A GOAL QUESTION METRIC BASED TOOL FOR GOAL ORIENTED BUSINESS PROCESS MODELING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

BAŞAK MERAL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

SEPTEMBER 2014

Approval of the thesis:

A GOAL QUESTION METRIC BASED TOOL FOR GOAL ORIENTED BUSINESS PROCESS MODELING

submitted by **BAŞAK MERAL** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Adnan Yazıcı _____
Head of Department, **Computer Engineering**

Assoc. Prof. Dr. Halit Oğuztüzün _____
Supervisor, **Computer Engineering Department, METU**

Examining Committee Members:

Prof. Dr. Ali Hikmet Doğru
Computer Engineering Department, METU _____

Assoc. Prof. Dr. Halit Oğuztüzün
Computer Engineering Department, METU _____

Prof. Dr. Nihan Kesim Çiçekli
Computer Engineering Department, METU _____

Assoc. Prof. Dr. Pınar Karagöz
Computer Engineering Department, METU _____

Dr. Selçuk Köprü
Teknoloji Yazılımevi _____

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: BAŞAK MERAL

Signature :

ABSTRACT

A GOAL QUESTION METRIC BASED TOOL FOR GOAL ORIENTED BUSINESS PROCESS MODELING

Meral, Başak
M.S., Department of Computer Engineering
Supervisor : Assoc. Prof. Dr. Halit Oğuztüzün

September 2014, 78 pages

In this work, a new visual and functional tool that is built to support new approaches for functional and non-functional parts of business process modeling is presented. The new tool, which is also capable of keeping numerical relationships between goals, is developed following an approach that helps to correlate business processes with goals. These goals and numerical relationships between goals make up a directed acyclic graph and they are represented as a Structured Equation Model graph. In order to obtain improved processes, Goal Question Metric approach is combined with BPMN and Goal Hierarchy. This makes the tool capable of supporting business process improvements.

Keywords: Goal, Goal Hierarchy, BPMN, Business Process Improvement, Goal Question Metric

ÖZ

HEDEF ODAKLI İŞ SÜREÇLERİ MODELLEME İÇİN HEDEF-SORU-ÖLÇEV TABANLI BİR ARAÇ

Meral, Başak
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi : Doç. Dr. Halit Oğuztüzün

Eylül 2014 , 78 sayfa

Bu çalışmada, iş süreçleri modellemenin işlevsel olan ve olmayan açılarına getirilen yeni yaklaşımlar için üretilen görsel ve işlevsel bir araç anlatılmaktadır. Süreçlerin hedeflerle ilişkilendirilmesini sağlayan bir yöntem için yeni bir araç geliştirilmiş ve bu aracın aynı zamanda hedefler arasındaki sayısal ilişkileri de kurması sağlanmıştır. Geliştirilmiş süreçler elde edebilmek için, Hedef-Soru-Ölçev yaklaşımı BPMN ve Hedef hiyerarşisiyle ilişkilendirilmiş, bu sayede iş süreçleri iyileştirme yapabilen bir araç ortaya konmuştur.

Anahtar Kelimeler: Hedef, Hedef Hiyerarşisi, BPMN, İş Süreçleri İyileştirme, Hedef Soru Ölçev

To people who make me who I am

ACKNOWLEDGMENTS

It would have been impossible to write this thesis for me if I was not surrounded by the awesome people I have in my life. While I would like to mention some of the names here, this is far from a complete list.

To begin with, I would like to express my deepest graditude to my supervisor Assoc. Prof. Dr. Halit Oğuztüzün for his guidance and effort in the preparation of this thesis. I thank him for providing me the motivation and discipline in every way possible.

I would like to thank my darling Ruşen Aktaş for his continuous support and patience. He has been the absolute source of joy and light in my life.

I am very lucky to have my beloved family always supporting me. I would like to thank my dear mother Meral Meral and my dear father Ekrem Meral for their love, sacrifices and care. I also thank the other part of my family, Can, Merve and İdil who always express their believe in me despite the thousands of kilometers of distance.

I thank my dearest friends Gizem Taşkesen, Begüm Seval, Buket Vural, Deniz Karatay and many other people I could not mention in this short page.

I thank Cahit Güngör, PhD student of my supervisor from Computer Engineering Department for the guidance and data used in this thesis.

Finally, I would like to thank my professors, colleagues and friends for their mere existence in my life. In addition, I appreciate the encouragement and insightful comments provided by the jury members, Prof. Dr. Ali Hikmet Doğru, Prof. Dr Nihan Kesim Çiçekli, Assoc. Prof. Dr. Pınar Karagöz and Dr. Selçuk Köprü.

TABLE OF CONTENTS

APPENDICES

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

LIST OF ABBREVIATIONS

BPMN	Business Process Modeling Notation
GQM	Goal Question Metric
BPM	Business Process Management
SEM	Structural Equation Model
BPD	Business Process Diagram
PDE	Plug-in Development Environment
BPS	Business Process Simulation
BSC	Balanced Scorecard

CHAPTER 1

INTRODUCTION

Business process improvement is a popular and growing field in today's world and receives attention from different kinds of work units. Professionals are directly interested in the latest improvements in the field, as well as academic personnels, since it is now proven that it is critical for an organization's management.

Business process improvement is critical for Business-IT alignment and for this reason, it is also vital for organizations. The importance attached to business process improvement by companies and their managers can be observed clearly in the survey prepared by Harvey Nash [1, 2, 3], in which, priorities of CIO's in various areas are ranked in terms of percentages. This study summarizes how companies in the world's leading countries pay attention to business process improvement.

Once considering that business process improvement is critical for both industry and academy, many people work on bringing new solutions or methodologies to the problem. One of those solutions is covered in this thesis. This thesis presents a new tool that follows a goal-oriented business process improvement approach with various steps. A goal-oriented approach named GoalDAG is introduced and it is related with BPMN processes. An Eclipse extension is developed to support GoalDAG and BPMN integration. For measurable process elements, Goal Question Metric approach is used and its relation with BPMN is provided. Also, a new transformation methodology is used for the data which is obtained from processes and it is used to transform processes for a specific stategic analysis tool named Tetrad.

1.1 Thesis Goal

The main goal of this study is to provide a tool support for Goal oriented business process improvement. To achieve this goal, a Goal Hierarchy methodology is combined with BPMN by providing a new Eclipse based tool. In order to analyze to processes, a new approach is developed that helps to prepare Goal Hierarchy for strategic analysis. For a complete cycle of business process improvement, Goal Question Metric approach is combined with BPMN within the tool. In this way, measurements are used for Goal oriented business process improvement.

1.2 Thesis Outline

The remainder of this thesis contains the following chapters.

Chapter 2 : Gives general information about technologies used in this study including BPMN, Eclipse BPMN Editor, Eclipse Plugin Development and Tetrad.

Chapter 3 : Summarizes the related work from the literature which is related with this thesis. In this chapter, some theoretical background is also introduced. Also, this chapter includes a comparison of previous work with this thesis.

Chapter 4 : Describes the new approaches presented in this study.

Chapter 5 : Provides steps of a case study which consists of the approaches that are introduced in this thesis.

Chapter 6 : Concludes the study.

Also, it presents the appendix.

CHAPTER 2

ENABLING TECHNOLOGIES

This section gives information about the technologies that form the basis of this thesis, which are used in order to achieve the goals of the study. For a better understanding of the following sections, it is crucial for the reader to have an idea about these technologies. In this chapter, some information and technical details about BPMN, Eclipse BPMN2 Modeler and Tetrad will be given.

2.1 BPMN

2.1.1 History and Definition of BPMN

Business Process Modeling Notation (BPMN) is a standard, which is developed by Business Process Management Initiative [4]. The latest major version of BPMN is BPMN 2.0 [5] , which is also used in this thesis work.

A business process (BP) is a set of one or more linked procedures or activities executed following a predefined order which collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional roles or relationships [6]. Business Process Modeling is the activity of representing business processes. It helps to see the current status of the processes of an enterprise, as well as analysing and improving those processes in the future. It is performed by managers and business analysts since they are trying to improve the entire quality and efficiency of the organization.

The primary goal of BPMN is to provide a notation that is readily understandable by

business users, ranging from the business analysts who sketch the initial drafts of the processes to the technical developers who are responsible for actually implementing the processes, and finally to the business staff who are deploying and monitoring such processes [7].

In BPMN 2.0, the newest major version of BPMN, there are some noticeable differences in some aspects. Formalizing the execution semantics of BPMN elements and, most importantly, since it is critical for this thesis, defining an extensibility mechanism for both process model extensions and graphical extensions is some of them. New models have been defined for wider modeling interactions. Furthermore, both XMI and XSD based meta-model and diagram definition model has been defined. Adding new Data objects and putting them to a separate category was also a new feature of the final major version. With the contributions of these features, BPMN 2.0 has become a de-facto standard in the area of Business Process Modeling.

2.1.2 Elements of BPMN

In order to represent a business process, BPMN defines a graphical notation which is Business Process Diagram(BPD). It is based on a flowcharting technique fitted for creating graphical models. Business Process Diagram is made up of a set of graphical elements and those elements have four main categories. These are:

- Flow Objects
- Connecting Objects
- Swimlanes
- Artifacts

Flow objects have three main types. These are *Events*, *Activities* and *Gateways*.

Events are represented by circles. An event signifies that something happens during the course of a business process. There are three types of events which are based on the time they affect the flow. These types are *Start*, *Intermediate* and *End* events.

Activities are represented by rounded-corner rectangles. They are generic terms which are performed by the company and can be atomic or non-atomic. Types of Activities are *Tasks* and *Sub-Processes*. Sub-Processes are distinguished from Tasks by a plus sign in the bottom center.

Gateways are represented by a diamond shape and are used to control the divergence and convergence of the Sequence Flow.

