Deriving System Model From Business Process Model

(JCR Case Study)

By

Ahmad Jehad Ghazal

Supervisors

Prof. Mohammad Al – Fayoumi
Dr. Faisal Aburub

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE MASTER DEGREE IN COMPUTER INFORMATION SYSTEMES

Computer Information System Department
In the Faculty of Information Technology
Middle East University for graduate Studies

Amman, Jordan
July, 2009

جامعة الشرق الأوسط

إقرار تفويض

أنا, أحمد جهاد غزال ، أفوض جامعة الشرق الأوسط للدراسات العليا بتزويد نسخ من رسالتي للمكتبات أو المؤسسات أو الهيئات أو الأشخاص عند طلبها حسب التعليمات النافذة في الجامعة.

التوقيع:

التاريخ: ٢٥/٨/٢٠٠٩

Middle East University

Authorization statement

I, Ahmad Jehad Ghazal, authorize the Middle East University for Graduate Studies to supply copies of my thesis to libraries, establishments or individuals on request, according to the university regulations.

Signature:

Date: 25/8/2009

COMMITTEE DECISION

This Thesis **(Deriving System Models from Business Process Models)** Was successfully defended and approved on 11 August 2009.

Examination Committee Signature

Prof. Mohammad Ahmad Al Fayoumi
Professor / Department of Computer Information System
(Middle East University for Graduate Studies)

Prof. Musbah Aqel
Professor / Department of Computer Science
(Middle East University for Graduate Studies)

Dr. Faisal Al-Said Fared Abu-Alrob
Associate Professor / Department of Computer Information System
(Al Petra University)

Dr. Ayman Ahmad yousef Issa
Assistant Professor / Department of Software Engineering
(Philadelphia University)

Declaration

I do hereby declare the present research work has been carried out by me, under the supervision of Professor. Mohammad A. Al-Fayoumi and Dr. Faisal Aburub. And this work has not been submitted elsewhere for any other degree, fellowship or any similar title.

Date: 25 / 8 / 2009

Ahmad Jehad Ghazal

Department of Computer Information Systems

Faculty of Information Technology

Middle East University for Graduate Studies

DEDICATION

To my father, mother, brother, and sister, for their love and support, they were the light in my academic path and without them; nothing of this would have been possible.

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to Professor. Mohammed Al-fayoumi and Dr. Faisal Aburub for their guidance and, support and motivation throughout my research.

And I also want to thank and express my sincere appreciation to Jordan cancer registry for their great support for my study in this thesis for providing me with the necessary information in order to complete my thesis.

List of Figures

List of Tables

Table of Contents

ABSTRACT

Business Process Modelling (BPM) is the activity of representing processes of an organization, so that the current process may be analyzed and improved in the future. It provides a facility to decision makers which help them filter out irrelevant complexity in the real world and concentrate on the most important parts of the process under study. BPM is typically performed by business analyst and managers who are seeking to improve process efficiency and quality. So, BPM is used mainly to provide a comprehensive business process analysis and improvement.

A system model can be defined as an abstract description of a system whose requirements are being analyzed. The availability of BPMs in any organizations may not tie to any corresponding software system. Rather, they may exist much before automation of the business process of any organization. Bridging the gap between BPMs and system models is important for the different stakeholder involved, so as to ensure that an anticipated software system will be in line with the business processes. The gab may be further widened given the rate at which software technology develops, for example with the emergence of grid technology in business environment. Therefore, the challenge here to investigate and synthesis approaches that will in some way support, and perhaps partially or fully automate the business process so that the process is not simply modeled but also enacted, through an IT-based solution.

This thesis aims to develop a new approach for linking business process models and system models. Our new approach passes through the six phases used in developing a business process model. The process modeling language RAD (Role Activity Diagramming) is used as one of the main technique. The suggested method derives system models based on the results of the enhanced understanding,

analysis, and evaluation of the business process models. Our suggested approach uses the healthcare process as running example.

The new approach which includes detailed modeling and analysis saves plenty of time which was wasted in data gathering phase in system development life cycle and producing a prototype model for the system from the business process which will be more significant.

ملخص

نمذجة العمليات التجارية (BPM) هي التي تمثل نشاط عمليات المنظمة ، وذلك أن العملية الحالية قد يتم تحليلها وتحسينها في المستقبل. وهو توفر تسهيلات لصانعي القرار لمساعدتهم في التقليل من التعقيدات في العالم الحقيقي ، والتركيز على أهم أجزاء العملية قيد الدراسة. نمذجة العمليات التجارية عادة يقوم بها المحللين والمديرين الذين يسعون لتحسين كفاءة ونوعية العملية. بذلك ، نمذجة العمليات التجارية تستخدم بشكل رئيسي لتوفير تحليل شامل للعمليات التجارية وتحسينها.

النظام النموذجي يمكن تعريفه بأنه وصف مجرد للنظام الذي يجري فيه تحليل الاحتياجات. توافر نمذجة العمليات التجارية في أي منظمه لا يجوز ربطها بأية برمجيات النظام تقابلها. بل إنها قد تكون موجودة من قبل التشغيل الآلي للعمليات التجارية في أي منظمة. سد الفجوة بين النظام و نمذجة العمليات التجارية مهمة لمختلف أصحاب المصلحة المعنيين ، وذلك لضمان وجود نظام برمجيات من المتوقع أن يكون متماشيا مع العمليات التجارية. الفجوة يمكن زيادة توسيعها نطاق معدل تطور تكنولوجيا البرمجيات ، على سبيل المثال مع ظهور تكنولوجيا الشبكة في بيئة الأعمال التجارية. وبالتالي ، فإن التحدي هنا للتحقيق والتوليف النهج من شأنه أن تدعم على نحو ما ، وربما جزئيا أو كليا في عملية أتمتة الأعمال ذلك ان العملية ليست فقط و ببساطة تم نمذجتهاوإنما أيضا سنت ، من خلال حلول قائمة على تكنولوجيا المعلومات.

هذه الرسالة تهدف الى وضع نهج جديد لربط نماذج العمليات التجارية ونماذج النظام. نهجنا الجديد يمر عبر ست مراحل المستخدمة في تطوير العمليات التجارية النموذجية. اللغة المستخدمة في عملية النمذجة هي لغة رادوالتي تعتبر من واحدة من أهم التقنيات. الطريقة المقترحة تستمد نماذج النظام على أساس نتائج تعزيز الفهم والتحليل والتقييم لنماذج العمليات التجارية. النهج المقترح يستخدم عملية إدارة الرعاية الصحية كمثال تطبيقي.

النهج الجديد الذي يتضمن تفصيلا للنمذجة والتحليل يوفر الكثير من الوقت الذي يضيع في مرحلة جمع البيانات في دورة حياة النظام ، وينتج نموذج أولي اكثر اهمية لنمذجة نظام من العمليات التجارية.

CHAPTER ONE

Introduction

In this chapter, we give a brief background about the thesis scope, then the problem of the thesis will be stated. We will give information about related works, the contribution of the thesis, and the outline of the thesis.

1.1 An Overview

Many modern organisations have large and complex processes which include individuals, groups, information technology, and a web of complex interactions among individuals, organisation, and technology. Customers expect high quality products and services while organisations face rapid changes in the business environment. Therefore, organisations need methods to support their activities, functions, and interactions in order to meet customer expectations.

Moreover, [Shin and Jemella, 2002] indicate that "information technology has profoundly changed the way we do business during the past decade". Therefore, organisations need to manage these technological changes and identify methods which will help them to perform these changes efficiently, effectively and in the most suitable way.

[Hindle ,1997] defines a business process as a sequence of logically linked activities, where each activity has a set of inputs and value added tasks that produce a set of outputs to meet organisational and customer requirements. Business processes involve people, machines, departments, companies, technology… etc, which interact with each other in order to achieve goals of the business. [Ould ,2005] stated that "a process involves activity: people and/or machines do things. A process also generally involves more than one person or machine: a process is about groups; it concerns collaborative activity. And process has a goal: it is intended to achieve something". Healthcare processes and banking processes are example of business processes. The word 'business' relates to 'what a certain process or organisation intends to carry out'. For example, the business of healthcare processes or organisations is to provide health care, while the business of banking processes is to provide banking services. According to [Rolstadas ,1995], a business process has interconnected activities that aim to provide output and service through using material, information, equipment and people

One method to support an organization is by automating its business processes using software systems. Unfortunately, developing software systems is sometimes difficult and expensive process as it needs to go through system development life cycle. So, it is important to make sure that the resulted software system is economically feasible and support business process of such organization effectively and sufficiently otherwise, the software systems will fail.

Process modeling can be used to provide a comprehensive understanding of business activities and functions and thence a base for detailed process analysis. Business process modeling can also be used to represent business processes of a certain organization. Process modeling can also be used to analyse and improve business processes. [Warboys et al, 1999] stated that there are a number of useful reasons for modeling, such as that it help to describe a process, analyse a process, and enact a process. [Giaglis, 2001] argues that modeling is at the core of organisational design and information systems (IS) development. He added that business process modeling provides a facility to decision makers which will help them filter out irrelevant complexity in the real world and concentrate on the most important parts of the process under study. Also, business process modeling can be used to contribute positively in the software development process. [Tam et al.

,2001] argued that BPM aims to identify critical processes, improve the overall performance of the business, form a tool for business process re-engineering, identify appropriate strategies for software package implementation, and help with software development. [Phalp, 1998] pointed out that business process modeling techniques can be used for traditional software development as well as for facilitating business processes improvement or restructuring. Therefore, business process modeling can be used to develop software systems that support business processes of a certain organization effectively. Also, the resulted software systems can be used to do early cost estimation study.

Thus, the usefulness of business process modeling in developing software systems motivated this research to propose a new approach to bridge the gap between the two. This new approach utilizes business process modeling to derive system models.

1.2 Research Problem

Business Process Modeling (BPM) can be defined as the representation of one or more of the process perspectives to understand, analyze, and/or improve automated and/or non-automated business processes. Hence, the availability of business process models in any organization is not tied to any corresponding software system. Rather, they may exist much before the automation of the business of any organization. However, business process modeling can be used to contribute positively in the software development process.

There is scope for expanding consideration of the role of IT in enhancing the business process. Bridging the gap between business process models and system models is important for the different stakeholders involved so as to ensure that an anticipated software system will be in line with the business processes. However, the gap may be further widened given the rate at which software technology develops, for example with the emergence of grid technology in business environments. Therefore, the challenge here is to investigate and synthesize approaches that will in some way support, and perhaps partially or fully automate the business process so that the process is not simply modeled but also enacted, through an IT-based solution. These approaches can range from being semi-formal to formal. As an example of a semi-formal approach, the notion of coloured Role Activity Diagram (RAD) models, Business Process Models can be used to pinpoint which activities can be automated, partially or fully. It might be possible to trace systematic relationships between coloured RAD models and Data Flow Diagram (DFD) (System Models). It would be useful to investigate if DFD for the anticipated software system can be partially derived using coloured RAD models of a business process, and incrementally refined as the business process is investigated and revisited.

1.3 Research Questions

This research aims to use business process modeling as a basis for developing a software system models. The main research questions are:

How can business process modeling methods be used to derive software system models?

To what extent the extracted system models are accurate and precise?

To what extent can the generated system models be utilised to drive subsequent software development phases?

1.4. Objectives of the Study

The research aims are identified as follows:

To develop a new approach to link between business process models and system models.

To investigate the accuracy of the resulted system models using case study.

1.5 Significance

This research uses the process modeling language RAD (Role Activity Diagramming) as one of its main techniques. Role Activity Diagrams (RADs) are diagrammatic notations to represent and model coordinated behaviour and interactions within a process.

In this thesis we will model the healthcare process using RAD. We will also suggest a method to derive system models based on the results of the enhanced understanding, analysis and evaluation of the business process models.

The derived system model will save us plenty of time which is wasted in data gathering phase in system development life cycle (SDLC) and producing a prototype model for the system from the business process which will be the significance of the research.

