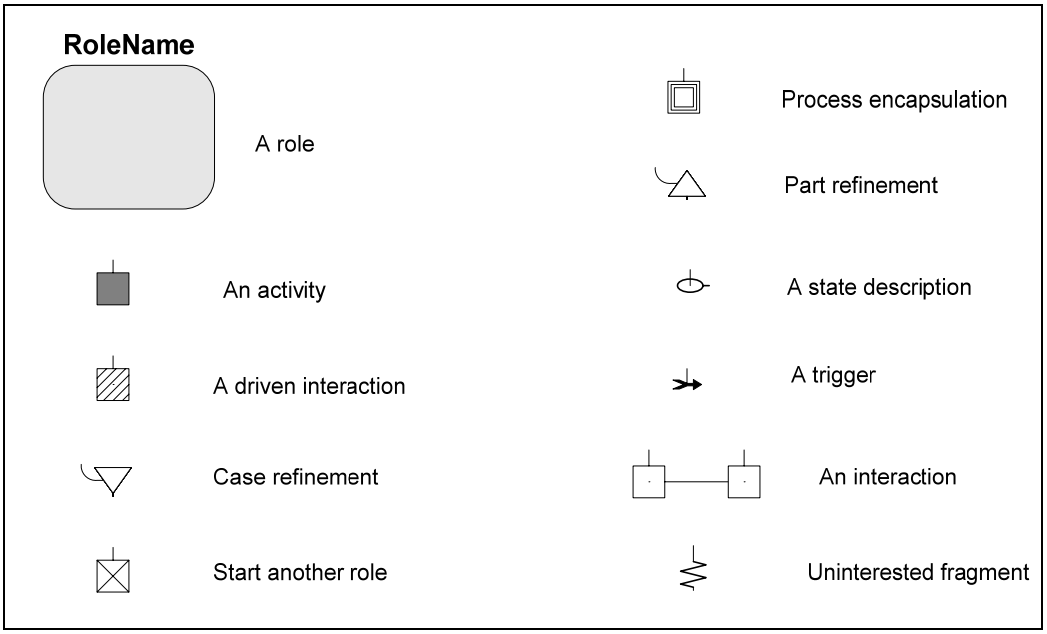

Figure 1: Main RAD Notations.

Therefore, we utilize Aburub's et al. [11] approach in role activity diagramming to derive the corresponding use cases. This will enrich the derived use cases with, to some extent, matured information about automated interactions and activities. As will be detailed in section 4, Figure 2 shows the improved RAD model of the cancer registration process in Jordan with bold boxes to represent automated activities (activities performed using a computer-based system), and regular boxes to represent non-automated ones (activities performed without using a computer system). Moreover, bold lines represent automated interactions (interactions performed using a computer system), and regular lines represent non-automated interactions (interactions performed without using a computer system).

Brief use case is the first level of detail of use cases that is elaborated to casual and fully-specified levels of detail later on in the software development process. Typically, brief use cases are the main source for the estimation required to perform feasibility studies. In a pilot empirical study, Issa [4] calculated a number of use case based metrics for three