Connecting Objects provide the function of connecting the Flow Objects together in a diagram to create a structure of a business process. There are three types of Connecting Objects. *Sequence Flow* is represented by a solid line and solid arrowhead and it is used to show the order of the activities within the process. *Message Flow* is represented by a dashed line and open arrowhead and is used to show the flow of messages between two separate participants that send and receive them. *Association* is represented by a dotted line and a line arrowhead and used to associate Artifacts with flow objects.

Swimlanes are the mechanisms used to organise activities. They divide activities into separate categories in order to show different functionalities and responsibilities. There are two types of swimlanes. *Pools* represent participants of the process. *Lanes* are sub-partitions within the Pools. They help to extend the length of the Pool both vertically and horizontally.

Artifacts are the objects that can be added to the Business Process Diagram with no restrictions. They do not have a direct affect on the Sequence Flow. There are three types of artifacts. *Data Objects* are mechanisms to show the required data by the activities. *Groups* are used for documentation and analysis purposes. *Annotations* are mechanisms that help modeler to provide additional text information about the process.

2.2 Eclipse BPMN2 Modeler

Eclipse BPMN2 Modeler[8] is an open-source Eclipse Project based on Graphiti. It is compatible with BPMN 2.0 specification and it uses Eclipse BPMN 2.0 Metamodel[9].

It is a graphical modeling tool and it allows the creation and editing of BPMN diagrams.

As mentioned in the Eclipse BPMN2 Modeler User Guide[10] the model is complex and very detailed. For this reason, the BPMN2 Modeler UI surfaces only the most commonly used elements. Eclipse BPMN2 Modeler supports the specification partially, like most BPMN execution engines in use. However, it can be easily customized to a larger extent to target any spec-compliant runtime with User Preference settings and/or specialized extension plug-ins.

Eclipse BPMN2 Modeler uses Graphiti [11] and it is compatible with different releases of Eclipse Modeling tools. Installation steps of Eclipse BPMN2 Modeler is described in the User Guide[10] and since those steps does not require any programming knowledge, this makes the Modeler suitable for professionals from different areas.

Eclipse BPMN2 Modeler has all the features of a Graphiti editor:

- the Drawing Canvas in the main area of the editor window,
- Tool Palette on the right of the editor which contains BPMN elements categorized according to their types,
- Property sheets and an Outline Viewer.

The Modeler is capable of validating the diagram according to the BPMN Metamodel. It is also capable of exporting diagrams in different image formats and sizes. A screenshot of Eclipse BPMN2 Modeler is presented in Figure2.1.

2.3 Eclipse Plug-in Development

Eclipse community offers a Plug-in Development Environment(PDE) [12] for developing Eclipse plug-ins. As mentioned in Plug-in Developlment Environment Guide [13] Plug-in Development Environment ensures developers to develop, create, test, debug, build and deploy Eclipse plug-ins. It has three main components:

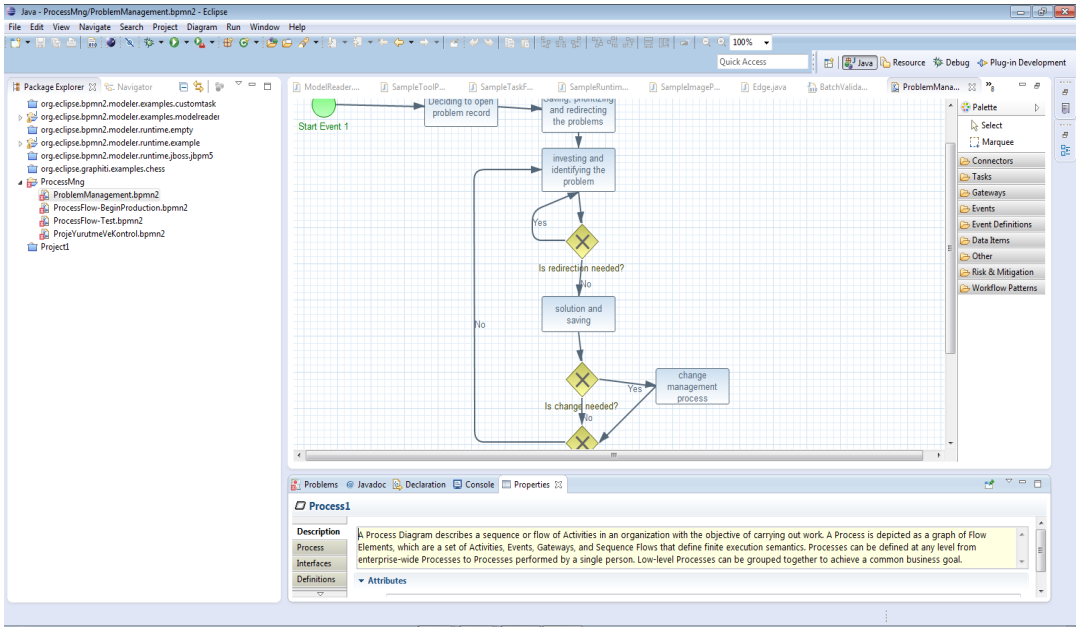

Figure 2.1: A screenshot of the Eclipse BPMN2 Editor.

- *UI*, set of models, editors and tools,
- *API Tools*, in order to assist API documentation,
- *Build*, ant based tools and script for automated processes.

For plug-in development, Eclipse offers two mechanisms; *extensions* and *extension points*. In Plug-in Development Environment Guide[13], they describe those concepts as follows: There is a simple metaphor for defining extensions and extension points, which is electrical outlets. The extension point corresponds to the socket or the outlet and the extension corresponds to the plug, or light bulb that connects to it. When a plug-in is designed to allow others to customize some parts of the functionality, extension points should be declared. Also, there are rules that are defined by extension points which are combinations of XML markup and Java interfaces and extensions should conform to those rules.

All Eclipse projects allow developers to customize and extend the functionalities of the project with the use of extension and extension point mechanisms. Since Eclipse BPMN2 Modeler is also an Eclipse plug-in project, it can be extended and modified easily.

2.4 Tetrad

Tetrad is a program that creates, simulates data from, estimates, tests, predicts with, and searches for causal and statistical models [14]. Program aims to provide sophisticated statistical methods with a friendly interface. Usage of the program requires no programming knowledge and only basic statistical knowledge.

Tetrad provides *exploration* and *discovery* algorithms as principled search algorithms. Those algorithms has the ability to calculate predictions of the effects of interventions or experiments based on a model. Those search procedures are guaranteed to converge to correct information about the true structure in the large sample limit, provided that structure and the sample data satisfy various commonly made assumptions.

Tetrad is limited to models of categorical data and to linear models(Structured Equation Models [15]). It describes causal models in three stages:

i. a picture, representing a directed graph specifying hypothetical causal relations among the variables,

ii. a specification of the family of probability distributions and kinds of parameters associated with the graphical model,

iii. a specification of the numerical values of those parameters.

With the methods provided by Tetrad, it is possible to perform analysis of Structured Equational Models.

CHAPTER 3

BACKGROUND AND RELATED WORK

Business Process Management (BPM) is a wide and popular area, which is defined as supporting business processes using methods,techniques and software to design, enact, control and analyze operational processes involving humans, organizations, applications, documents and other sources of information [16]. This area receives intense attention from researchers and companies for years and now, this attention is shifting towards the area of improving business processes [17]. Companies aim to improve their performance by improving their processes. In this improvement cycle, design and redesign phases have an important role. In the redesign phase, concepts like previous experiences of processes, measurable outputs of existing processes and duration are becoming important [18]. Every company has various alternatives to change the processes and they aim to replace the current one with the best alternative.

There are various kinds of techniques for business process improvement. One common technique for business process improvement is Business Process Simulation(BPS). This method helps to understand, analyze and design the processes. It helps to evaluate and compare the redesigned processes. With the use of simulation, impact of a process design on process performance can be estimated quantitatively. Since the choices are supported quantitatively, it is likely to make the best decisions with simulation. In [19], steps of the business process simulation is described and it is mentioned that the key point is to change the flow of work and collecting results.

Instead of changing the flow of work and collecting results of simulations, another approach come up with the idea of goals for improving the processes. This idea is also the ancestor of our methodology. Our methodology also uses the idea of goals

for business process improvement. Below in this chapter, details of the goal based methodology will be explained.

3.1 Goal Hierarchy Approach Used In This Study

As mentioned above, there are many techniques for business process improvement and goal oriented business process improvement is one of them. This study uses a special goal oriented approach and for a better understanding, it is crucial to explain the basis theoretical steps of the approach.

This study uses a goal-hierarchy model, called GoalDAG which is proposed by Cahit Güngör[20]. This model is an acyclic digraph (DAG) with weighted edges and relation functions. The weights of the edges are called contribution weights. DAG is a graph which does not contain loops, multiple arcs and directed cycles [21]. On the other hand, GoalDAG, the DAG used in this study is an inverted DAG where the direction of the nodes is from dependent to the dependee objective.

The formal definition of GoalDAG is as follows;

$$GoalDAG =< V, E >$$

$$where$$

V is the set of nodes in G and E is the set of edges, where each node is a representation of a BSC metric. The relations between BSC metrics are depicted as edges, from dependee to dependent, in accordance with Strategy Map hierarchy.

Measurement; is a function which evaluates the value of the node representing a BSC metric at any given time t.

$$m(i,t); i \in V, t \in Date$$

Relation Function; is a function that represents how two BSC objective are correlated with each other.

$$rf(i,j); i,j \in V$$

Contribution Weight; is a value that represents the degree of contribution of a BSC objective to the parent BSC objective, with respect to other BSC objectives which are also contributing to the same parent BSC objective.

$$cw(i,j); i,j \in V$$

$$0 \leq cw(i,j) \leq 1$$

$$\forall i \in V : \sum_{j \in edge(i,j)} cw(i,j) \leq 1$$

Since rf is a function, it can represent a wide range of correlations between two goals. In this context, it is generally used to represent a linear relation of two metrics, however, relation function can be adopted according to the most fit function, such as logarithmic, polynomial, exponential and especially natural logarithmic.