1.6 Research Investigation Plan

Understanding and improving business processes, including their quality attributes, can be performed using the following sequence of connected and logically ordered steps:

Step 1:

Conduct interviews and observations in order to understand the healthcare process in Jordan.

Step 2:

Use RAD to model the healthcare process.

Step 3:

Conduct interviews to validate the resultant RAD models.

Step 4:

Analyse the current healthcare process through its RAD models.

Step 5:

Suggest method(s) to derive the system models based on the results of the enhanced understanding, analysis and evaluation of the business process models.

Step 6:

Develop system models using DFDs

Step 7:

Validate the resultant system models using the case study

CHAPTER TWO

Literature Survey

2.1 Business Process:

Business process could have three waves through decades in the first wave the focus was on repeated sequential activity and used simple diagrams such as flowchart to model and stored it in procedure manuals to understand how to improve the process as in total quality management (TQM).

In the second wave, information was more important than processes and the processes could no longer be changed without expensive re-engineering of the underlying information system. And also enterprise resource planning (ERP) took over the Processes by defining the processes of the whole areas of organizational activity and build them into their systems , As a result everyone got the same competitive advantage.

In the third wave there was a term first said by Howard smith and Peter finger the term was business process management (BPM), in this wave the focus is on the processes not information first the process become visible, changeable on the fly and got its own separate existence [Ould ,2005] .

There are three types of business processes:

1. Management processes - the processes that govern the operation. Typical management processes include "Corporate Governance" and "Strategic Management".

2. Operational processes - these processes create the primary value stream, they are part of the core business. Typical operational processes are Purchasing, Manufacturing, Marketing, and Sales.

3. Supporting processes - these support the core processes. Examples include Accounting, Recruitment and IT-support

[Davenport, 1993] defines a business process as:
”A structured, measured set of activities designed to produce a specific output for a particular customer or market. It implies a strong emphasis on how work is done within an organization, in contrast to a product focus’s emphasis on "what". A process is thus a specific ordering of work activities across time and space, with a beginning and an end, and clearly defined inputs and outputs: a structure for action. ... Taking a process approach implies adopting the customer’s point of view. Processes are the structure by which an organization does what is necessary to produce value for its customers.”

The emphasis in this definition is on the structural component of a process.

The second definition is by [Hammer and Champy’s, 1993] in which they define a process as ”a collection of activities that takes one or more kinds of input and creates an output that is of value to the customer.”

According to [Johansson et al, 1993] they define a process as” a set of linked activities that take an input and transform it to create an output. Ideally, the transformation that occurs in the process should add value to the input and create an output that is more useful and effective to the recipient either upstream or downstream.”, The third definition by [Rummler and Brache ,1995] stating that ”a business process is a series of steps designed to produce a product or service. Most processes (...) are cross-functional, spanning the ‘white space’ between the boxes on the organization chart. Some processes result in a product or service that is received by an organization's external customer. We call these primary processes. Other processes produce products that are invisible to the external customer but essential to the effective management of the business. We call these support processes."

This definition focuses on the external customers of an organization.

And [Ould, 2005] defined the process from two perspectives, first definition is that "A process is a coherent set of activities carried out by a collaborating group to achieve a goal, the chunking of organizational activity into processes must be driven by understanding of what the business the organization is in".

The second definition is that "A process is a coherent set of actions carried out by a collaborating set of roles to achieve a goal".

As we can see from both definitions by Olud, they both describe the process in simple way but the second definition is more detailed.

Finally [Taylor, 2007] says that "A Business Process is a specific ordering of activities and decisions across time, space and participants. A business Process has a beginning, an end, and clearly defined inputs and outputs, actions and steps".

In my opinion Ould`s second definition is the most accurate one. If I was to put my own definition of business process because it's described business process in more detail than other definitions, in terms of action is part of activities and activities are part of a process.

We can compile the following list of characteristics for a business process:

"1. Definability: It must have clearly defined boundaries, input and output.

2. Order: It must consist of activities that are ordered according to their position in time and space.

3. Customer: There must be a recipient of the process' outcome, a customer.

4. Value-adding: The transformation taking place within the process must add value to the recipient, either upstream or downstream.

5. Embeddedness: A process can not exist in itself; it must be embedded in an organizational structure.

6. Cross-functionality: A process regularly can, but not necessarily must, span several functions."

[Available from: http://en.wikipedia.org/wiki/Business_process].

2.2 Business Process Modeling:

"Business process modeling (BPM) in systems and software engineering is the activity of representing processes of an enterprise so that the current process may be analyzed and improved in future. BPM is typically performed by business analysts and managers who are seeking to improve process efficiency and quality. The process improvements identified by BPM may or may not require Information Technology involvement, although that is a common driver for the need to model a business process by creating a process master"

[Available from: http://en.wikipedia.org/wiki/Business_process_modeling]

There are some business rules that govern how things should be done such as policy, procedure, standards, to understand how that relate to BPM. [Havey, 2005] defines BPM as "a set of technologies and standards for the design, execution, administration, and monitoring of business processes. A business process is the flow or progression of activities each of which represents the work of a person, an internal system, or the process of a partner company toward some business goal".

To explain what Michael heavy meant by his definition he proposed an architecture that shows the collaboration between business process standards as shown in Figure (2.1)

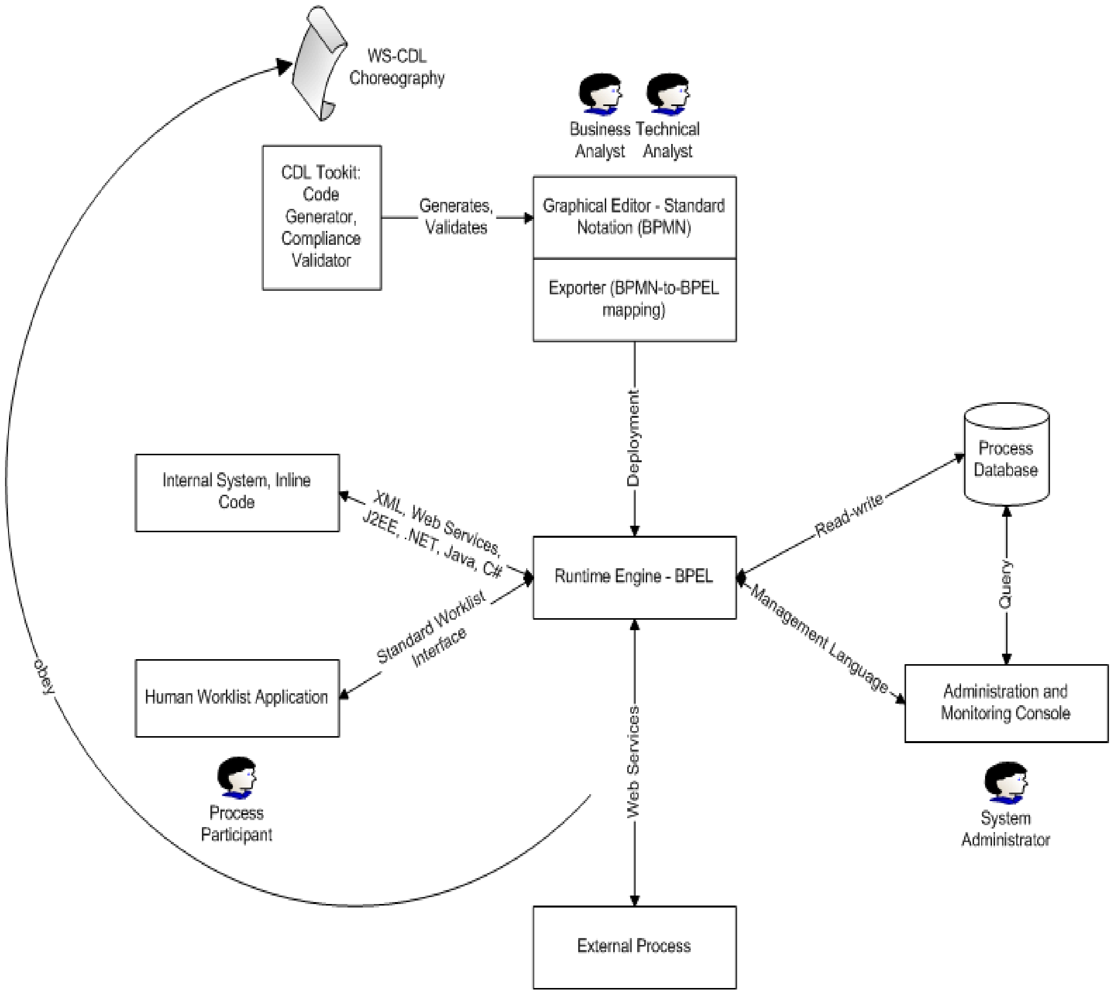


Figure 2.1. BPM architecture

By looking at the definitions above we can say that by modeling a process it can simply provide the individuals and groups a better understanding of the organization business, and also if an organization is concerned about the quality of service or product they should concern first with the process that deliver it ,therefore process comes before information so when we decide how to carry out our business and process we can know what information we need and so the information based system that supports them.

A model is right if it helps, it helps if it reveals things we want to know, or helps us answer a question, or can be analyzed, or can be adjusted to test proposed changes, or simply aids understanding, so saying that the parallel between a map and a model should be clear before we carry out some process modeling we need to know what we want exactly from the model so we can choose the scale and the sort of detail it shows.

So in the model it self we should show that the business process has a goal and got internal or external entities including specific inputs and outputs and also have resources and got activities that are carried out in some order that may affect more than one organizational unit and in the end create some value .

Process modeling must enable us to:

* take an architectural view of our process.

* expose our process as a discovery activity.

* define our processes appropriately in a regulated environment.

* use our models to diagnose our process.

* design new process from scratch

* design work flow that can be supported by work flow management system.

* produce the process definitions that can be enacted in a BPMS.

Modeling a process faces significant challenges. Modeling should ensure thoroughness and consistency in the capture of relevant information so that both developers and business analysts can understand the business requirements that are captured in the model. During modeling, alternatives and exceptions to standard procedures must be captured in addition to normal operations. Professionals with different scopes of interest and expertise can build process models to meet a wide range of business objectives.

The task of modeling a process consists of defining the details of a business process flow and modeling all the data, resources, and other elements that the flow uses. A business process is composed of process steps that are normally connected through control flows, and these control flows connect activities with decision nodes. A decision node holds the business rules that are evaluated to decide what path in the process should be followed. Modeling includes the decomposition of a BP into sub-processes and adding required process elements to the model.

Furthermore business process modeling can be defined as a representation of reality in order to understand, improve, and manage that reality.

2.3 Importance of Business Process Modeling:

The main question we should ask ourselves is why business process is important? When you are looking for a better outcome, should see what are doing and try to change it. We always think about the detail. We look at the activity we are involved with and try to find a better way of being more efficient. The 'what you do' should be broader than the activity, the document, the technology. The organization should focus on the sum of all these and how they flow together in complete business processes.

Of course the 'different outcome' we all crave is a business that is more efficient, drives down costs and improves customer service. A business process will therefore cut across the whole organization and does not focus on the individual department.

So what else can we do to improve efficiency if not the process? An organization is built on the three pillars of:

1. People.

Having the right people, motivated and performing is naturally a key requirement to performance.

2. Technology

Providing the people with the right tools to do their jobs well is also vitally important. Computer technology has revolutionised the office environment, and with web technologies and mobile computing we are all becoming much more efficient for longer.

3. Process.

This is what holds it all together. The glue between the assets.

Business process ensures that the requirements must be in high quality and meets the following conditions:

1. Unambiguous. Everyone can agree to a single interpretation of what is required.

2. Complete. All "in-scope" stakeholders have offered their input and no one has been left out who will be impacted by the requirements defined. A requirement is complete if it needs no further explanation.

3. Testable. It's possible to test and verify that each requirement actually meets its intended use.

4. Modifiable. It's possible to change or update a given requirement if user needs change due to changes in policy, practice or regulation. A requirement is modifiable when it can be changed without impacting other parts of the requirement.

5. Correct. Requirements are correct when stated accurately, describe what's needed, and have all stakeholders in agreement with what is documented.