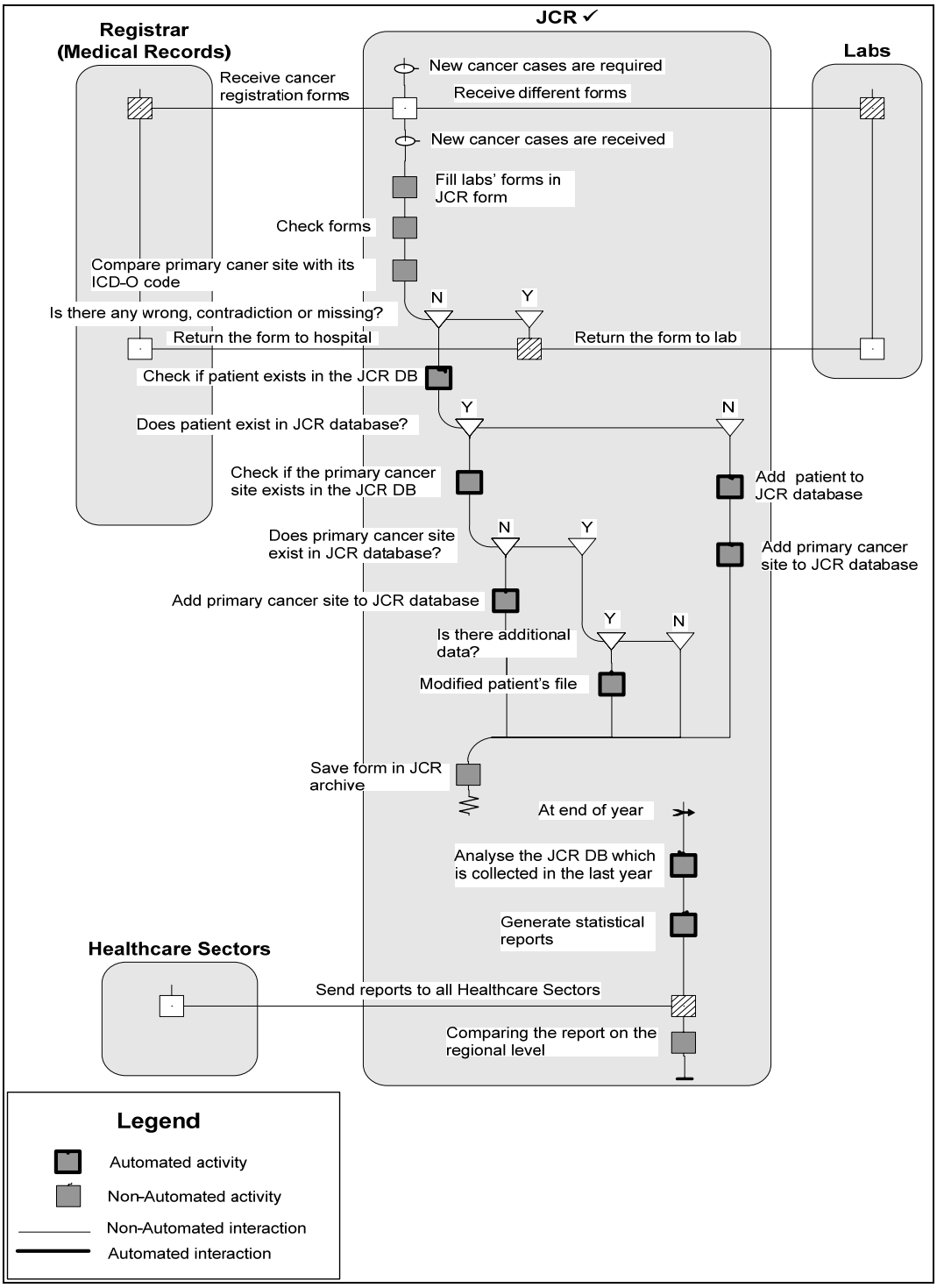

Figure 2: Improved RAD model of Cancer Care Registration Process in Jordan [12].

test projects consisting of 64 use cases. One of the calculated metrics is the average number of interactions in a brief use case. He found that brief use cases may not contain more than 5 interactions due to imprecise requirements and the unclear vision of clients. The result of dividing the number of automated transactions (interactions and activities) of a business process modelled using improved RAD notations by the average number of interactions per brief use case represents the number of candidate use cases to be derived from the process model:

$$Number\ of\ Candidate\ Use\ Cases\ Per\ BPM = \left\lceil \frac{Number\ of\ Transactions\ per\ BPM}{Average\ Number\ of\ Interactions\ per\ Use\ Case} \right\rceil \quad (1)$$

According to Issa et al. [4], the availability of the candidate number of use cases in a system is the main key to perform project feasibility analysis. This is attributed to the calculated average development effort (0.67 man-month) per use case metric [13]. This metric can be used in conjunction with the number of candidate use cases to calculate the total development effort as:

$$Effort = Total\ Number\ of\ Candidate\ Use\ Cases \times Average\ Effort\ per\ Use\ Case \quad (2)$$

Consequently, project managers can use the estimated effort with other project parameters (e.g. project deadlines, resources, etc.) to conduct their early stages what-if analysis to inform the feasibility and profitability of the project in-hand. Figure 3 summarizes the proposed approach to drive software systems feasibility studies using business process models.