In order to understand GoalDAG representation deeply, Balanced Scorecard (BSC) and Strategy Map concepts should also be explained.

Balanced Scorecard [22] is a concept introduced by Kaplan and Norton in 1992. Before those years, performance evaluations were usually dependent on traditional financial measures. They worked well for the industrial area but they are not compatible with up-to-date companies. Also, another group of managers and academic researchers came up with the idea of focusing on improving operational measures. Although this is a new point of view, this idea restricts managers to choose between financial and operational measures. Kaplan and Norton introduces their new concept, Balanced Scorecard to solve this confusion. Balanced scorecard is a set of measures that gives top managers a fast but comprehensive view of the business [22]. It includes financial measures and comlements them with operational ones. It provides answers to questions from four different important perspectives. These are customer perspective, internal perspective, innovation and learning perspective, and financial perspective. They also limit the number of measures.

Balanced scorecard puts strategy and vision instead of control at the center and that approach makes it fitted for many companies. It focuses on reaching the pre-defined

goals but in this approach, a manager does not tell the employee how to arrive the destination. By combining those perspectives, balanced scorecard keeps companies moving forward instead of backward.

Strategy Map [23] concept was also introduced by Kaplan and Norton in 1996. It is an improved version of balanced scorecard and it adds some new features to the existing one. It is claimed that traditional measurements are compliant with a pre-defined plan. However, balanced scorecard aims to help align individual, organizational and cross-departmental initiatives to achieve a common goal. To have a clear use of balanced scorecard, organization's long term strategical plans must be provided by the measures. Although the multiplicity of measures of balanced scorecard seems confusing, when four different perspectives have well defined strategic measures, it makes the concept desirable for the outcomes.

Defining strategic goals for each four perspective and linking them to achieve a common goal is important for the following reasons. It helps to see the vision of the entire organization. It helps employees to see their contribution in organizational success. It focuses change efforts and finally by putting cause-and-effect relations, it improves organization's self-learning.

3.2 Studies Which Combine Goal Concept and Process Improvement

In his study [24] Bleistein mentions that, there are number of studies which uses Goal concept for organizational strategy [25, 26], however, those studies do not have strong effects on business strategy analysis. Bleistein proposes an analysis technique that helps to validate and verify the business strategy, however, in his study, the concept of linking Goals and processes remaing lacking. Although Bleistein is one step forward then others, there are still no study which uses relations between goals, subgoals and metrics.

There are also other studies which relate the concepts of Goal, BPMN, and business process improvement. Some of them has done significant contributions to the area, whether or not they are exactly parallel to the field of this thesis. In their study, A Goal-Oriented Approach for Variability in BPMN [27] , Santos and Castro introduced

the approach of variability in business process models through goal models. Goal models are used to analyze variability so that processes can be redesigned.

In their study, Extending the EPC and the BPMN With Business Process Goals and Performance Measures [28], Korherr and List introduce performance measures and process goals for BPMN and EPC by extending the metamodels. Although the idea is creative, extending the metamodel has drawbacks such as compatibility. Also they do not provide business process improvements with the study.

Another Goal oriented BPM approach has been introduced by Burmeister and Arnold in their study[29]. They introduce a new model which combine business processes, goals and contexts and this model can be executed by a BDI agent. This method has contributions to process improvement since it provides strong runtime agility, feedback loops and modularity. They do not have a general contribution for process improvement since the main focus is on agility, however, ideas are instructive.

In their study [30], Koliadis and Ghose introduced a new methodology called GoalBPM for relating process models with Goals. In this concept, goals are representing high-level stakeholder goals and their aim is to combine goal-oriented requirements engineering with BPMN. In their study, business process models are designed to satisfy the requirements that are specified by Goals. Processes or goal models can be changed and system can be tested again in order to see the current status. In this way, system can be checked in order to see that it satisfies the specifications. Their study provides a cycle for requirements engineering however it does not provide a method for process improvement.

Another goal oriented requirements engineering approach [31] was introduced by Decreus and Poels. They have extended B-SCP framework which is designed for requirements engineering and developed an Eclipse based B-SCP editor for strategic requirements. They finally transformed BPMN skeletons to their B-SCP model. In this way, they have proposed a naive way to use BPMN in order to support strategic processes. This study has strong contributions in requirements engineering since they combine it with BPMN. However, they are not related to business process improvement. Although their area is not parallel with this thesis, they show a pleasant way of combining Goals and BPMN objects in an Eclipse extension.

Go4Flex [32] is yet another goal-oriented process modeling research project, which has the aim of bringing together business and IT parts by establishing higher-level modelling concepts for workflows. Their main idea is building a strong context perspective of a workflow with goals and goal relationships. In this way, understandability and abstractness of business processes increase since the goals are representing the reasons of executions. In this study, a user-friendly goal oriented tool is produced for increasing the usability of business processes, however they do not contribute many features for process improvement.

3.3 Goal Question Metric and Related Studies

This thesis work aims to propose a new tool for GoalDAG methodology in support of busines process improvement. To achieve that goal, we propose a technique that is based on measurements of process elements. In this methodology, process elements are measured and some analysis are done according to this information. In this measurement step, we decided to use Goal Question Metric [33] approach. The reason of this choice is that, this approach is a common one in the area of software measurement. With this choice, the measurement module of our study fits a standard. In order to understand the approach deeply, some details will be given below.

3.3.1 Explanation of Goal Question Metric Approach

The Goal Question Metric approach is based on three steps [34]. Those steps are, identifying goals for projects or project steps, tracing those goals and providing a framework for interpreting the data. The origin of the approach was a system that is designed for evaluating defects in a NASA project. The approach was used to define and evaluate goals for a specific project in a limited environment. Its use has been expanded to a larger context after some iterations. It is used as a goal setting step in a software development organization.

In the final version of Goal Question Metric approach, a measurement model has been introduced. The model helps to correlate measurement information with goals by using appropriate questions. The model has three levels which are *Conceptual*

Level, Operational Level, Quantitative Level.

First level is the *Conceptual Level* which stands for defining a goal for an object that will be measured. This goal can be defined with respect to different points of view. Objects of measurements can be products, processes and resources. Products can include various kinds of items that are usually deliverable. Processes are usually related to activities that can be measured by time. Resources are the items used by the processes.

The second level is the *Operational Level* in which a set of questions are asked. Those questions are used in order to identify the achievement of a specific goal. According to the decisions given in the conceptual level, questions try to identify the object measurement.

The third level is the *Quantitative Level*. In this level, collected data is associated with the questions. The aim is to answer the questions in a quantitative way. This data can be either objective or subjective. In other words, it can either depend on the object or it can depend on both object and the viewpoint. For example, when the goal is user satisfaction, data could be subjective.

3.3.2 Applications of Goal Question Metric Approach

There are some studies which used Goal Question Metric approach for measurements. Some of them are worth to mention since they bring new solutions to that area. In this part, a brief summary of such studies will be given.

In one of these studies, Victor R. Basili who is the inventor of Goal Question Metrics approach proposes a new mechanism for the paradigm [35]. In this study, Goal Question Metric approach is used to set operational goals for software projects. Goals can be related to tasks from a wide variety of perspectives such as managers or organization itself. In this study, Basili first expresses the GQM paradigm to the user for an introduction, then moves to the application part which can be a guide for people who are interested in the paradigm. It introduces a path for defining the goals. In this path, object of study is defined and questions like why and who are asked. After goals, questions from different areas are asked. They can be logical attributes

like complexity, cost attributes like effort or time and change attributes like errors or faults. In the study, all those goals and questions are expressed and metrics are collected with expressing the reasons of why are they chosen. This study can serve as a simple guideline for people who want to get involved in GQM paradigm. However, it does not provide any innovation for the approach.

Another study which is related to GQM approach is done by L. Lazic and N. Mastorakis in the area of software testing. In their study [36], discuss software test metrics and their use for process improvements in an organization. Although the paper mainly focuses on test metrics, their usage of Goal Question Metrics as a test metric approach is important. At first, they introduct the concept of testing process and they define a new metrics philosophy for software testing. After those steps, they show how to correctly handle those software test metrics. In the core part of the study, they propose a new framework, which uses GQM approach. In the first step of this framework, measurement needs of organization is defined by GQM approach. In the second part, Balanced Scorecard is used to make sure that the coverage of measurements is appropriate for the performance. This study is important since it proposes a new framework which uses GQM with another important concept, BSC. In this way, they try to guarantee the coverage of goals. A good way of measurement is proposed and this technique is used for process improvement. However, since it is in the field of software testing, they do not focus on organizational improvement.

In another study, P. Berander and P. Jonsson used GQM approach for an efficient measurement framework definition [37]. In the study, they claim that the GQM approach has some drawbacks. Their contribution is that, they solve problems such as number of metrics, coverage of all relevant perspectives. They provide a way to categorize the questions so that they ensure completeness. Also, they provide a method for prioritizing the questions so that the number of questions can be reduced. This extended version of GQM approach is applied in two case studies which are about change management and requirements engineering. The results of the case studies show that, their approach is successful for such industrial fields. They introduce new aspects for measurement and they try to ensure that these aspects help to increase the efficiency of business process improvements. This extended version of GQM approach is beneficial for those who want to use the GQM approach in their own business process

improvement methodologies.

3.4 Combination of Business Processes and Measurements

In their literature review [38], Gonzalez and Rubio overviews the studies that focus on process measures. In one of these studies, it is mentioned that, in order to improve the process performance and predictability, measurement activities are critical [39]. In another one of them [40], it is argued that a well-designed business process measurement can influence process performance in a desired direction when measurements are used to control the process efficiency. More mature organizations can be obtained when process measurements are integrated carefully [41]. Those studies have strong arguments on how measurements can be used within processes in order to obtain better processes. However, none of them used standards for processes or measurements such as BPMN or GQM.

In the literature review [38], they also grouped those studies and it is observed that most of the studies in that area are related with measuring the complexity with 44%. Measuring the understandability of the processes follows the complexity with 21%. There are also other criteria such as quality or entropy with lower percentages.