6. Necessary. The requirements are actually required for fulfillment of a business-level objective and not just a "wish-list" item for future enhancement.

7. Feasible. Requirements are considered feasible if current technology actually supports the ability to carry out the requirement as defined within known identified constraints.

8. Consistent. No conflicts with other requirements exist.

9. Traceable. Each uniquely-identified requirement links to a test and each proposed test links back to a specific requirement. Business requirements link to user requirements; and user requirements link to either functional requirements or quality-of-service requirements, or both.

10. Prioritized. There is a clear statement of which requirements are most important relative to other requirements. This enables phasing of development efforts and execution of tasks to support breaking the implementation down into smaller, more easily managed steps. There is also value in defining dependencies between requirements as part of the prioritization process. [Available from: http://www.metarasa.com/management/process/the-importance-of-business-processes].

2.4 Business Process Modeling Techniques:

Business process modeling tools provide business users with the ability to model their business processes, implement and execute those models, and refine the models based on as-executed data. As a result, business process modeling tools can provide transparency into business processes, as well as the centralization of corporate business process models.

Some of the Business process modeling tools summarized as follow:

* Use Case Diagrams:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioural diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals represented as use cases, and any dependencies between those use cases.

The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

* Activity Diagrams:

Activity diagrams are a loosely defined diagram technique for showing workflows of stepwise activities and actions, with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

* Business Process Modeling Notation (BPMN):

The BPMN is a standard for business process modeling, and provides a graphical notation for specifying business processes in a Business Process Diagram (BPD) [. Simpson 2004], based on a flowcharting technique very similar to activity diagrams from Unified Modeling Language (UML) [32]. The objective of BPMN is to support business process management for both technical users and business users by providing a notation that is intuitive to business users yet able to represent complex process semantics. The BPMN specification also provides a mapping between the graphics of the notation to the underlying constructs of execution languages.

*Cognition enhanced Natural language Information Analysis Method business (CogNIAM): Strives to effectively involve users and customers in the and designing process. This involvement can be achieved by using a modeling small but exact subset of natural language as a means of communication and . The use of the specifications information systemdefining business rules and natural language subset and associated CogNIAM methodology means understand and unambiguous communication between stakeholders stakeholdersability for all [Nijssen, 2008].

* Extended Business Modeling Language (XBML):
XBML stands for extended Business Modeling Language and is used to define the [Tyler and Baker , 2007] It is based upon a 5 organization of an business processes dimensional business framework (What, Who, Where, When and which) and is uniquely supported by approximately 55 rules that govern the usage, "output" and "syntax" of the language.

* Event-driven Process Chain (EPC):
Event-driven Process Chains are a business process modeling technique; mainly used for analysing processes for the purpose of an ERP implementation [Hommes, 2004] An EPC is an ordered graph of events and functions. It provides various connectors that allow alternative and parallel execution of processes. [Tsai et al, 2006].

* IDEF0 used since early 1990s:
IDEF0 (Integration Definition for Function Modeling) is a function modeling methodology for describing manufacturing functions, which offers a functional modeling language for the analysis, development, reengineering, and integration of information systems; business processes; or software engineering analysis [Belvoir, 2001].

* Unified Modeling Language (UML)

The Unified Modeling Language (UML) is an open method used to specify, visualize, modify, construct and document the artifacts of an object-oriented software intensive system under development, UML offers a standard way to write a system's blueprints, including conceptual components

Such as:

* Actors
* Business processes and
* System's components, and activities

In the end there are too many techniques and tools to model business processes so [Regev et al, 2002], classified all approaches into four categories according to the main view they take over the business process dynamics:

1. Input / Output Diagrams

These diagrams focus on the inputs and outputs of processes, creating a causal relationship between different processes. The best known example of this type of diagram is IDEF 0, which permeates defence organisations worldwide.

2. Workflow:

These diagrams focus on the order of activities like a recipe for baking a cake. Examples of this type of model include BPMN, IDEF 3 and UML Activity Diagrams.

3. Agent-based Diagrams:

These diagrams focus on the agents that perform the activities and the best example I'm aware of is UML Collaboration Diagrams.

4. State Flow:

State flow diagrams are based on the state of an object and how the state changes based on different events. UML State charts are a prime example.

2.5 System Models:

A system model defined as an abstract description of a system whose requirements are being analyzed.

System modeling helps the analyst to understand the functionality of the system and models are used to communicate with customers.

System modeling shows how the system should be working. This technique is used to examine how various components work together to produce a particular outcome. These components make up a system, which is comprised of resources processed in various ways (counselling, diagnosis, treatment) to generate direct outputs (products or services), which in turn can produce both direct effects on those using them and longer term, more indirect results on users and the community at large.

By diagramming the linkages between each system activity, system modeling makes it easier to understand the relationships among various activities and the impact of each on the others. It shows the processes as part of a larger system whose objective is to serve a specific client need. System modeling is valuable when an overall picture is needed. It shows how direct and support services interact, where critical inputs come from, and how products or services are expected to meet the needs in the community. When teams do not know where to start, system modeling can help in locating problem areas or in analyzing the problem by showing the various parts of the system and the linkages among them. It can pinpoint other potential problem areas. System modeling can also reveal data collection needs: indicators of inputs, process, and outcomes (direct outputs, effects on clients, and/or impacts). Finally, system modeling can be helpful in monitoring performance.

System modeling uses three elements: inputs, processes, and outcomes.
Inputs: are the resources used to carry out the activities (processes). They can be raw materials, or products or services produced by other parts of the system. For example, in the malaria treatment system, inputs include anti-malarial drugs and skilled health workers. Other parts of the system provide both of these inputs: the drugs by the logistics subsystem and the skilled human resources by the training subsystem.

Processes: are the activities and tasks that turn the inputs into products and services. For malaria treatment, this process would include the tasks of taking a history and conducting a physical examination of patients complaining of fever, making a diagnosis, providing treatment, and counseling the patient.

Outcomes: are the results of processes. Outcomes generally refer to the direct outputs generated by a process, and may sometimes refer to the more indirect effects on the clients themselves and the still more indirect impacts on the wider community.

Outputs are the direct products or services produced by the process. The outputs of the malaria treatment system are patients receiving therapy and counselling. [Available from: http://www.qaproject.org/methods/resysmod.html]

System modeling is a technique to express, visualize, analyze and transform the architecture of a system, a system may consist of software components, hardware components, or both and the connections between these components. A system model is then a skelet al model of the system.

System modeling is intended to assist in developing and maintaining large systems with emphasis on the construction phase. The idea is to encapsulate complex or changeable aspects of a design inside separate components with well-defined interfaces indicating how each component interacts with its environment. Complete systems are then developed by composing these components. System modeling can increase reliability and reduce development cost by making it easier to build systems, to reuse previous built components within new systems, to change systems to suit changing requirements such as functional enhancement and Platform changes, and to understand systems. In this way, a system model can satisfy different requirements such as documenting the system, providing a notation for tools such as consistency checkers and can also be used in the design stage of system development.

Modeling techniques differ in the extent to which their construct highlight the information that answers the analyst questions. To provide this information, a modeling technique should be capable of representing one or more of the following process perspectives:

1. The functional perspective: representing what process elements (activities) are being performed.
2. The behavioral perspective: representing when activates are performed , as well as aspects of how they are performed through feedback loops, iteration decision making conditions and so on .
3. The organizational perspective: representing where and by whom activities are performed, the physical communication mechanism used to transfer entities, and the physical media and locations used to store entities.
4. The informational perspective: representing the informational entities (data) produced or manipulated by a process and their relationships [Curtis et al, 1992].

There are Different Types of System Model:

* Data processing model showing how the data is processed at different stages.
* Composition model showing how entities are composed of other entities.
* Architectural model showing principal sub-systems.
* Classification model showing how entities have common characteristic.
* Stimulus/response model showing the system's reaction to events.

2.6 System Models Techniques:

1. Data Flow Diagram (DFD):

Is a graphical representation of the "flow" of data through an information system. It differs from the system flowchart as it shows the flow of data through processes instead of hardware. These show the processing steps as data flows through a system.

DFDs are an intrinsic part of many analysis methods and simple and intuitive notation that customers can understand and also Show end-to-end processing of data. DFDs model the system from a functional perspective Tracking and documenting how the data associated with a process is helpful to develop an overall understanding of the system. DFD may also be used in showing the data exchange between a system and other systems in its environment.

2. Entity Relationship Diagram (ERD):

An ERD is a model that identifies the concepts or entities that exist in a system and the relationships between those entities. An ERD is often used as a way to visualize a relational database: each entity represents a database table, and the relationship lines represent the keys in one table that points to specific records in related tables. ERDs may also be more abstract, not necessarily capturing every table needed within a database, but serving to diagram the major concepts and relationships.

3. State Transition Diagram (STD):

A state diagram is a type of diagram used in computer science and related fields to describe the behaviour of systems. state diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. There are many forms of state diagrams, which differ slightly and have different semantics; State diagrams can describe the possible states of an object as events occur. Each diagram usually represents objects of a single class and tracks the different states of its objects through the system [Fowler and Scott, 2000]

4. IDEF Techniques (IDEF1x):

IDEF (Integration Definition) is a family of modeling languages in the field of systems and software engineering. They cover a range of uses from function modeling to information, simulation, object-oriented analysis and design and knowledge acquisition. These "definition languages" have become standard modeling techniques.

Specifically, the initial (and most-widely recognized) languages are IDEF0, which is a functional modeling language building on SADT, and IDEF1, which addresses information models; an adaptation of IDEF1, called IDEF1X, was subsequently created to address database design issues.

2.7 linking System Models and Business Process Models:

According to [Phalp, 1998], "The CAP framework for business process modeling" Business process modeling is an area of work that is increasingly used in conjunction with software development. For example, many development methods note the importance of strategic or business modeling, typically as a prerequisite to analysis. In addition, Systems Engineering for Business Process Change suggests the need to model the business process in maintaining and evolving existing (legacy) Systems.

In order to model business processes, one needs to consider what notations are most suitable, and what methods to adopt. However, the most appropriate notation typically depends on a number of contextual issues, the purpose of the modeling, the audience for the models and so on. Furthermore, this context changes with the progress of the modeling. Hence, the modeler needs guidance about appropriate approaches at different points in the modeling programme. [Phalp, 1998] introduces a framework for business process modeling that provides such guidance without prescribing particular notations. This is achieved by describing business process modeling in terms of three iterative and generic categories or phases: Capture, Analysis and Presentation. The paper shows how different kinds of notational approaches can be used within these categories, discussing the choices available to the modeler. The (CAP) framework is generally applicable, and is illustrated both by a simple theoretical example, and by examples from industrial business process modeling.

[Giaglis, 2001] "A taxonomy of business process modeling and information system modeling techniques" stated that modeling always has been at the core of both organizational design and information systems (IS) Development. Models enables decision makers to filter out the irrelevant complexities of the real world, so that efforts can be directed towards the most important parts of the system under study. However, both business analyst and IS professionals may find it difficult to navigate through a maze of theoretical paradigms, methodological approaches, and representational formalisms that have been proposed for both (BPM) and information system modeling (ISM), [Giaglis, 2001] seeks to put an order to this chaos by proposing an evaluation framework and novel taxonomy of BPM and ISM techniques. These findings coupled with a detailed review of BPM and ISM techniques can assist decision makers in evaluating and selecting suitable modeling techniques, depending on the characteristic and requirements of individual projects.

[Bergholtz et al, 2002] "Process Models and Business Models – a Unified Framework" Said that in e-Commerce, there are two fundamental types of models, business models and process models. A business model is concerned with value exchanges among business partners, while a process model focuses on Operational and procedural aspects of business communication. Thus, a business model defines the what in an e-Commerce system, while a process model defines the how. The purpose of their paper is to analyse the contents of business models and process models and to show how they can be integrated. They are using ebXML as a conceptual and notational framework for their approach. The theoretical foundations of our approach are based on the Language/Action approach and REA. They illustrate how their approach can be used to facilitate integration, process specification and process pattern interpretation.