IV. CASE STUDY: THE CANCER REGISTRATION PROCESS

The process of Cancer Registration (CR) in Jordan will be used as an example to demonstrate the application of the approach explored in the last section.

Cancer registration in Jordan is managed by the **J**ordan **C**ancer **R**egistry (**JCR**) established in 1996 as a population-based registry as a unit of the Ministry of Health. Malignant and some benign cases have been recorded by JCR for people living in Jordan since January 1996 [12].

Cancer registration is the systematic collection and classification of data on all types of cancers and persons diagnosed with cancer. Cancer registries aim to determine the distribution of cancer, monitor the growth of cancer per type, evaluate the current treatment process, and monitor patient survival rates. This process has been modelled using improved RAD as shown in Figure 2. The main roles involved are: *JCR; Laboratories; Registrars; and Healthcare Sectors.*

When the JCR receives cancer registration forms from registrars and laboratories, patients' personal information is checked to confirm that there is not any missing information, and then the primary cancer sites are verified against the international ICD-O (International Classification Diseases for Oncology) encoding. These forms are also checked for any contradictions, omissions, etc. After that, each patient's details and respective cancer details are added to the JCR database, or updated if the patient's details already exist in the JCR database. The information collected is analysed, and statistical reports are generated which are sent to the respective health sectors. A cross-comparison between JCR reports and interrelated ones are performed to assess the results. Table 1 describes the roles, activities and interactions in the cancer registration process, [14].

The developed RAD model shown in Figure 2 can be used to identify the number of automated and non-automated activities and interactions performed throughout the process as shown in Table 2. Each activity or interaction generates transaction. As discussed in previous section, the automated activities and interactions will be allocated to use cases and the average number of interactions per use case is 5. Thus, the anticipated number of use cases to realize the functionalities of the cancer registration process is ***2***.

Consequently, the anticipated number of use cases is used to calculate the required total effort as 1.34 man-months. This estimated total effort is then used in conjunction with other project parameters to conduct what-if analysis; and hence, perform early feasibility study for the anticipated software system required to automate the cancer registration process in Jordan.

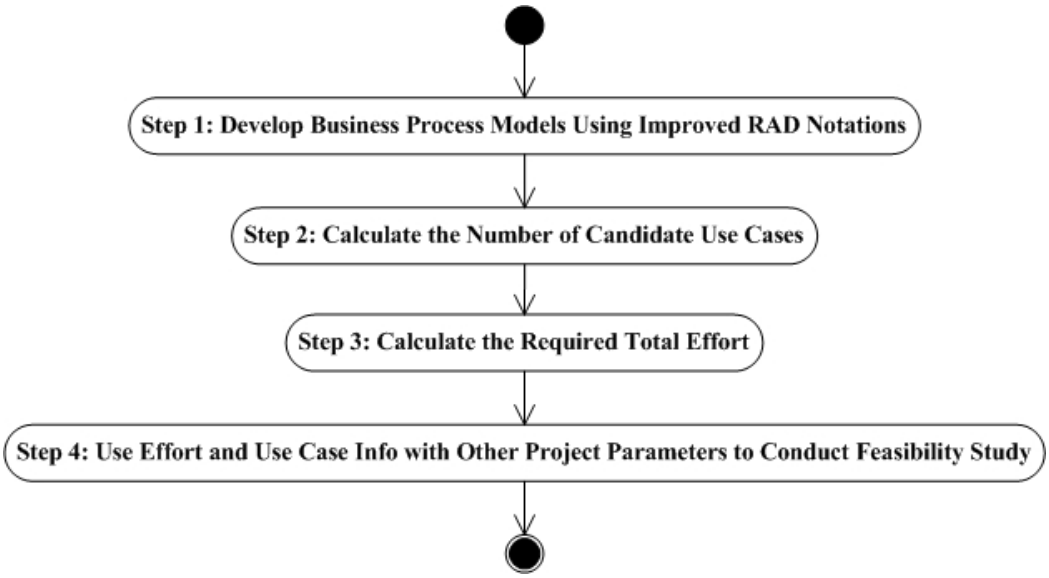

Figure 3: The Proposed Approach to Perform Feasibility Studies Using Business Process Models.