One of studies which combines measurability concept with business processes is introduced by Cardoso [42, 43], which is defined as process complexity. Process complexity is defined as the degree to which processes are difficult to analyze, understand or explain. Control-Flow complexity is also presented for business process analysis. In another group of works [44, 45, 46] a set of measures for evaluating BPMN processes are expressed. They measure the structural properties of processes and those measurements are empirically validated. As a result, some measures are obtained to say that they are related with usability and maintainability of the processes. Although all those studies do not have strong effect on process improvement, they are important since they introduced the idea of measurable processes.

Another approach is proposed by Rolon and Cardoso [47], following their studies on control-flow complexity. In this study, they focus on measuring control-flow complexity. They work on business processes modeled with BPMN and their aim is to

use and validate the control-flow complexity metrics. In the measurement step of their BPMN processes, GQM approach is used. They used GQM template to define research objectives as analyse measures of the structural complexity of BPMs. Although the study has important steps in measuring control-flow complexity, it does not have a strong BPMN-GQM integration.

CHAPTER 4

THE SUPPORTED METHODOLOGY

This study aims to support GoalDAG[20] hierarchy for goal oriented business process modeling. The main goal of this study is, to integrate GoalDAG hierarchy with organization's processes in order to provide a methodology for process improvement. Since BPMN is a common and useful way of representing processes, GoalDAG should be linked to BPMN. This helps us to converge to a solution which involves a new tool which will consist BPMN features as well as GoalDAG hierarchy. To be able to perform improvements, Goal Question Metric approach is followed. The main goal of this study is to be able to perform a full cycle business process improvement. This full cycle will be consist of following steps:

1. Measurement information of process elements should be entered with Goal Question Metric approach.
2. Measurement information should be given to Tetrad for analysis.
3. Results of the analysis should be seen in the editor as a Goal Hierarchy.

A new BPMN extension has been developed in order to form relations between BPMN processes and GoalDAG. Requirements of such a tool will be detailed in this section and features of the tool which are satisfying those requirements will be explained in detail.

A program has been developed in order to prepare data for statistical analysis from the processes in order to be able to analyze and improve the processes. Improvements of the processes and goals can be undertaken according to the results of statistical

analysis.

GQM approach has been correlated with BPMN within the Eclipse Extension in order to perform complete business process improvement. The requirements of this relation are also explained in this chapter which help to cover all aspects of this procedure.

4.1 BPMN Extension To Represent Goal Hierarchy

As mentioned above, the main goal of this study is to perform business process improvements using a Goal hierarchy. Since GoalDAG is chosen as the representation of the goal hierarchy and this GoalDAG will be related to business processes, developing a new tool that will satisfy those requirements became essential. BPMN is chosen since it is the de-facto standard for business process modeling. Eclipse has an easy-to-use and easy-to-extend BPMN editor, Eclipse BPMN2 Editor which was described in chapter 2. Eclipse BPMN2 Editor allows developers to extend the editor to fulfill their own needs. To make a connection between BPMN and GoalDAG, Eclipse BPMN2 Editor was chosen for the development.

In this section, firstly, requirements of the extension will be explained. Secondly, implementation details of the extension will be expressed and finally the final version of the extension will be shown.

4.1.1 Requirements of the BPMN Extension To Represent Goal Hierarchy

In this study, one of the main concerns of this goal oriented business process improvement strategy is to be able to represent the goal hierarchy and business processes in the same editor. When the editor has this capability, a user will be able to relate the goals with the processes. This concern specifies the first requirement of our Eclipse extension. To satisfy this requirement, Eclipse extension should be able to represent the GoalDAG graphically. Also, the extension should be able to relate the goals with the processes.

One of the solutions to represent the GoalDAG in the BPMN2 editor and make relations between processes and GoalDAG is to extend the BPMN metamodel used by

Eclipse BPMN2 Editor. With this solution, new objects and rules can be added to the metamodel and the needs of the extension can be satisfied. Another solution of this problem is, an extension can be developed without extending the metamodel. This solution can consist new BPMN objects that will represent the GoalDAG. However, these objects will be extended version of already existing BPMN objects. In this way, GoalDAG can be represented as a group of Goal and Edge objects. Those Goal and Edge objects will be extensions of BPMN objects. The second solution is chosen for this study since it is the most fitted solution for our needs. The reasons of this choice will also be explained below.

Eclipse BPMN2 Editor has the capability of storing BPMN processes in some formats. These formats are the files that have .bpmn and .bpmn2 extensions. Those formats are the generic formats that are compatible with all BPMN execution engines and other BPMN editors. This makes processes that are produced by Eclipse BPMN2 Editor compatible with other BPMN related tools. Eclipse BPMN2 Editor have this capability since it uses BPMN metamodel that is defined by Object Management Group. When a developer wants to extend the BPMN metamodel and make such an Eclipse extension, Eclipse does not have any restriction. It allows developers and users to extend the editor and metamodel in this way. They assume, some developers or projects might have such requirements so that they can extend the metamodel. In this method, they build a new Eclipse BPMN2 Editor extension and their new tool satisfy their needs. Processes that are produced with this tool can cover their needs. However, since the metamodel is extended, those processes are no longer compatible with other BPMN related tools.

In our solution, those restrictions brings us to solve our problem without extending the BPMN metamodel. In this way, we have chosen to represent GoalDAG with Goal and Edge objects which are extensions of existing BPMN objects. When this choice is made, .bpmn files that are produced by our extension are compatible with all other BPMN tools. This is the consequence of not extending the BPMN metamodel. Since Goal and Edge objects are also BPMN objects, it is possible to store them within the .bpmn files. This choice makes our extension compatible with other BPMN editors and execution engines. In other words, .bpmn files that are produced by Eclipse BPMN2 extension are compatible with other BPMN tools.

Once this choice is made, the next step of this study is making right decisions about how to represent the GoalDAG. GoalDAG will be consist of Goal and Edge objects. Goal and Edge objects will be extension of already existing BPMN objects. They will store information about GoalDAG and, they will also be compatible with all BPMN tools. In this step requirements of Goal and Edge objects are defined clearly in order to make right decisions. Requirements of Goal and Edge objects are explained below.

- Goal objects will keep information about its related flow element.
- Goal objects will not be directly related to the process.
- Goal and Edge objects should be stored in BPMN files.
- Goal objects will be connected to each other with Edge objects.
- Edges will keep information about contribution weights.

4.1.2 Design and Implementation of the BPMN Extension To Represent Goal Hierarchy

In the previous subsection, the requirements of the BPMN Extension was expressed. In order to satisfy those requirements, most fitted BPMN objects that will be ancestors of Goal and Edge objects can be determined.

- As mentioned above, Goal objects should keep extra information about process elements. However those objects should not be directly related to the process flow. The most appropriate solution for this requirement is to extend *TextAnnotation* object. In this way, Goals can be represented in BPMN editors.
- As mentioned above, Edges should connect Goals and they also should be able to keep information about contribution weights. The most appropriate solution for this requirement is to extend *Association* object. In this way, Goals can be connected to each other with Edges and GoalDAG can be generated. Edges can also be represented in BPMN editors.

In order to represent Goals and Edges in Eclipse BPMN2 Editor, an extension has been developed. That tool was built with the use of extension and extension point concepts that are described in chapter 2. Technical details of this extension are provided below.

Eclipse extensions are Eclipse plugin projects. Our BPMN extension is also an Eclipse plugin project, which is specifically designed for our needs. Eclipse plugin projects depend on a main configuration file, which is plugin.xml. Within the plugin.xml extensions and extension points are defined. Those configurations vary according to the requirements. Since Eclipse BPMN2 Modeler is written in Java, extensions should also be written in Java. The key point of the implementation process is that, appropriate Java classes should be implemented which supports the definitions in the plugin.xml file.

Constructing the plugin.xml starts with defining the general package extension point. Generated package extension point for our extension is defined as follows:

```
<extension point=
 "org.eclipse.emf.ecore.generated_package">
<package
class="org.eclipse.bpmn2.modeler.runtime.example.
SampleModel.SampleModelPackage"
genModel="model/SampleModel.genmodel"
uri="http://org.eclipse.bpmn2.modeler.runtime.example">
</package>
</extension>
```

Since our Eclipse extension is a runtime extension, the next step that should be done in the plugin.xml should be defining the runtime extension. In this definition, runtime id and runtime classes are defined. The program can be capable of starting correctly through this definition.

```
<extension
point="org.eclipse.bpmn2.modeler.runtime">
```

```
<runtime name=
 "Sample Business Process Engine" versions="1.0"
id="org.eclipse.bpmn2.modeler.runtime.example"
class="org.eclipse.bpmn2.modeler.runtime.example.
   SampleRuntimeExtension"
description="BPMN2 Modeler customizations for a
   sample runtime engine">
</runtime>
```

Runtime id is also defined for the program to find the correct classes to initialize.

```
<model runtimeId=
        "org.eclipse.bpmn2.modeler.runtime.example"
 uri="http://org.eclipse.bpmn2.modeler.runtime.example"
 resourceFactory="org.eclipse.bpmn2.modeler.core.model.
 Bpmn2ModelerResourceFactoryImpl">
</model>
```

After the configurations listed above, the Eclipse extension is able to start with little modifications. For this extension to be able to start, an Activator class which implements *BundleActivator* is implemented. Also, SampleRuntimeExtension class which implements *IBpmn2RuntimeExtension* is also implemented. This configuration makes the extension have a similar runtime with existing BPMN extension. Those classes have variables and methods for the program to start correctly. After those variables are initialized and methods are implemented properly, the extension can be run without any problem. This extension has no difference than existing Eclipse BPMN2 Modeler. If the program needs modifications at the runtime phase, those classes can be modified accordingly. However, since our application does not need such modifications, we have not done modifications in those classes.

After constructing a runtime extension that can be run without any problem, objects that will be used can be defined. Those objects definitions will be made by extending some BPMN objects, so the configuration in the plugin.xml will be accordingly.