[Dijkman and Joosten, 2002] "Deriving Use Case Diagrams from Business Process Models" introduce a technique to simplify requirements capture. The technique can be used to derive functional requirements, specified in the form of UML use case diagrams, from existing business process models. Because use case diagrams have to be constructed by performing interviews, and business process models usually are available in a company, use case diagrams can be produced more quickly when derived from business process models. The use case diagrams that result from applying the technique specify a software system that provides automated support for the original business processes. They also show how the technique was successfully evaluated in practice.

[Odeh and Kamm's, 2003] "Bridging the Gap between Business Models and System Models" their approach is 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 his / her understanding of organizational contexts. Hence, the same business process may lead to different number of use cases with different responsibilities in the different contexts. Their 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.

[Issa and Aburub, 2007] "Performing Early Feasibility Studies of Software Development Projects Using Business Process Models" used coloured RAD to derive use cases using business process modeling. This enriches the derived use cases with, to some extent, matured information about automated interactions and activities.

In this research a new approach will be developed to bridge the gap between system model and business process modeling.

CHAPTER THREE

Research Methodology

3.1 Introduction:

In this chapter the researcher will explain the research methods that will be used to investigate the derivation of system models using business process models and how these methods will serve the researcher goal by explaining each method component and why it's important in general.

The research methods that have been used to investigate the derivation of system models using business process models are case study, process modelling, interview, and system modelling

3.2 Case Study

Case study is a research method used to investigate in detail and intensively an individual or an institution in a unique setting or situation [Salkind , 2003].

Many authors have categorised case studies into several types. [Collis and Hussey, 2003] categorised case studies, in an organisational context, into three forms as follows:

explanatory case study: where existing theory is used to understand and explain what is happening;

descriptive case study: where the objective is restricted to describing current practice;

experimental case study: where the research evaluates the implementation of new procedures and techniques in an organisation;

illustrative case study: where the research attempts to illustrate new and possibly innovative practices adopted by particular companies.

In the present research, the descriptive and experimental forms of case study have been used, to explore in depth a particular business process (the healthcare process in Jordan).

3.3 Interviews

The interview is a qualitative research method in which the interviewer collects data through asking a set of prepared questions for interviewees to answer. According to [Thomas, 2003]interviews typically involve questions asked orally by researchers and answered orally by individuals. Interviews traditionally are conducted face-to-face and one-to-one, where the interviewer speaks directly with one interviewee at a time. By using technology, interviews can be conducted using different methods: on the Internet, the interviewer may send typed questions via the computer network to the interviewees who answer in typed form; interviews can also be conducted on the telephone.

Interviewing is an efficient way to collect information and to understand an organisation's processes, according to [Thomas, 2003]: "interviews are efficient for collecting information about people's knowledge, personal backgrounds, and opinions". This research has conducted semi-structured and non-structured interviews with staff members of the case study organisation.

3.4 Process Modeling:

Role Activity Diagrams (RADs) are a useful way of describing processes. They are valuable in documenting processes as they are now, and as they might be in the future.

Role Activity Diagrams are a reasonably simple diagramming technique. It is not difficult to learn how to draw them and it is not difficult for most people to interpret them.

The creation of Role Activity Diagrams relies upon an ability to scope a study, to decide on the level of interest and to determine the boundaries of each role in a diagram. This is where the real skill of using Role Activity Diagrams comes in. In reality, the reader will rely upon experience gained through using Role Activity Diagrams in projects. This experience will teach how Role Activity Diagrams can best be used, and what they are most useful for.

The Role Activity Diagram in figure (3.1)

This diagram shows two roles: customer and cashier for a retail outlet, e.g. a supermarket. Having entered the customer may choose to select goods or leave. Once goods have been selected the customer must make a payment before leaving, however, a number of selections can be made before paying. On payment, there is an interaction with the cashier. This is only possible if there is an instance of a signed-on cashier for the customer to interact with.

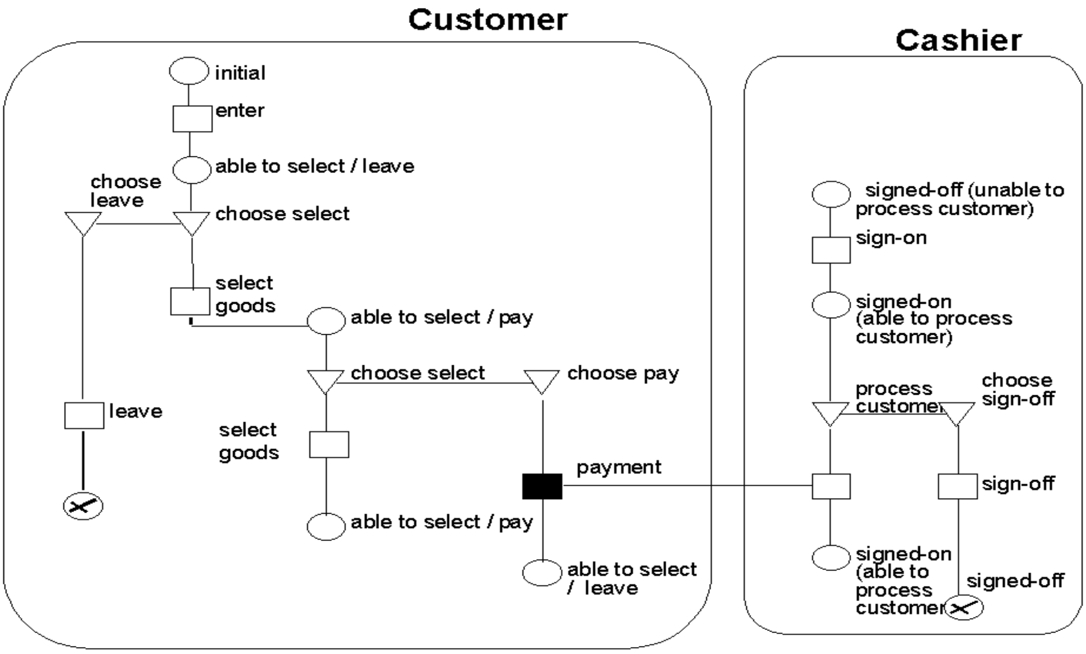
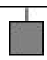
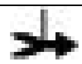
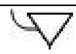
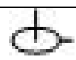


Figure 3.1 supermarket RAD

In the table below is RAD notations:

Name	Notation
Role	
Activity in Role	
Trigger	
Interaction	
Case Refinement	
State Description	

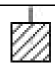
A Driver in an interaction	

Table 3.1 RAD Notations

You cannot intelligently use a RAD approach with multiple complex systems interacting over a centralized generic data model. Since there is no written Analysis Document to refer to, requirements can be easily forgotten. If an omitted requirement can't be easily added to a completed system, fundamental changes to the data model will be required adding both time and cost to the entire project.

Because we are not capturing requirements in written form, but instead, are relying on informal notes and quick application development, developers will invariably forget things that users have told them. This means that users may have to be re-interviewed about the same material multiple times. The development team may discuss a particular requirement with users and inadvertently omit it from the system. Also, users will occasionally make decisions about requirements and limitations that they are willing to accept that they will later deny.

In this thesis the researcher uses RAD as a modelling tool for its easy to understand as many managers can interact and understand it simply by looking at its notations and interactions between the roles and for the reason that RAD is a great way to show a process and provide ease and flexibility for modelling business behaviour.

3.5 System Modelling:

Data Flow Diagram (DFD) is a process-oriented graphical representation of an application system. In the words of [Hoffer etal, 1999], a DFD "is a picture of the movement of data between external entities and the processes and data stores within a system."

The data flows of the opened process are connected in the new diagram to the process related to the opened process. Vertices, and the flows and objects connected to them, are transferred with the flows that are connected to the decomposed process.

In figure 3.2 an example of level 0 DFD for Mail Order Company as the customer make an order by many ways. the order goes to the process order transaction process that handle the order, then it give a feedback to the customer about the order state and update each of the (order file, customer file, and also product detail file), that gives information to the process shipment process about the order and product detail, which gives the customer the shipping notice and invoice. After that the order is ready to be calculated and so the information about the customer and his order bill is send to the payment transaction process for the final payment.

A data flow diagram, like other models, has limitations. It is important to be aware of these limitations. Otherwise, it is easy to misunderstand data flow diagrams by making invalid inferences or by trying to read into data flow diagram meanings which were not intended.

A data flow diagram does not show flow of control. It is not a flowchart. When alternative outputs may result from decisions within a transformation; a data flow diagram shows only the alternative outputs, not the decisions.

A data flow diagram does not show details linking inputs and outputs within a transformation. It shows only all possible inputs and outputs for each transformation in the system.

A data flow diagram does not show time. This is the best rule to guide a novice in understanding data flow diagrams. A data flow diagram does show relationships which are closely related to time, but it is better not to think of them in temporal terms.

It may show delay, but the duration is not shown. We may find it helpful to think of a data store as a time-delayed repository of data, and data flow as transmitted instantaneously between transformations or between transformations and data stores. But we may also think of data store as a collection of data with the same structure, each member of the collection being unique.

The question for why we use data flow diagram in this thesis is because a traditional approach hasn't been used yet to derive a system model from business process model yet and DFD also deals with process as its main event and given a set of requirements in DFD can translate them into a model more accurately. The DFD shows the processing steps as data flows throw a system using a simple intuitive notation that customer can understand and its shows end to end processing of data

The table below shows the DFD notations:

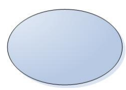
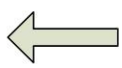
Name	Notation
Process	
Data Store	
Data Flow	
Entity	

Table 3.2 DFD Notations

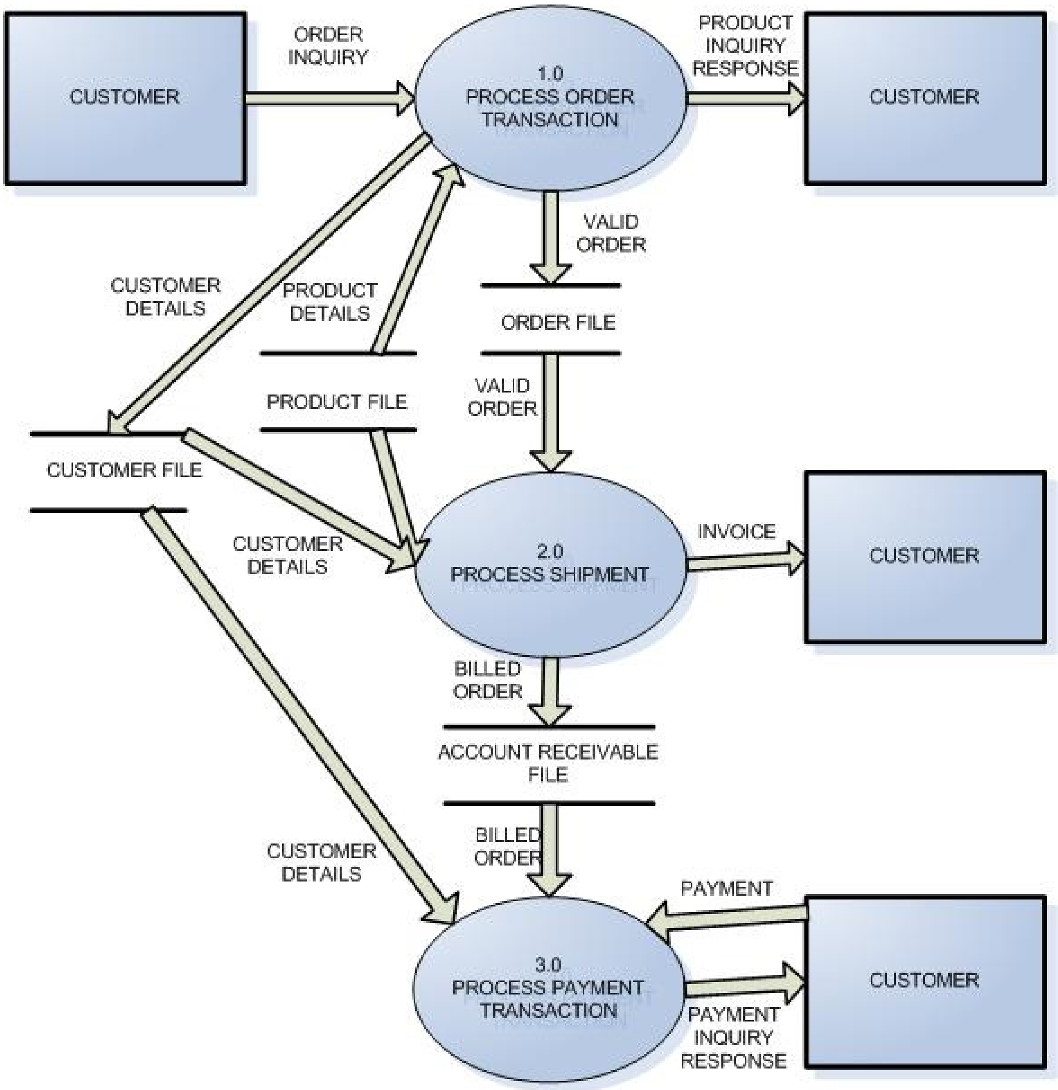


Figure 3.2 an example of level 0 DFD for Mail Order Company

CHAPTER FOUR

New Approach of Linking Business Process Models and System Models (Linking Approach)