Table 1: The Cancer Registration Process: Roles, Activities and Interactions.

Goal: collection and classification of data, and cancer control		
Role	**Activities**	**Interactions**
JCR	Check forms, verify if patient and primary cancer site exist in JCR database, saving new cases and new primary cancer sites, generate annual reports, compare the reports regionally	Registrar (Medical Records), Laboratories, Health sector
Laboratories	Send JCR form	JCR
Registrar	Send JCR form	JCR

Table 2: Summary CR Process Activities and Interactions.

Process	**No. of auto-actions**	**No. of non auto-actions**	**No. of auto-interactions**	**No. of non auto-interactions**
Cancer registration	8	5	0	5

V. CONCLUSION AND FUTURE WORK

This paper has investigated the use of process modeling to conduct early feasibility study of software development projects. A new structured approach is developed to perform feasibility study using process models represented using RAD. This approach is based on the derivation of number of use cases from RAD models and then utilizing this information to estimate total effort leading naturally to project feasibility analysis. The proposed approach includes four phases namely: develop business process models using improved RAD notations, calculate the number of candidate use cases, calculate the required total effort, and use effort and use case information with other project parameters to conduct feasibility study.

The proposed approach has been demonstrated using Cancer Registration process in Jordan as a case study. The results of practical application of the proposed approach indicate the ease and timeliness of which software development projects feasibility studies can be performed in the early stages of software development using business process models.

Further validation and evaluation for the proposed approach to feasibility studies are undertaken using a number of current software development projects. Also, further work is planned to thoroughly investigate the possibility of utilizing business process models to extract other software metrics in relation with software architecture and logical design. Finally, the aim is to utilise RAD business process models as an input to an automated software cost estimation tool based on use cases.

REFERENCES

[1] Boehm, B., Horowitz, E., Madachy, R., Reifer, D., Clark, B., Steece, B., Brown, A., Chulani, S. and Abts, C., *Software Cost Estimation With Cocomo II*. Upper Saddle River, NJ : Prentice Hall, (2000).

[2] Westland, J., The Cost of Errors in Software Development: Evidence From Industry *Journal of Systems and Software*, 62 (1), 2002, pp.1-9.

[3] Odeh M., Kamm R., Bridging the gap between business models and system models. *Information and Software Technology*, (45) 15, 2003, pp. 1053-1060.

[4] Issa, A., Odeh, M., and Coward, D., Software Cost Estimation Using Use-Case Models: A Critical Evaluation *in In Proceedings of the 2nd IEEE International Conference on Information and Communication Technology: From Theory To Applications* Damascus, Syria, 2006, pp.1-6.

[5] Karner, G., *Resource Estimation for Objectory Projects.* Objectory Systems, 1993.

[6] Ould M., *Business process management, a rigorous approach*. Swindon: British Computer Society, 2005.

[7] Giaglis, A taxonomy of business process modelling and information system modelling techniques. *The International Journal of Flexible Manufacturing Systems*, 13, 2001, pp 209 – 228.

[8] Saven, S., Business process modelling: review and framework. *International Journal of Production Economics*, 90 (2), 2003, pp 129-149.

[9] Jacobson, I., *Object-Oriented Software Engineering: a Use Case Driven Approach*. Rev. printing. Addison-Wesley, 1992.

[10] Kruchten, P., *The Rational Unified Process : an Introduction*. Swedish edition. Boston, Mass. ; London : Addison-Wesley, 2002.

[11] Aburub, F., Odeh, M., Beeson, I., Modeling Healthcare Processes Using Role Activity Diagramming. *International Journal of Modeling and Simulation*, (Accepted and to Appear).

[12] Aburub, F., A Business Proceess Improvement Methodology based on Process Modeling, applied to Healthcare Sector. Ph.D. Thesis, University of the West of England, UK, 2006.

[13] Issa, A., Odeh, M., and Coward, D., Using Use Case Patterns To Estimate Reusability in Software Systems *Information and Software Technology*, 2006, 48 (9), pp. 836-845.

[14] Aburub, F., Odeh, M., and Beeson, I., Modelling Non-Functional Requirements of Business Processes. *Information and Software Technology*, (Accepted and to Appear).