The definition of Goal object is as follows:

```
<customTask
 description=
  "This task represents a Goal in the Goal Graph."
 featureContainer=
  "org.eclipse.bpmn2.modeler.runtime.example.
   SampleCustomTaskFeatureContainer"
 id="org.eclipse.bpmn2.modeler.runtime.example.goal"
 name="Goal"
 category="Goal Graph"
 icon="goal.jpg"
 runtimeId="org.eclipse.bpmn2.modeler.runtime.example"
   type="TextAnnotation">
 <property name="elementId" value=
   "org.eclipse.bpmn2.modeler.runtime.
        example.goal" type="EString"/>
 <property name="name" value="Goal" type="EString"/>
 <property name="text" value="Goal" type="EString"/>
 <property name="measurement" value="0" type="EInt"/>
</customTask>
```

With this definition, an object which is extended from TextAnnotation is defined. It has its own element id which is used in storing this object within .bpmn files. It is working under the runtime id defined above. It's name is Goal and it is located under Goal Graph section. It also have measurement feature, which is in type of integer. With this feature, a Goal object that is extended from TextAnnotation is capable of keeping measurement information.

After defining Goal object, it can be seen in Palette and can be added to the editor as a BPMN object. However, since we defined the measurement information as a property, one more definition should also be done. Since each BPMN object in Eclipse BPMN2 Modeler has its own proper section, a new property tab should also be defined for Goal

object. With this definition, property tab can be used with measurement information. The property tab definition for Goal object is as follows:

```
<propertyTab
 id="org.eclipse.bpmn2.modeler.runtime.example.goal.tab"
 replaceTab=
        "org.eclipse.bpmn2.modeler.textAnnotation.tab"
 class="org.eclipse.bpmn2.modeler.runtime.example.
   SampleGoalPropertySection"
 type="org.eclipse.bpmn2.TextAnnotation"
 label="Goal Attributes">
</propertyTab>
```

Now, our tool consists the Goal objects with a properly constructed property tab for our needs.

After defining the Goal object, Edge object can be defined. The definition of the Edge in the plugin.xml is as follows:

```
<customTask
 description="Associates Goals in Goal Graph."
 featureContainer="org.eclipse.bpmn2.modeler.
        runtime.example.
   SampleExtendedAssociationFeatureContainer"
 id="org.eclipse.bpmn2.modeler.runtime.example.assoc"
 name="Goal Edge"
 category="Goal Graph"
 icon="flow.png"
 runtimeId=
 "org.eclipse.bpmn2.modeler.runtime.example.assoc"
 type="Association">
 <property name="name" value="Edge" type="EString"/>
 <property name="contWeight" value="1" type="EDouble"/>
```

```
 <property name="elementId" value=
 "org.eclipse.bpmn2.modeler.runtime.example.assoc"
        type="EString"/>
</customTask>
```

With this definition, an object that is extended from Association is defined. Edge object also has its own element id which is used in storing this object in .bpmn files. Edge object is also working under the runtime id defined above. It's name is Goal Edge and it is also located in Goal Graph section. Also, Edge object have a contribution weight and that is in type of Double. With this feature, Edge object which is extended from Association can keep information of contribution weight.

The property tab arrangement which is done for Goal object should also be done for Edge object because of the same reasons. For the feature of contribution weight to be seen in property tab, a similar definition was made. Property tab definition for Edge object is as follows:

```
<propertyTab
 id="org.eclipse.bpmn2.modeler.runtime.
        example.association.tab"
 replaceTab="org.eclipse.bpmn2.modeler.assocation.tab"
 class="org.eclipse.bpmn2.modeler.runtime.example.
    SampleAssociationPropertySection"
 type="org.eclipse.bpmn2.Association"
 label="Edge">
</propertyTab>
```

Now, our tool can have Edge objects with a properly constructed property tab for our needs.

There is one last configuration for both Goal and Edge elements. For them to be seen in the editor properly at runtime, model enablement should also be defined.

```
<modelEnablement
 runtimeId="org.eclipse.bpmn2.modeler.runtime.example"
 type="Process"
 profile="Default Process"
 ref="org.eclipse.bpmn2.modeler.runtime.none:
        Process:Default Process">

 <enable object="TextAnnotation"/>
 <enable object="TextAnnotation" feature="name"/>
 <enable object="TextAnnotation" feature="measurement"/>

 <enable object="Association"/>
 <enable object="Association" feature="contWeight"/>

 <enable object="ManualTask" feature="timeToComplete"/>
</modelEnablement>
```

Now, both Goal and Edge objects are configured for our needs.

The methods and configurations of our BPMN extension was explained above. Now, the results of those steps will be shown in sequence.

Final version of the Goal objects shown in the extension is presented in Figure 4.1.

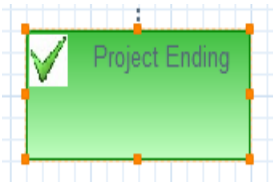

Figure 4.1: Graphical representation of Goal objects

Property tab of the Goal objects is presented in Figure 4.2.

Those were the graphical representations of Goal objects. As mentioned several times in this section, it is also important to be able to store the Goal objects within the .bpmn files. As an output of our extension, Goal objects can be stored in .bpmn files in such

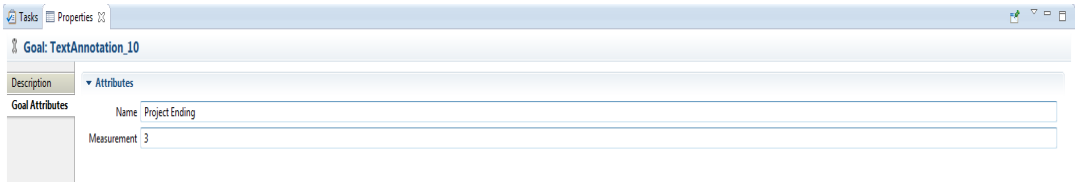

Figure 4.2: Property tab of Goal Objects

way:

```
<bpmn2:textAnnotation id="TextAnnotation_10"
 tns:elementId="org.eclipse.bpmn2.modeler.
        runtime.example.goal"
 tns:name="Project Ending" tns:measurement="3">
 <bpmn2:text>Project Ending</bpmn2:text>
</bpmn2:textAnnotation>
```

Final version of the Edge objects shown in the extension is presented in Figure 4.3.

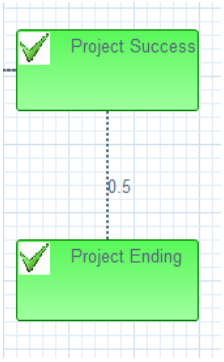

Figure 4.3: Graphical representation of Edge objects

Property tab of the Edge objects is presented in Figure 4.4.

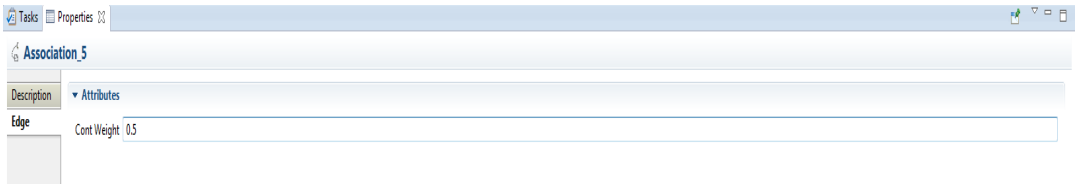

Figure 4.4: Property tab of Edge Objects

Contribution weights are also critical information about GoalDAG and they should also be stored within .bpmn files. As an output of our extension, Edges can be stored

in .bpmn files as follows:

```
<bpmn2:association id="Association_5"
 tns:elementId="org.eclipse.bpmn2.modeler.
        runtime.example.assoc"
 tns:name="0.5" tns:contWeight="0.5"
 sourceRef="TextAnnotation_10"
 targetRef="TextAnnotation_9"/>
```

4.2 Tetrad Input

As mentioned in the previous sections, main goal of this study is to develop a methodology for goal oriented business process improvement. In our methodology, goal oriented structure is represented by GoalDAG, which was described in chapter 3. As a first step of our methodology, a new tool has been developed for representing our goal hierarchy. With the use this new Eclipse BPMN2 Extension, it is possible to represent GoalDAG within BPMN files. It is also possible to make relations between process elements and Goals.

With the development of this first step, Goals and Edges can be represented in a BPMN editor and a GoalDAG can be constructed. However, this was not sufficient for goal oriented business process improvement. In order to be able to make improvements in organization's processes, some analysis should be done and results should be collected. If your aim is to make goal oriented process improvement, those analysis and result collection steps should be implemented for goals. As the second step of our study, our goal data has been prepared for strategic analysis. As the result of this anaylsis step, business processes can be improved.

Strategic analysis tools and methods are out of scope of this study. However, it is essential to say that, there are various number of such tools and methods with different usages. Tetrad was chosen for this study because it is the strongest one among the tools which can make analysis on Structued Equation Models. It also provides a user friendly interface. Technical details of Tetrad were described in chapter 2.

As mentioned in chapter 2, Tetrad aims to provide sophisticated statistical methods in a friendly interface to users without any programming knowledge. It provides analysis methods for Structured Equational Models and it also provides a graphical representation for models. Tetrad provides Structured Equational Model analysis in two different ways. In the first way, it allows to create data within the program and carry out analysis according to them. In the second way, it allows to load data into the program in a special XML format and makes analysis according to the provided XML files.

Since Tetrad accepts Structured Equational Model data in XML format, this feature of the program makes it possible to convert the Goals that are obtained with GoalDAG BPMN Extension. In this way, statistical analysis of the Goals can be done and business processes can be improved according to the results. As the second part of this study, a program has been developed to transform Goals that are obtained from Eclipse GoalDAG Extension to the XML format that can be accepted as input by Tetrad.