4.1 Introduction

This chapter aims at developing new approach to link the system models and business process models. The approach includes six phases as follows: developing a business process model using RAD, determine automated activities, discover initial system requirements, determine system specification, develop DFD based on system requirements, and DFD refinement

4.2 New Approach

The new approach including the following phases:

Phase 1: Developing a Business Process Model Using RAD

Modeling a process presents significant challenges. Modeling should ensure consistency and thoroughness in capturing relevant information so that both business analysts and the developers can understand the business requirements. Those are captured in the model. During modeling, alternatives and exceptions to standard procedures must be captured, in addition to normal operations. Professionals with different scopes of interest and expertise can build process models to meet a wide range of business objectives.

An analyst constructs a business process (BP) model based on requirements collected from discussions with the business requirements owners. These requirements are gathered using appropriate methods such as interviews. In our case; the analyst uses these requirements, together with an analysis of the existing processes, as input to construct the model. Existing process models could be used for analysis of the existing processes or to create new process models by modifying existing models rather than recreating it from scratch.

Modelling starts with the structuring of a BP into sub-processes. Subsequent analysis is done on each sub-process of interest to identify the components, services, data inputs and outputs, policies, and measurements.

The task of modeling a process consists of defining the details of a business process flow and modeling all the data, resources, and other elements that the flow uses. A business process is composed of process steps that are normally connected through control flows, and these control flows connect activities with decision nodes. A decision node holds the business rules (transition conditions) that are evaluated to decide what path in the process should be followed. Modeling includes the decomposition of a BP into sub-processes and adding required process elements to the model. The analyst can apply Pre-existing modeling artifacts.

In this thesis we use RAD to model a Process as explained below:

First we determine the roles: A set of activities which when taken together achieves some particular goal.

Note that roles may or may not be equivalent to organisational job titles. For example, 'Manager' might be a role. And also 'Financial Performance Review' – it all depends on the scope and focus of the model being created.

It is sometimes useful to use gerunds to denote roles. This is because it helps some people to emphasise the activities taking place rather than just the person or job-title involved. Hence, 'Manager' might become 'Managing,' whilst 'Financial Performance Review' could be 'Financial Performance Reviewing.'

Roles are drawn as sets of boxes:

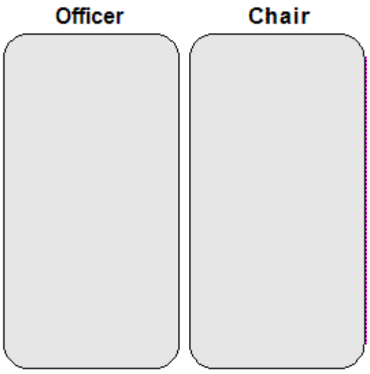


Figure 4.1. Role bodies

In Figure 4.1, we start to construct a Role Activity Diagram by focusing on two roles 'Officer' and 'Chair.'

Second we determine Activitiess: The items of work that people do.
Activities are represented as boxes within a role.

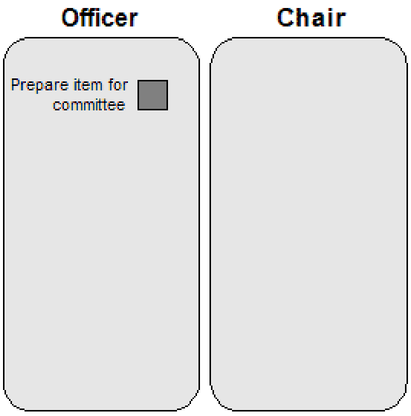


Figure 4.2. An Activity

In Figure 4.2 we see that an activity has been added to one of the role bodies. This is 'Prepare item for committee' and is performed within the role 'Officer.'

Role Activity Diagrams are state diagrams. The vertical lines linking activities denote the different states of the role. Formally then it can be understood that completion of the activity 'Prepare item for committee' represents a transition to a further state in which 'Make Item Available' is undertaken. This is called activity ordering

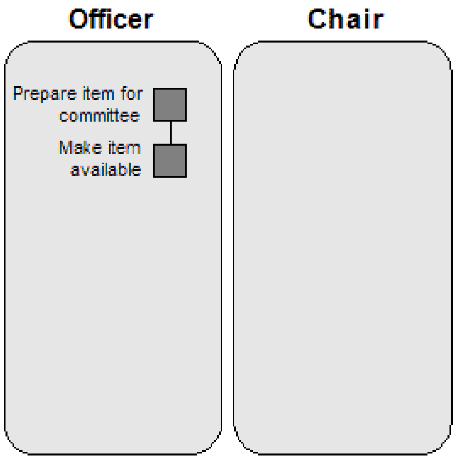


Figure 4.3 Ordering

It is highly unlikely that the ordering of activities is as precise in the real world as it is in a Role Activity Diagram. People work in a complex manner often tackling more than one task at a time. This is always a difficulty with formalised models of work practice. The modeller will be wise to pay heed to these restrictions.

Third we determined interactions: The point at which a role interacts with another role in order to fulfil an objective.

Interactions are shown by a horizontal line linking two boxes. They designate synchronous behaviour between the roles. They are the easiest to understand when two people are involved. For example, we can imagine that when the Officer carries out 'Make item available,' there is an interaction with another role that is carried out by someone acting as a Chair.

In other cases we can understand them as points of synchronisation between roles. This is not obvious: we might think of two functions undertaken by the same person but between which there is a logical interaction.

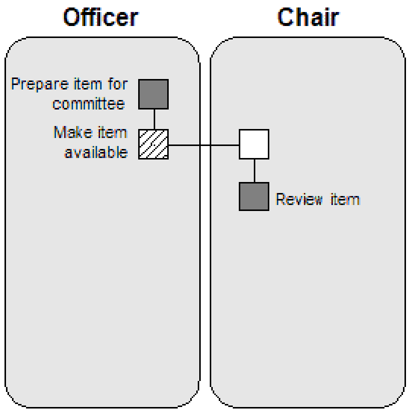


Figure 4.4 Interaction

The conditions under which different activities take place are called Choices. These are shown as linked circles. In the diagram below it can be seen how the 'Chair' reviews the item for inclusion. If it is ok, no action is taken. If it is not ok, the Chair reports the fact to the officer.

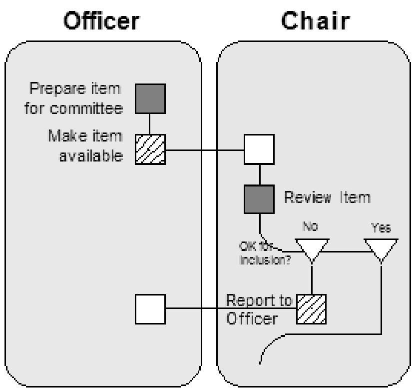


Figure 4.5 Choice

And we also should determine:

Cardinality: (the numerical relationship between role types)

Cardinality is difficult in Role Activity Diagrams. Interpreting the semantics of the previous diagram in a precise way reveals how this can manifest. It can be seen that two roles interact. The 'Chair' role receives an item for the committee from the 'Officer' role. The Chair reviews it and may reject it. It will then read other materials for the meeting and prepare notes. So, what happened to all the other items for the committee? The diagram shows an interaction between two roles. The implication is that the committee considers only one item.

It is therefore helpful to show the cardinality occurring across the interaction. This is done simply by denoting a relationship of the following sorts:

*(One to one (default)

*(One to many (1:m)

*(Many to one (m:1)

Of course, there are many officers who submit items to the chair of any given committee. Therefore, there is a many- to- one relationship between role types. This is denoted by marking the role titles appropriately.

Part Refinement: shows activities in sub-threads to the main thread. This means that the ordering of these sub-threads is not significant.

Linked triangles indicate that activities below them can be undertaken in any order. Therefore, 'Prepare notes for meeting' can be undertaken before, at the same time as, or after 'Read other materials.'

In the terminology of Role Activity Diagrams this is a 'Part Refinement.' What it really says is that the vertical is a state that can be refined into further parts. It is then valid for the actions in the separate threads to trigger in any sequence but they must all complete before the main action thread is re-joined.

State: It is sometimes useful to identify and label particular states in a Role Activity Diagram.

A state is marked explicitly by concentric circles. This is done to promote clarity and to show iteration

Iteration: A return to a previous state of the role.Iteration can be shown in two ways:

*State markers can show iteration

*An arrow which linking two states in a role

Waiting: External events or inputs are sometimes needed before work can continue.

Role Deliverables: Identifying the concrete outputs of a role is sometimes useful to provide a highlighted description of the outputs of roles. This enables the reader to quickly peruse the diagram, focusing upon how each role works on the deliverables described

This kind of highlighted description can be provided below each role body. The text should be phrased in a way that helps the reader to quickly assimilate the dynamics of the process being described.

Phase 2: Determine automated activities

Activity: A description of a piece of work that forms one logical step within a process. An activity may be a manual activity, which does not support computer automation, or automated activity. An automated activity requires human and/or machine resources(s) to support process execution.

***Manual Activity:** An activity within a business process which is not capable of automation and hence lies outside the scope of a workflow management system. Such activities may be included within a process definition, for example to support modeling of the process, but do not form part of a resulting workflow.

***Automated Activity:** An activity which is capable of computer automation using a workflow management system to manage the activity during execution of the business process of which it forms a part.

* A process definition generally consists of many process activities which are logically related in terms of their contribution to the overall realisation of the business process.

* Manual activities may form part of a business process and be included within its associated process definition, but do not form part of the automated workflow resulting from the computer supported execution of the process.

* An activity may therefore be categorised as "manual", or "automated".

We might use RAD models, in process redesign or improvement, to distinguish between the activities that are partly or fully automated, and the activity carried out by people. To do so, we color the automated activities in a color, and the non-automated in another color to make it easier to differentiate between them. In figure 4. 6 we show the part of officer and chairmen interaction, with green boxes to represent automated activities, and red boxes to represent non-automated ones. Just by dividing up the role into activities, and considering the nature of the activities, or by visual inspection of the diagram to see the proportion of automated activities, we may be able to select activities suitable for automation, or come to the view that a particular role is under- or over-automated. The RAD model at least gives us a starting point for such deliberations. Appropriate annotation of the models, in addition, could enable us to bring under considerations aspects such as resources, timings, volumes and methods, which are not covered in the basic notation.

If the organization proceeds to automate some activities, it may then be beneficial to remodel the process in question to make a clear separation between those activities carried out by computer and those remaining under human control. It may, in any case, be useful to model management control more explicitly to show with greater precision how data validation or monitoring is accomplished [Issa and Aburub, 2007].