This part of the study has three main steps. First step of this part is to parse .bpmn files in order to extract information. There are two different alternatives to extract information from those .bpmn files that consists Goals. First way of this problem is to use standard XML parsers. In this way, too many string operations should be done to parse the .bpmn files. Also, Goals that are represented in .bpmn files have different attributes. Using standard XML parsers to extract those information can increase the complexity of the program and can cause time and memory problems for bigger files. In the other way, some methods that are provided by developers of Eclipse BPMN2 Modeler can be used. They have provided some basic methods to *read* the .bpmn files and extract information. When those methods are used, time, memory and efficiency problems can be eliminated. So, for this step, those methods are used. In this parsing operation, TextAnnotation elements representing Goals and Association elements representing Edges are identified. Thereafter, they are transformed into Java objects. In this way, a data structure that holds Goals and Edges is constructed.

As the second step, all the objects that are obtained at the first step is filtered. After this process, Goals and connections between those Goals are obtained. By means of

this process, Goals and Edges are converted to the Objects which will become suitable for Tetrad.

As the third step, XML files that are suitable for Tetrad is created. The data structures that are obtained from the results of the second step is used in this step. By this step, Goals and Edges are represented in XML files. The critical part of this step is that, XML format should be convenient for Tetrad notation. In this step, in order to maintain the concurrency with Tetrad, those objects are transformed into String objects. After they are formatted properly for Tetrad, XML files can be constructed.

After the steps that are defined above, Goals and Edges that are modeled by GoalDAG BPMN Extension became suitable for being analyzed within Tetrad. In XML files, Goals and Edges that are prepared for Tetrad are represented as follows:

```
<variable>Project_Ending</variable>
<edge value="0.5">Project_Ending
        --&gt; Project_Success</edge>
```

4.3 BPMN Extension To Combine Goal Hierarchy and Goal Question Metric

With the combination of the methodologies described previously in this chapter, it is possible to analyze processes. Goals can be related to processes and contribution of the intermediate goals to the main goals can be seen. However, this approach does not provide a full cycle of business process improvement since it does not support any iteration on process analysis and improvement steps. In this section, features of the BPMN extension to support Goal Question Metric approach will be described. These features will be able to support a cycle of business process improvement.

With the methodology described previously in this chapter, a Goal Graph can be given to Tetrad as input and a strategic analysis can be done. However, with this approach, results cannot be directly correlated with the processes. Tetrad makes it possible to take input of measurements of nodes as well as the Structured Equational Model [15] graph. It has the capabilities of constructing the SEM graph according to the analysis of the given measurements. In order for a full cycle of business process

improvement, those features of Tetrad are chosen. The reasons of this choice will be explained below.

Tetrad can be used to analyze the GoalDAG of a given process. With this analysis, business process improvements can be done for the processes. If a previously designed GoalDAG is used for the analysis step, it will not be possible to see the direct effects of process elements to the main goals. However, if measurements of the processes are used in order to construct a GoalDAG, it will also be possible to see the effects of the process elements to the Goals. Methodologies described in the previous sections help analyzing processes. In order to be able to perform a continuous cycle of business process improvement, process elements and Goals are correlated and the effects of the measurements to the GoalDAG will be displayed directly.

In order to make analysis within Tetrad, measurements of the nodes should be given. In this design, each node in SEM graph represents Goals in GoalDAG. However, those Goals does not represent process elements. In order to link the process elements with Goals, the GoalDAG implementation which was described in the first section of this chapter was used. In this way, each Goal will also become related to an element within the process. Since all process elements are steps that are done by organizations, they can be measured.

4.3.1 Requirements of the Extension To Combine Goal Hierarchy and Goal Question Metric

Goal Question Metric [34] is a common standard in the area of software measurement. That is the reason of the choice of GQM in this study. Since process elements should be measured to analyze and improve the processes, GQM is chosen as the measurement standard. With this solution, each process element can be represented as a measurable unit. Also, with the use of the GoalDAG extension, each measurable process element can be linked to Goals.

With the implementation described in section 4.1 it possible to represent Goals within the same editor with the processes. It is also possible to link the process elements with the Goals. In order to make a full cycle improvement that is described in this sec-

tion, a module that will be able to take measurement information of process elements should be implemented. Also, this module will be compatible with Goal Question Metric. In other words, measurements that will be entered will be compatible with Goal Question Metric.

Since the key point is to combine Goal Question Metric approach with BPMN processes, a new methodology should be developed. As described in chapter 3 Goal Question Metric is an approach which has three levels. First level is the Goal, second level is the Question and the third level is the Measurement. Since our aim is to provide a measurement standard for business process elements, those levels should be followed for them. The steps that are followed to construct the new methodology are as follows:

- Since we want to measure the elements, our first level will be process elements. Goals in the Goal Question Metric approach will be represented by business process elements.
- In the second level, appropriate questions should be asked for goals. Our aim for the second level will be providing a structure that will allow us to combine goals and questions.
- In the third level, data should be collected. Those data will be related to the question that is asked in the second level. Our aim for the third level will be providing a structure that will allow us to combine goals, questions and metrics.

4.3.2 Implementation of the Extension To Combine Goal Hierarchy and Goal Question Metric

In the previous subsection, requirements of the Eclipse Extension to combine Goal Hierarchy and Goal Question Metric are listed. Those requirements brings us to a solution of extending the editor. Since we want to correlate those three levels in our BPMN extension, they should be represented in a single unit. Since goals will be represented by process elements, the key point is to relate questions and measurements to the process elements. Since we want them to be in a single BPMN element, question

and measurement parts should be new features added to the objects. Technical details of this new feature are explained below.

Since the main goal is to be able to add question and measurement information to the process elements, defining new elements for BPMN is unnecessary. The approach that is used in representing GoalDAG can also be followed for this feature. Since any process element that is related to a Goal can be measured with GQM approach, it is logical to make a general extension which can be used for all elements. The key point of this extension is that, question and measurment information should be added to the elements. Also, it should also be provided that, processes that are sketched with this extension should be generic. In other words, those processes should be compatible with other BPMN tools and engines. This means, .bpmn files that are produced by this extension should not include element types that are not in BPMN Metamodel. Those restrictions bring us to a solution which is similar to the one that is described in section 4.1. Also, in this solution that is used to add this new feature, extending the BPMN Metamodel is unnecessary. It also makes produced .bpmn files incompatible with other tools. Adding new elements that will keep questions and measurement information in the metamodel is also unnecessary. That solution will cause .bpmn files include elements that are not compatible with other BPMN tools. The best approach for this problem is, extending some existing BPMN elements by adding them some features. Those element extensions will provide us to keep measurement information of the elements.

Since the choice is to extend BPMN elements, configurations that is related to element extension should be done. Classes that are used to extend existing elements should also be implemented. Since a successfully running BPMN extension was implemented in section 4.1 , it became easier to add the new features. In order to add new features to the existing BPMN extension, it is sufficient to make definitions of elements that will be extended. Since the aim is to extend the main process elements, Task element should be used. Details of this extension is as follows.

```
<customTask
 description="This task represents a
        Measurable Task for GoalDAG."
```

```
 featureContainer=
 "org.eclipse.bpmn2.modeler.runtime.example.
        SampleCustomTaskFeatureContainer"
 id="org.eclipse.bpmn2.modeler.runtime.example.task"
 name="Task"
 runtimeId="org.eclipse.bpmn2.modeler.runtime.example"
 type="Task">
 <property name="elementId" value=
        "org.eclipse.bpmn2.modeler.runtime.example.goal"
        type="EString"/>
 <property name="question" value="0" type="EString"/>
 <property name="measurement" value="0" type="EInt"/>
</customTask>
```

With this definition, an object which is extended from Task is defined. It has its own element id which is used in storing this object within .bpmn files. It is working under the runtime id defined in section 4.1. It has a question part which is in type of String. It also has a measurement feature, which is in type of integer. With this feature, a Task object that is extended from Task is capable of keeping measurement information of asked questions for specified goals.

After defining Task object, it can be seen in Palette and can be added to the editor as a BPMN object. However, since we defined the question and measurement parts as a property, one more definition should also be done. Since each BPMN object in Eclipse BPMN2 Modeler has its own proper section, a new property tab should also be defined for Task object. With this definition, property tab can be used with question and measurement informations. Property tab definition for a Task object which is compatible with GQM is as follows:

```
<propertyTab
 id="org.eclipse.bpmn2.modeler.runtime.example.task.tab"
 replaceTab="org.eclipse.bpmn2.modeler.task.tab"
 class="org.eclipse.bpmn2.modeler.runtime.example.
```

```
        SampleTaskPropertySection"
 type="org.eclipse.bpmn2.Task"
 label="Task Attributes">
</propertyTab>
```

Final version of a Task object is presented in Figure 4.5.

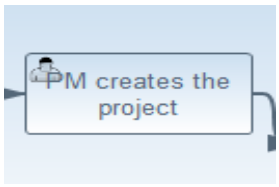

Figure 4.5: Graphical Representation of Task Oject

Final version of the Property Tab of a Task object is presented in Figure 4.6.

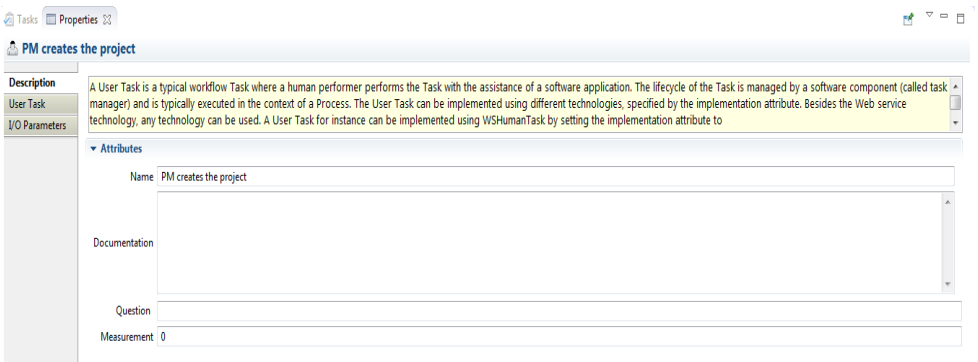

Figure 4.6: Property Tab of Task Object

As an output of the measurement module of our extension, Task objects that are used in the Goal Question Metric approach can be stored in .bpmn files in the following way:

```
<bpmn2:task id="Task_18"
 tns:elementId=
  "org.eclipse.bpmn2.modeler.runtime.example.task"
 tns:name="Project Ending"
  tns:question="Did the project end successfully?"
        tns:measurement="4">
```

```
 <bpmn2:text>Project Ending</bpmn2:text>
</bpmn2:task>
```

The key point in this module is that, Task object that is extended is not a new element. It only adds a new information to the existing Task object. This new information is represented under the Property Tap of the Task object. This makes it different from the Goal object that is defined in section 4.1 since the Goal object is a new element that is extended from TextAnnotation element. Because of this difference, Task objects are still located under their original folder in the Palette.

4.4 BPMN Extension To Show Analysis Results

In the previous section, the measurement module of Eclipse BPMN2 Extension is introduced. This module is used to support Goal Question Metric approach in Eclipse BPMN Editor. When this approach is supported, business process improvements can be done according to the measurements. In this section, modules of the extension that help showing analysis resultss will be explained. The main idea of this module is to show the contribution weights identified by Tetrad on Goal Hierarchy which is sketched in Eclipse extension.

4.4.1 Requirements of the BPMN Extension To Show Analysis Results

In order to show the resulting Goal Hierarchy in the editor, new features should be added to the extension. The key point in these features is that, measurements that are collected within the editor are analyzed by Tetrad and results are collected. The editor should be capable of showing the results of Tetrad so that a full cycle of business process improvement can be observed with the use of the editor.

4.4.2 Design of the BPMN Extension To Show Analysis Results

In the previous subsection, requirements of the Eclipse extension are listed. In order to satisfy those requirements, a new features should be added to the Eclipse BPMN

Extension. Those new features will make the editor capable of giving the collected measurements to Tetrad for analysis and showing the analysis results at the runtime. In other words, analysis results of the collected data will be shown as a Goal Graph.

Those requirements lead us to a solution consists of two main stages. In the first stage, measurement information that is compatible with Goal Question Metric approach is collected and prepared for Tetrad analysis. In the second stage, analysis results obtained from Tetrad is shown in Eclipse BPMN2 Extension.

4.4.2.1 Implementation of Collecting Measurements

In order to prepare appropriate data for Tetrad to make analysis on our business processes, measurement information should be given to Tetrad in correct formats. As mentioned in section 4.3, measurement information can be loaded to the system and it is compatible with Goal Question Metric approach. This information can be entered to the editor with the use of interfaces that are described in section 4.3. In order collect all those information together and combine them in a single data format, new features are added to the extension.

Since our aim in this step is to collect the data entered to the editor, a structure should be developed that will be running under the extension. That structure should be capable of reaching process elements that keeps measurement information as well as GoalDAG Hierarchy. Since each Goal object in the GoalDAG Hierarchy is related to a process element, it will be sufficient to reach to the process element or Goal object. Since Goal objects are modified versions of TextAnnotation objects, it will be easier to reach them to collect information.

In order to obtain measurement information, a new Constraint is defined. Constraints are parts of Eclipse Extensions with the use of validation. In this case, a new constraint is defined to validate TextAnnotation objects since they are representing Goal objects. The definition of the constraint is as follows:

```
<constraint
  lang="Java"
```

```
  class="org.eclipse.bpmn2.modeler.runtime.
         example.validation.TextAnnotationConstraint"
  severity="ERROR"
  mode="Batch"
  name="\%constraint.name.1"
  id="org.eclipse.bpmn2.modeler.runtime.
        example.validation.TextAnnotation"
  statusCode="1">
 <description>
 \%constraint.description.1
 </description>
 <message>
  \%constraint.message.1
  </message>
  <target class="TextAnnotation"/>
</constraint>
```

In this definition, a new class *TextAnnotationConstraint* is implemented. This definition allows us to validate each TextAnnotation object within the process. In that way, the constraint module of the program is able to reach every TextAnnotation object iteratively. Since process elements that are related to Goals are connected to Goal objects and Goal objects are extended from TextAnnotation objects, in this way, it is possible to reach every measurement entered to the editor. Since Eclipse Constraint mechanism allows user to validate each object iteratively, all measurements information can be gathered and collected together after iterations.

After all iterations, all measurements are collected in a single data structure. In this step, this data structure is a List since it is an appropriate Java object for a collection of data. Since the aim is to be able to analyze the measurements, Tetrad functions for SEM analysis can be used in this step. The data structure that is generated from the measurements are given to Tetrad function for SEM analysis.

4.4.2.2 Implementation of Showing Analysis Results

After the first step of this section that is descibed above, measurement data that is entered to the editor can be analyed by Tetrad. In order to complete the cycle of improvement, the results of the analysis should be shown to the user. Tetrad constructs a SEM graph that consists of Nodes and Edges. Nodes correspond to the Goals in our GoalDAG hierarchy and Edges correspond to the Edges in GoalDAG Hierarchy. The edges and contribution weights of the resulting Goal Graph will be identified by Tetrad. Since edges are Association objects that have contribution weight information, it is possible to set contribution weights of each object. In order to show all contribution weights a new feature is defined. This new feature is also defined as a constraint. This constraint will validate the process according to the results that are collected from Tetrad and set contribution weight information of edges. The definition of the constraint is as follows:

```
<extension
point="org.eclipse.emf.validation.constraintProviders">
<category
 name="\%category.name.20"
id="org.eclipse.bpmn2.modeler.
       runtime.example.validationCategory"/>
<constraintProvider cache="true">
<package namespaceUri=
"http://www.omg.org/spec/BPMN/20100524/MODEL-XMI"/>
<constraints categories=
"org.eclipse.bpmn2.modeler.runtime.
       example.validationCategory">

<constraint
lang="Java"
class=
"org.eclipse.bpmn2.modeler.runtime.example.validation.
ProcessConstraint"
```

```
 severity="ERROR"
 mode="Batch"
 name="\%constraint.name"
 id="org.eclipse.bpmn2.modeler.runtime
        .example.validation.Process"
 statusCode="1">
 <description>
 \%constraint.description
 </description>
 <message>
   \%constraint.message
 </message>
 <target class="Process"/>
</constraint>

</constraints>
</constraintProvider>
</extension>
```

This definition provides a new constraint that will validate the process. A new class named *ProcessConstraint* is implemented. Contribution weights are setted to the edges within that class.

Also, for the two new constraints that are expressed above to work properly, constraint binding definitions should be adjusted. Binding definitions are as follows:

```
<extension
 point=
 "org.eclipse.emf.validation.constraintBindings">
 <clientContext
 default="false"
 id="org.eclipse.bpmn2.modeler.
        runtime.example.validationContext">
```

```
 <enablement>
 <test
 property=
 "org.eclipse.bpmn2.modeler.property.targetRuntimeId"
 value=
 "org.eclipse.bpmn2.modeler.runtime.example"/>
 </enablement>
 </clientContext>
 <binding
 context=
  "org.eclipse.bpmn2.modeler.runtime.
    example.validationContext"
 category=
  "org.eclipse.bpmn2.modeler.runtime.
    example.validationCategory">
 </binding>
</extension>
```

This binding definition binds the constraint with the defined runtime and enables it to work properly.

4.5 Final version of the BPMN Extension To Represent Goal Hierarchy

Earlier in this chapter, requirements of a new tool was described deeply. The solution steps to cover those requirements was explained. Technical details of those solutions was also expressed. A final version of the extension was built after those incremental steps. The final version of the whole extension is presented in Figure4.7:

In the final version of the editor, there is an N-N relationship between tasks and Goals. In other words, more than one task can affect a goal. If a goal is affected by two or more tasks, the contribution of tasks to the Goal is represented by weights of the associations. Also, a single task can affect more than one goal with same or different

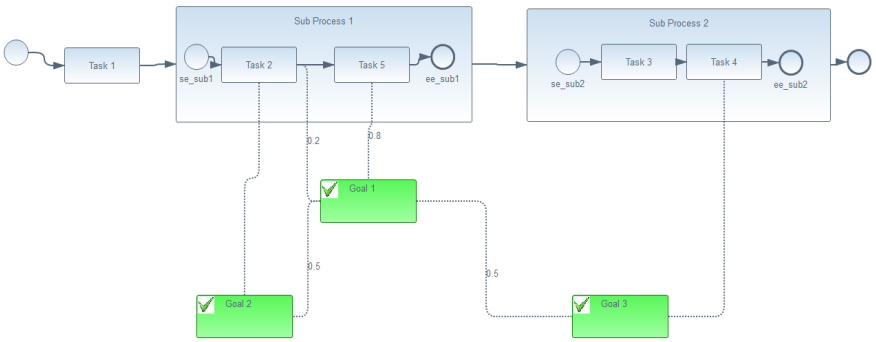

Figure 4.7: BPMN Extension for GoalDAG

weights.

In this example, a sample process was composed. Three goals are defined with names *Goal 1* , *Goal 2* and *Goal 3*. Contributions of Goal 2 and Goal 3 to Goal 1 is 0.5. Goal 3 is related to Task 4, Goal 1 is related to Task 2 and Task 5 and Goal 2 is also related to Task 2. Goals are also combined together to construct a goal hierarchy with the use of edges. In the Palette, a new section named Goal Graph is added. In this section Goal and Edge objects are located. Those objects can be added to the editor with basic drag and drop capabilities of the Palette. All those features make this new tool a fulfilling extension for our needs.

CHAPTER 5

PROJECT MANAGEMENT CASE STUDY

In order to show the benefits of the tool described in chapter 4, a case study is undertaken. As mentioned, the main goal of this work is to provide tool support for a methodology for business process improvement. It describes a concept called GoalDAG, and aims to make business process improvements according to GoalDAG.