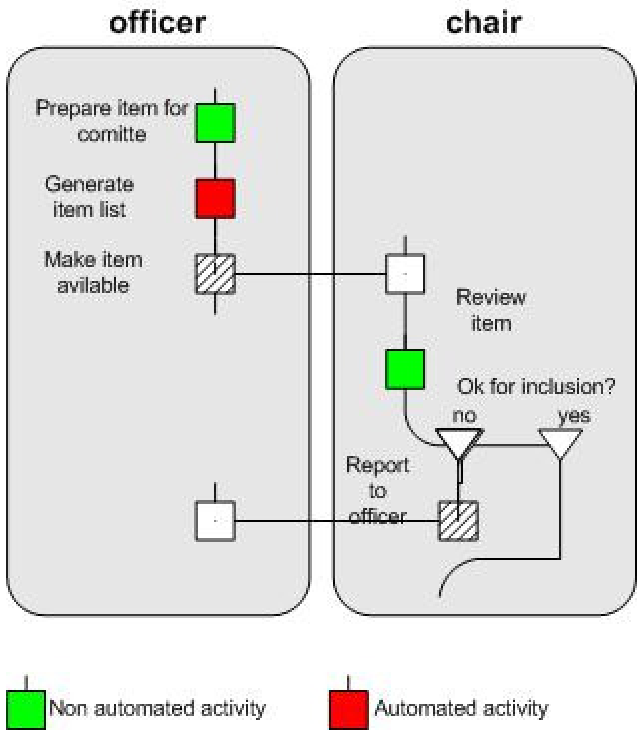


Figure 4.6 officer and chairman part interaction

Phase 3: Discover initial system requirements

, software engineering and systems engineeringRequirements analysis in encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting , such as beneficiaries or users.stakeholders of the various requirements Requirements analysis is critical to the success of a development project. [Steve, 1996]Requirements must be actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

The system requirements are classified into:

Functional requirements Which are observable tasks or processes that must be performed by the system under development. For example, a functional requirement of a stock trading system "must update and remember stock prices"; for a web search engine, "must accurately parse Boolean queries"; for an automated teller machine, "must process withdrawals and dispense cash to the customer".

In figure 4.6 we could say that the functional requirement for it is generate an item list because it's an automated activity.

Non-functional requirements are qualities or standards that the system under development must have or comply with. However they are not tasks that are automated by the system. Example of non-functional requirements for a system include: a" system which must be built for a total installed cost of $1,050,000.00"; "system must run on Windows Server 2003"; "system must be secured against Trojan attacks".

It is important to note that these kinds of requirements always exist, regardless of the approach or method used to manage software development. A software development methodology helps to identify, document, and realize the requirements.In this thesis the researcher will focus on the functional requirements [Available from: http://wiki.answers.com/Q/What_are_functional_and_non-functional_requirements_of_system_development_methodology&src=anTT)]

Regardless of system scope specification of the required functionality is essential in order to be able to design the workflow or computational process components. The functionality of a system is the essential purpose of the system and can be defined by a series of individual functions.

Functional requirements represent activities we need to achieve using computer system. They can be synthesized using RAD model by checking every activity in the RAD model and selecting automated activities as these activities represent what customers need to achieve using computer system.

Phase 4: Determine system specifications

This phase determine how each requirements can be achieved in order to model them using DFD.

How to specify what system should do is written in a document, the basic document that is needed is called a Requirements Specification. In other words a description of what you want the system to do. This document may also be called a Business Needs Specification. The Requirements Specification should generally not be written in computer terms or contain assumptions about how the system should be written unless these are part of the requirements.

For example many users do not initially supply us with sufficient information upon which to make a reasonable estimate of how big a project is and of how much is going to cost. In many cases, they may not appreciate what information is required.

A requirements specification is normally produced in response to a user requirements specification or other expression of requirements, and is then used as the basis for system design. The system requirements specification typically differs from the expression of requirements in both scope and precision that may cover both the envisaged system and the environment in which it will operate, but may leave many broad concepts unrefined. Traditionally, requirements specifications take the form of natural-language documents.

Each entry in the list of requirements should be specified in a format similar to the following:

1. A description of the requirement: This will probably be a couple of lines explaining what you want to achieve. Make sure that you provide sufficient detail. E.g. Produce a report of spend/department/year on demand with the user selecting the Department and the financial year required (note that you would also need to define how your financial year is calculated!).
2. How important this requirement is (essential, preferred, nice to have, not essential, etc).
3. Any known design/implementation issues relating to this requirement.
4. Does this requirement interact with other requirements?
5. Anything else affecting this requirement (e.g. only authorised personnel should have access.

And also describe a set of scenarios that illustrate, from the user's perspective, what will be experienced when using the system under various situations. E.g. this would describe how different users would use the system and the typical sequence of actions that each would take.

Phase 5: Develop DFD based on system requirements

Data flow diagrams allow to model how data flow through an information system, the relationships among the data flows and how data come to be stored at specific locations. They also show the processes that change or transform data. These diagrams concentrate on the movement of data between processes.

Data flow diagramming rules (and naming conventions):

* The inputs to a process are different from the outputs of that process (because processes are meant to transform data and not only pass them over).

* Objects on a DFD have unique names (if two flows have the same name they must represent identical set of data).

There are four types of DFDs used in the systems development process:

* Current physical – process labels include an identification of the technology used to process the data, data flows and data stores are also labeled with the names of the actual physical media on which data flow or in which data is stored

* Current logical – here physical aspects of the system are removed as much as possible so that the current system is reduced to its essence, to the data and the processes that transform them.

* New logical – differs from the previous one by having additional functions, obsolete functions removed and inefficient flows reorganized

* New physical – represents the physical implementation of the new system (reflects the analyst's decision about which system functions will be automated which is manual).

It is good to create each of these DFDs but current physical DFDs should not be so detailed and we should concentrate on the DFDs for the new logical system.

DFDs, apart from being process modeling tool can also be used for a process called gap analysis, which are discrepancies between two or more sets of DFDs, representing two or more states of an information system, or discrepancies within a single DFD (redundant data flows, data captured but not used by the system, data updated in more than one location etc).

Once the system requirements are well understood and detailed models of the requirements are completed, it is important to establish which of the functional requirements are most critical.

Processes cannot have outputs only, cannot only have inputs, and must have a verb phrase label. Data can move to a data store from only a process, not from another data store or an outside source. Similarly, data can be moved to only an outside destination or to another data store by a process. Data to and from external sources and destinations can be moved by only processes. Data flows move in one direction only. Both branches of a forked or a joined data flow must represent the same data. A data flow cannot return to the process from which it originated.

Sources and destination are always outside of the system being considered. They are of interest to the system being considered only because they represent sources of data coming into the system and destinations for data leaving the system. If any data processing occurs inside a source or destination, it should be of no interest to the system being modeled. If the processing is of interest, however, or if the identified source/destinations have several inputs and outputs to and from the rest of the system, it may be better considered as an internal process.

Logical data flow diagrams help you better understand the essence of the system, the data and the processes that transform them, regardless of actual physical form. Further, the new logical data flow diagrams can then show any additional functionality necessary in the new system, to indicate which, if any, obsolete components have been eliminated, and any changes in the logical flow of data between system components, including different data stores.

Phase 6: DFD refinement

This phase aims to adjust the resulted DFD model in order to make sure that the final model will meet the customer requirements. The refinements may be performed using different methods as follows:

Decomposition is the iterative process by which a system description is broken down into finer and finer detail, creating a set of diagrams in which one process on a given diagram is explained in greater detail on a lower–level diagram.

The conservation of inputs and outputs is called balancing (process must have the same number of inputs and outputs on every level of decomposition).

The principle of balancing and the goal of keeping a DFD as simple as possible lead to four additional advanced rules for drawing DFDs:

* A composite data flow on one level can be split into component data flows at the next level.

* Inputs to a process must be sufficient to produce the outputs from the process.

* At the lowest level of DFDs new data flows may be added to represent data that are transmitted under exceptional conditions (these data flows typically represent error messages).

* To avoid having data flow lines cross each other, you may repeat data stores or sources on a DFD (special additional symbols are used for repeated elements)

The highest level DFD is called a context diagram. It represents the system as a single process, with all the related entities and the data flows in and out of the system. The next level diagram, called a level-0, decomposes the one process from the context diagram into between two to nine high-level processes. Each process in a level-0 diagram can be decomposed if necessary. Each resulting diagram is called a level-1. Processes in a level-1 diagram should be decomposed, each resulting diagram would be called a level-2 diagram. Each of these processes would be decomposed on a level-3 diagram, and so on.

You can stop decomposing a DFD when the following six conditions are met:

(1) Each process is a single decision or calculation or a single database operation, such as retrieve, update, create, delete, or read.

(2) Each data store represents data about a single entity, such as a customer, employee, product, or order.

(3) The system user does not care to see any more detail, or when you and other analysts have documented sufficient detail to do subsequent systems development tasks.

(4) Every data flow does not need to be split further to show that different data are handled in different ways.

(5) You believe that you have shown each business form or transaction, computer screen, and report as a single data flow.

(6) You believe there is a separate process for each choice on all lowest–level menu options for the system.

DFD consistency is the extent to which information contained on one level of a set of nested data flow diagrams is also included on other levels. Balancing errors are one type of consistency violation mentioned in the textbook. For instance, a payment data flow that appears on a level-1 diagram, but not on the level-0 diagram, is a consistency violation.

The phases are summarized as shown in figure 4.7

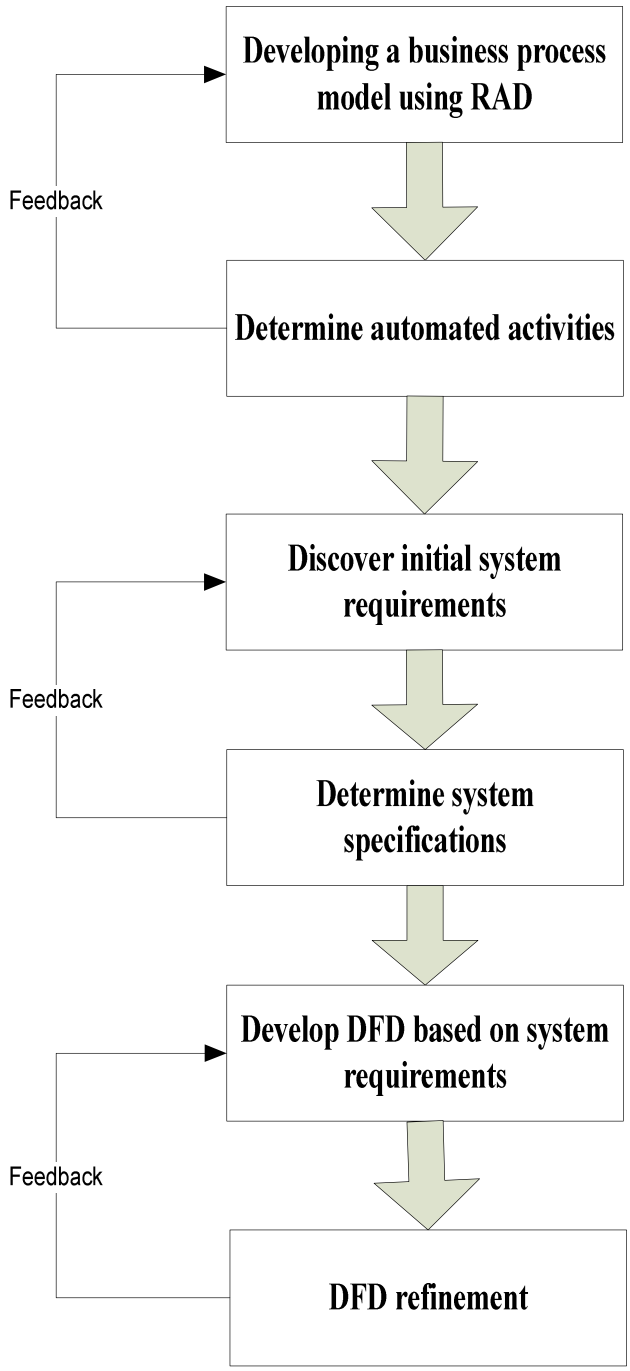


Figure 4.7 New Approach of linking business process models and system models

4.3. Jordan Cancer Registry Case Study

It's a data collection for cancer types, which identify each cancer type and monitor the growth of a cancer, plus evaluating its treatment process.

The main four roles :

1. Jordan cancer registry (JCR)
2. Laboratories
3. Registrars
4. Health care sector

When JCR receives forms from laboratory and registrar , it starts filling data from the lab form to JCR form , and the form from registrar confirms the medical record of the patient to allow JCR to proceed on .

Next step JCR compares primary cancer sites with its ICD-O code (International classification diseases for oncology).Then if there is any missing data the form will be return to both registrar and laboratory, but if none is missing the JCR checks if the patient is in its database.

In case the patient does not exist, JCR adds a new record in the database, and then add the Primary cancer site to JCR database.

On the other hand if the patient record exists, JCR checks if the primary cancer site exists in the data base, if not JCR adds the primary cancer site to the database.

If yes, JCR checks if there is any additional data and adds them to the patient file and modifies it, then JCR makes a backup to the form in its archive.

Finally the JCR analyzes the database that has been collected through the past year, and generates statistical reports and sends them to all health care sectors.

We made a business process model using RAD, as shown in Figure (4.9).

In general, both laboratories and registrar send forms to JCR which in turn make the necessary process that we will explain later in detail. Then the JCR sends the final outcome to the health care sector as shown on Figure (4.8).

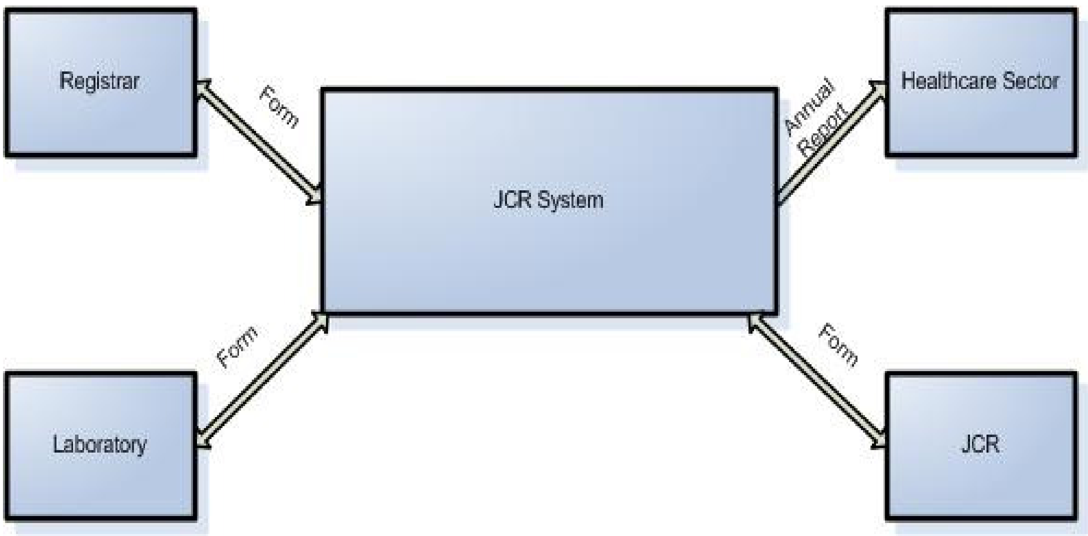


Figure 4.8 context diagram of JCR

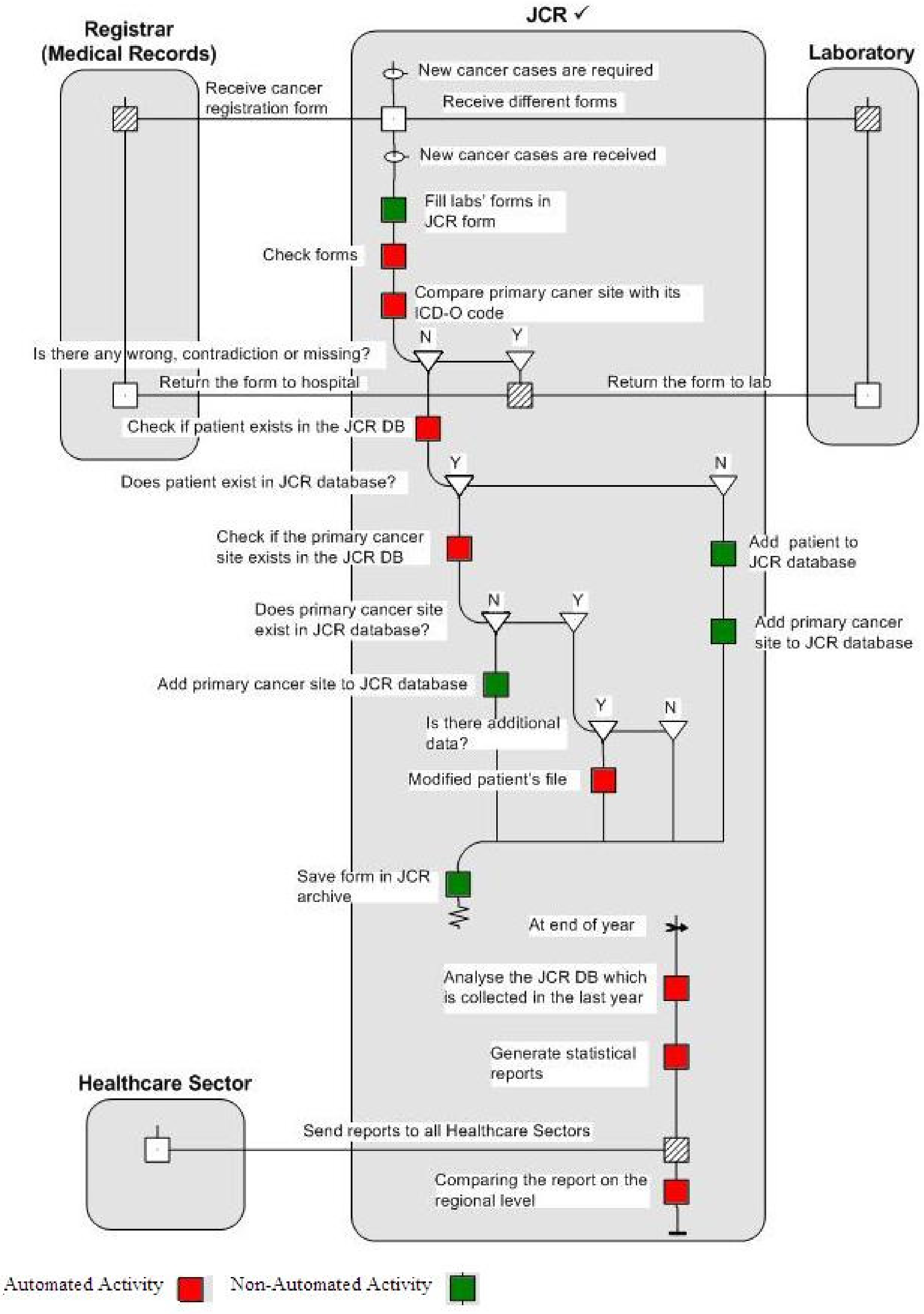


Figure (4.9) JCR RAD

4.3.1 Functional Requirements of JCR

1. Check forms: JCR process checks automatically for any missing data and then notify the person handling the system if there is any.

2. Check primary cancer site with its icd-o code
Check if the cancer type is correct by matching it automatically with the international icd-o code.

3. Add New Cancer Patient: That checks if the patient is new or old one by looking through the JCR database if he is new he add him to DB if old he read the data from the DB.

4. Add New Cancer Site: If the patient is old it checks if he have primary cancer site by checking through JCR database for any record for cancer type.

5. Analyze JCR DB: After the end of each year the JCR analyzes its own DB in terms of number of dead people, number of cured patients, number of each cancer type... etc.

6. Generate statistical report: After analyzing, the DB JCR then generates statistical report on how the cancer is related with the environment and also how cancer is related with food habits…etc

7. Comparing report on regional level
After getting reports from healthcare sectors the JCR then compare the reports automatically on regional level to see the percentage of cancer in each region.

After determining the functional requirements for the JCR process we can derive a system model using DFD that is version 1 before refinement as shown in figure 4.10and we made version 2 after the refinement to meet the user needs as shown in figure 4.11.

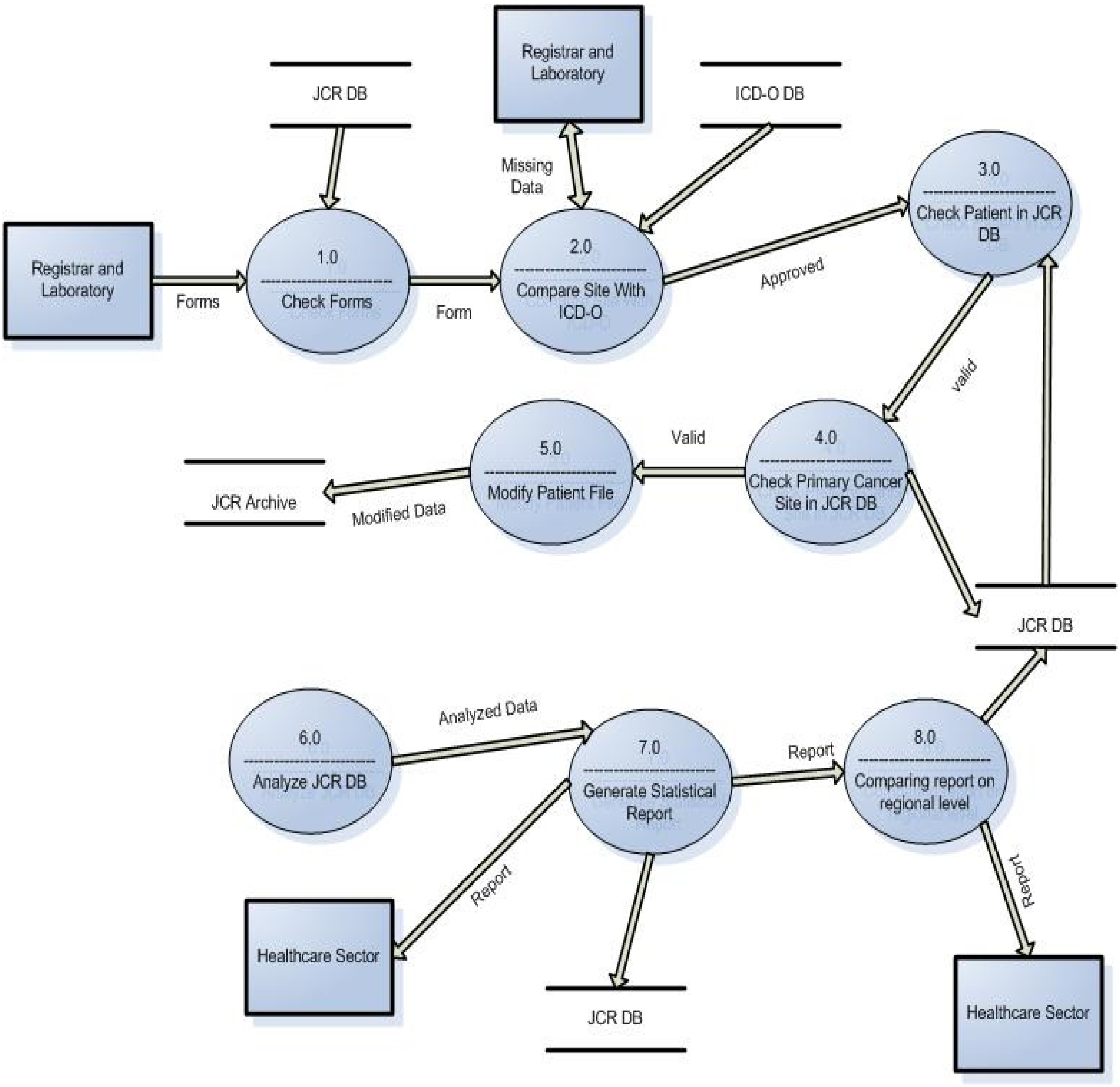


Figure 4.10 JCR DFD Version 1

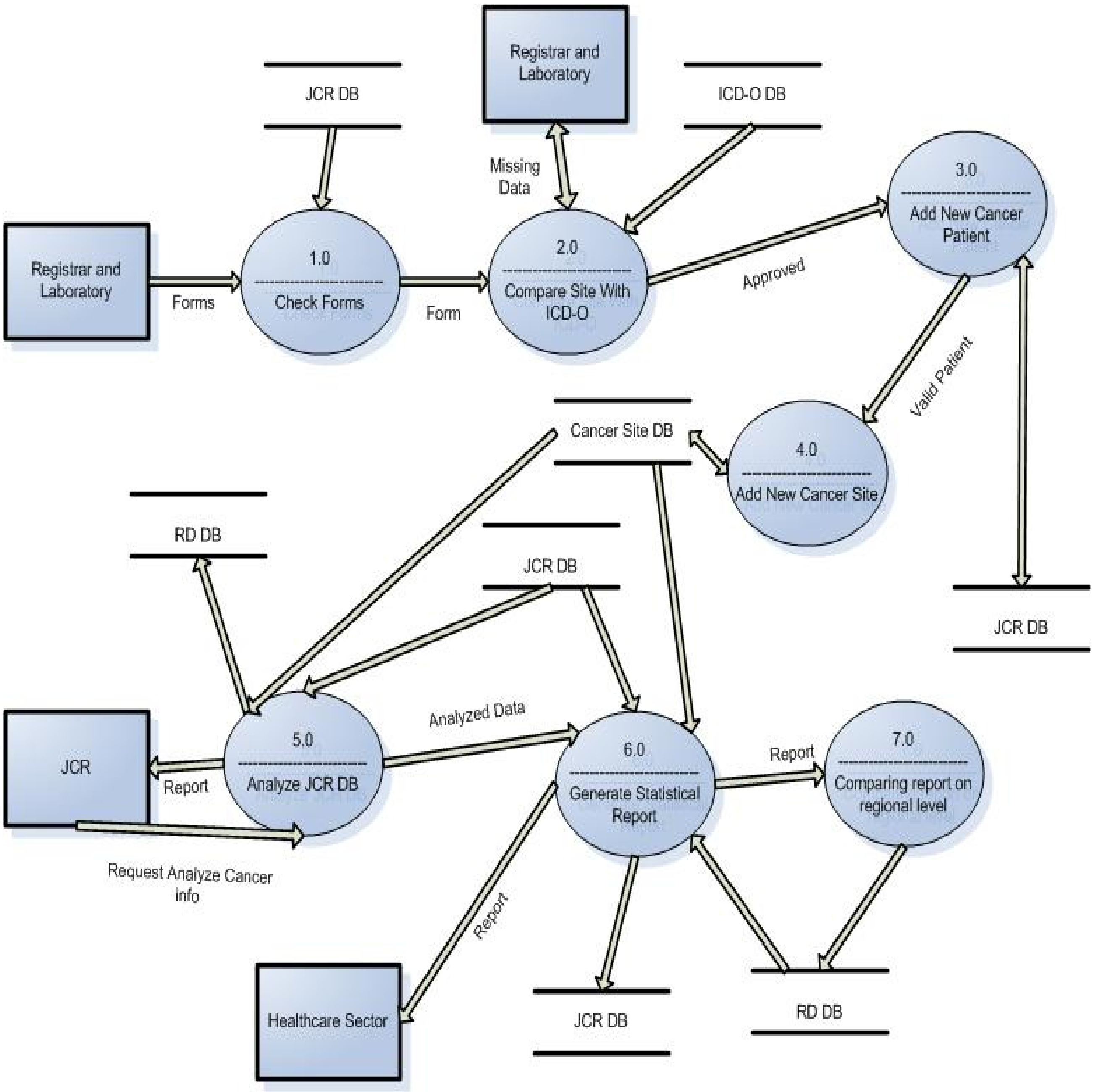


Figure 4.11 JCR DFD Version 2

4.4. Critical Analysis

Every system development got a SDLC when developing it and it contains five phases: data gathering and planning, analysis, design, implementation and maintenance, and its known that the most time consuming phase of them all is the data gathering.

So what if we present a methodology that helps the analyses in shortening the time spent in collecting the requirements needed for the analysis phase to create a system model.

Our approach simply use the business process model that most organizations have if not all to capture the requirements we need and to derive a system model from that business process model.

The methodology we used is determining the automated activities which contain machine or computer involvement and less likely a human interaction that are shown in the business process model that the organization provided when determining these automated activities the functional requirements of the system is shown for the analyses and therefore he/she can simply derive a system model out from the business process model.

By using this approach we succeed in accomplishing two things:

1. We succeed in bridging a gap between system models and business process models because in the past the existence of software systems was not necessary

tied to the business processes in another word the systems models and business processes.

2. We made the time consuming problem spent in the data gathering phase less by providing a preliminary layout of a model using DFD.

Chapter Five

Conclusion and Future Work

The researcher presents the conclusions of this work and an idea for future work.

5.1 Conclusion

This thesis introduced a new methodology called linking methodology that aims to bridge the gap between system model and business process model. Particularly, it aims to transform RAD model into data flow diagram. The methodology includes six phases namely: developing a business process model using RAD, determine automate activities, discover initial system requirements, determine how each system requirement can be achieved, develop DFD based on system requirements, and DFD refinement. A lot of work is performed in both fields system and business process modeling to study how can be derived system models form business process models, We used the concept of automated and non-automated activities in order to synthesis system requirements which then can be used to derive the system models from business process models .

5.2 Future Work

The methodology that has been developed in this research has been investigated and critically evaluated using the cancer registration process in Jordan. Furthermore, the methodology can be investigated using different applications and tests domains to confirm the validity and usefulness of the methodology.

In Chapter 4, the new approach has revealed how can be linked between business models and system models. A useful development in relation to this approach might be if some method or tool could be developed for applying the approach, which has in this research been done informally and manually.

References

[1]Aiken, P., Muntz, A., Richards, R., (1994). DoD Legacy Systems: Reverse Engineering Data Requirements. Communications of the ACM; Vol. 37 (5), p2641.

[2] Anni Tsai et al. (2006). "EPC Workflow Model to WIFA Model Conversion". In: 2006 IEEE International Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, pp. 2758-2763

[3] Bart-Jan Hommes (2004). The Evaluation of Business Process Modeling Techniques. TU Delft. p.137.

[4] Bent Flyvbjerg, (2006) "Five Misunderstandings about Case Study Research." Qualitative Inquiry, vol. 12, no. 2, pp. 219-245

[5] Cedric G. Tyler, Stephen R. Baker (2007). Business Genetics. John Wiley and Sons, 2007. ISBN 0470066547. p. xiii

[6] Collis, J., Hussey, R., (2003). Business research: a practical guide for undergraduate and postgraduate students. Basingstoke: Palgrave Macmillan. ISBN: 0333983254.

[7] Curtis w. Kellner , M.I ,and over J. (1992), "Process Modelling" Communication of the ACM VOL.35, NO 9 .

[8] Ethan Rasiel and Paul Friga ,(2002) published by McGraw-Hill, from the book "The McKinsey Mind,"

[9] Fort Belvor, (2001) Systems Engineering Fundamentals. Defense Acquisition University Press, 2001.

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

[11] Gil Regev, Ilia Bider, Alain Wegmann,(2002) Defining business process flexibility with the help of invariants. Software Process: Improvement and Practice

[12] G.M. Nijssen, (2008) PNA Group 2008

[13] Henry J. Johansson et al. (1993). Business Process Reengineering: BreakPoint Strategies for Market Dominance. John Wiley & Sons

[14] Hindle, J., (1997). Process improvement and information management. Health Manpower Management, 23(5), pp 184-186.

[15] Issa, A., Aburub, F. (2007). Performing Early Feasibility Studies of Software Development Projects Using Business Process Models.

[16] James Taylor (2007) Smart (Enough) Systems http://www.ebizq.net/blogs/decision_management/2007/05/some_business_process_definiti.php.

[17] Martin E. Modell , (2007) A Professional's Guide to Systems Analysis,

[18] Martin Fowler, Kendall Scott:(2000) UML Distilled Addison-Wesley

[19] McConnell, Steve (1996). Rapid Development: Taming Wild Software Schedules (1st ed.). Redmond, WA: Microsoft Press. ISBN 1-55615-900-5. http://www.stevemcconnell.com/.

[20] Michael Havey (2005) Essential Business Process Modeling.

[21] Michael Hammer and James Champy (1993). Reengineering the Corporation: A Manifesto for Business Revolution, Harper Business.

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

[23] Ould M.,(2005) Business process management, a rigorous approach. Swindon: British Computer Society.

[24] Peter SG Carter, Available from: (http://www.metarasa.com/management/process/the-importance-of-business-processes/), [Accessed 25/06/2009]).

[25] Phalp, K.,(1998). The CAP framework for business process modelling. Information and Software Technology, 40, pp 731 – 744.

[26] Rayson, P., Garside, R., and Sawyer, P. (2008). Assisting Requirements Recovery fromLegacy Documents. Available from: http://www.comp.lancs.ac.uk/computing/users/paul/publications/rgs00_sebpc.pdf. [Accessed 02/02/2008].

[27] Richard C. Simpson (2004), An XML Representation for Crew Procedures, Final Report NASA Faculty Fellowship Program (Johnson Space Center).

[28] Robert K. Yin , (2002). Case Study Research. Design and Methods. Third Edition. Applied social research method series Volume 5. Sage Publications. California. ISBN 0-7619-2553-8.

[29] Rolstadas, A.,(1995). Performance management: a business process benchmarking approach. Chapman & Hall. ISBN: 0412605600.

[30] Rummler & Brache (1995). Improving Performance: How to manage the white space on the organizational chart. Jossey-Bass, San Francisco.

[31] Salkind, N., (2003). Exploring research. Upper Saddle River, N.J.; UK: Pearson Education International: Prentice Hall. ISBN: 0131229168.

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

[33] Shepard, Jon; Robert W. Greene (2003). Sociology and You. Ohio: Glencoe McGraw-Hill. pp. A-22. ISBN 0078285763. http://www.glencoe.com/catalog/index.php/program?c=1675&s=21309&p=4213&parent=4526.

[34] Shin, N., Jemella, D.,(2002). Business process reengineering and performance improvement: the case of chase Manhattan bank. Business Process Management Journal, 8 (4), pp. 351 – 363.

[35] Soy, Susan K. (1997). The case study as a research method. Unpublished paper, University of Texas at Austin.

[36] Stephen A (2006) Process Modeling Notations and Workflow Patterns,. White of IBM Corporation .

[37] Tam, A., Chu, L., Sculli, D., 2001. Business process modelling in small to medium sized enterprise. Industrial Management and Data Systems, 101(4), pp 144-152.

[38] Thomas Davenport (1993). Process Innovation: Reengineering work through information technology. Harvard Business School Press, Boston.

[39] Thomas, R. M., (2003). Blending qualitative & quantitative research methods in theses and dissertations. Thousand Oaks, Calif., London: Corwin Press: Sage. ISBN: 0761939318.

[40]USAID, Available from: (http://www.qaproject.org/methods/resysmod.html), [Accessed 18/04/2009]).

[41] Warboys, B., Kawalek, P., Robertson, I., Greenwood, M., (1999). Business information systems: a process approach. Maidenhead: McGraw-Hill. ISBN: 0077094646.

[42] Wiki answers, Available from: (http://wiki.answers.com/Q/What_are_functional_and_non-functional_requirements_of_system_development_methodology&src=ansTT) [Accessed 13/04/2009].

[43] Wikipedia, Available from: (http://en.wikipedia.org/wiki/Business_process), [Accessed 1/04/2009]).

[44] Wikipedia, Available from: (http://en.wikipedia.org/wiki/Business_process_modeling), [Accessed 20/03/2009]).