In order to achieve the goal of the study, a new BPMN extension that will represent GoalDAG was developed. Second, data transformation from BPMN processes that are constructed with the extension to Tetrad is provided. Those methodologies made it possible to analyze the business processes using the provided GoalDAG.

Although performing an analysis became possible, those methodologies developed have not been enough to make a full cycle business process improvement. To fulfill those needs, Goal Question Metric approach is combined with Goal Hierarchy and implementations have been done to support their relations. With this approach, it has been possible to add measurement information to the processes. With these measurements, a new analysis can be done by Tetrad and the results can be shown in the GoalDAG. Since the new version of is able to show both the measurements and generated GoalDAG, it is possible to make improvements.

5.1 Choosing Processes For Business Process Improvement

In order to demonstrate the steps described above, project management processes are used. For this work, project management processes of a software development work

are considered. However, with the methodologies described in this thesis, any process related to any area can be targeted with same techniques.

Project management processes are usually critical for organizations. They show the steps of how projects are executed in the company. The inputs of the process usually come from different branches of the organization. In other words, activities of different branches are usually small parts of the project management process. However, there is only one output of the process. A newly developed software is the product of all these steps and teams. In order to make the project management process more efficient, business process improvement can be applied.

The project management process used in this case study is gathered from project management documents of a major financial institution. At the beginning, there was not a single process, so project managers were not able to see the entire process. The effects of the steps to the project goals were also not clear. As the very first step, project management documents are combined and a single complex project management process is formed. It consists of four main subprocesses which are Starting The Project, Planning The Project, Project Execution and Control, and Ending the Project.

Each subprocesses have their own start and end events. They also contain different kinds of *Tasks*. The final version that is sketched in Eclipse BPMN editor is presented in Figures A.1, A.2, A.3, A.4.

5.2 Identifying Goals and Analyzing the Project Management Process

In order to analyze the project management process, its Goals should be identified. After the definition and GoalDAG construction, they should be given to Tetrad as input. Here, since the process is project management and the main idea is to successfully create a product, some standards should be used. For the definiton of the Goals PMBOK [48] is used. In this book, critical points of project management processes are highlighted. Those points are followed to constuct a GoalDAG. Goals that are taken from PMBOK are as follows:

- Project Success

- Project Ending
- Starting The Project
- Change Management Quality
- Status Check Performance
- Planning the Project
- Feasibility Report Quality
- Project Plan Quality
- Resource Access
- Negotiating Schedule
- Obtain Infrastructure Resources
- Obtain Human Resources
- Project Support
- Effort Distribution
- Taking Care of Individual Characteristics
- Equitable distribution of effort
- Basis of effort distribution
- Granularity of Effort Distribution
- Cost Performance
- Time Performance
- Quality Performance

The final version of the GoalDAG is constructed with the use of BPMN extension that is described in section 4.1. The GoalDAG which is constructed according to the Goals that are taken from PMBOK is presented in Figure 5.1.

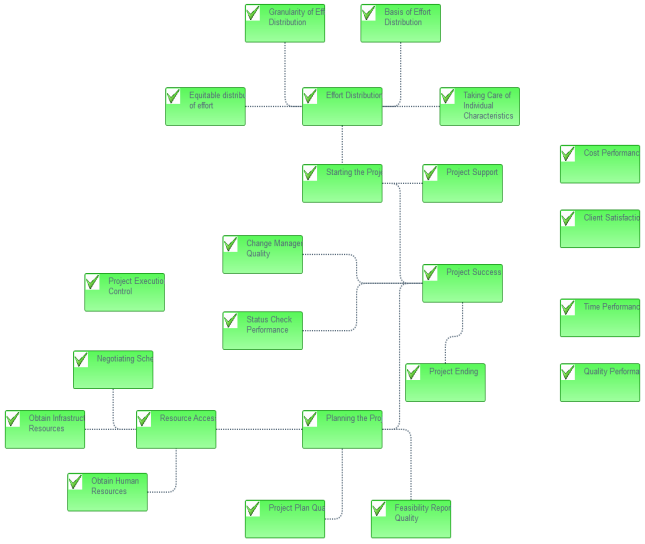

Figure 5.1: GoalDAG graph of project management process

In this version of the GoalDAG, the Goals are taken from PMBOK, as mentioned. Since the GoalDAG also consists of Edges, they should also be identified. Although the PMBOK does not directly result in hierarchy of Goals, it provides information about how Goals affect each other. Studying this information, Edges are identified.

After GoalDAG is constructed by BPMN extension, the next step is to prepare input for Tetrad. With the approach that is described in section 4.2 GoalDAG is converted to a Structural Equation Model Graph. This data type can be loaded to Tetrad with appropriate XML files. SEM graph which is loaded to Tetrad after these steps is presented in Figure 5.2.

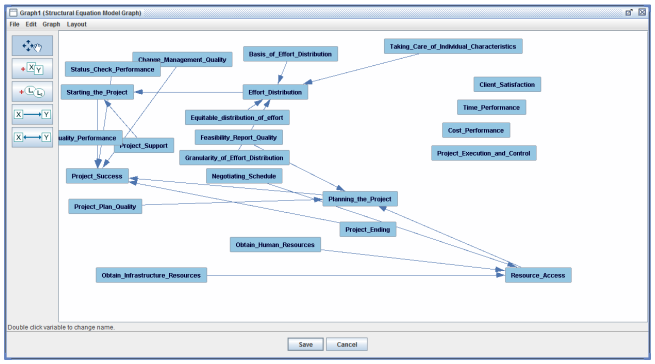

Figure 5.2: The version of GoalDAG after Tetrad transformation

When a SEM graph is loaded to Tetrad, it is possible to perform an analysis of the graph. Tetrad uses a combination of some statistical algorithms in the background and returns the user a final analysis result. The result given by Tetrad that contains contribution weights is presented in Figure 5.3.

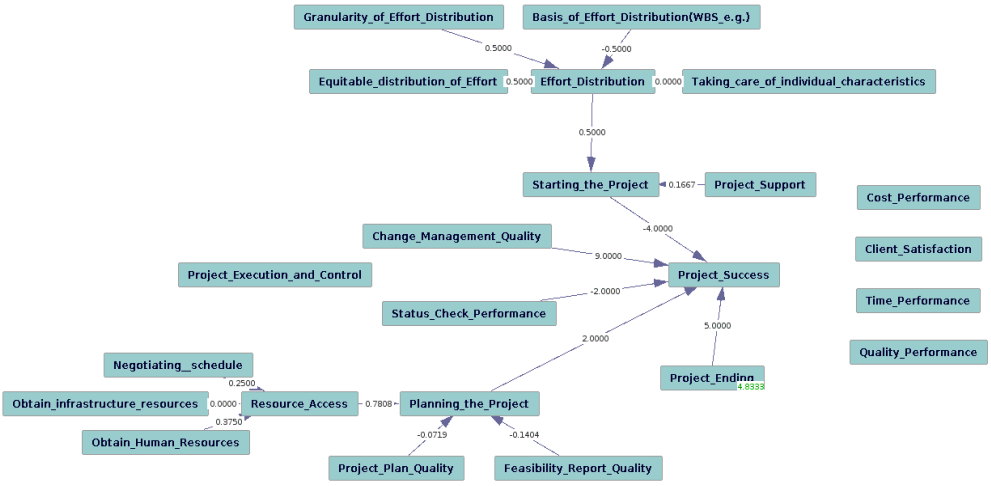

Figure 5.3: Analysis results given by Tetrad with contribution weights.

In this stage, Tetrad takes a graph without contribution weights as input. As a result, it produces an output which is a SEM graph containing contribution weights. Since the input graph does not contain any numerical information, Tetrad identifies the weights according to the number of incoming and outgoing edges of nodes. With the resulting graph, a process can be analyzed with a GoalDAG which contains contribution weights. This gives information about the effects of Goals to each other and to the main Goal of the process. In this specific case, *Project Success* is the main Goal of the process and there are five different Goals that are contributing to the Project Success with different weights. By looking at the value of each contribution weight, a detailed analysis can be performed for the process.

5.3 Introducing GQM to Process Elements for Business Process Improvement

In this step of the case study, the features developed to support Goal Question Metric approach which have been discussed in section 4.3 are used. In the first step of the case study that is explained above, Goals were sketched by using Eclipse BPMN Extension and in the second step GoalDAG was transformed into Structural Equation Model for strategic analysis. These steps have provided sufficient process analysis, however, the business process improvement was not complete.

In order to complete a cycle of business process improvement, GQM approach was combined with BPMN. This section provides steps to show benefits of these features. Processes are sketched in Eclipse by using the latest version of the extension. In this way, questions can be correlated with the process elements and measurement information of the process elements can be collected. The collected information can be given to Tetrad and obtained contribution weights can be displayed in Eclipse BPMN extension. After these steps, a user is able to see the resulting GoalDAG of the measurements obtained by process elements. These steps can be applied in an iterative way so that the changes in the results can be observed. This method can provide an improvement structure to the user. The user is able to observe how measurements can change the overall GoalDAG of the process.

As an example, three different steps with two iterations are applied to show how a user can observe business process improvements. In the first iteration, the measurement information collected from the process management data is entered to the system to see the resulting GoalDAG. As a result, a GoalDAG is displayed, showing the contribution weights of the Goals. This demonstrates how Goals have affected the main Goal. In the second iteration, some measurements are changed and a new version of GoalDAG is obtained with different contribution weights. This change in the contribution weights show the user how the changes in measurements affect the Goals of the overall system. In this way, a user will be able to see the possible improvements about the process.

In this example, Goals that are taken from PMBOK are used. Since the question and metric information is related with the process elements, the Goals are linked to

process elements. Since some of them are not directly related to any process element of project management process, they are eliminated. The mapping between Goals and process elements are presented in Table 5.1:

Table 5.1: Goals and related process elements

Goals and Related Tasks	
Goal	Related Task
Quality Performance	Quality Assurance Procedures
Client Satisfaction	PM organizes the assessment meeting
Time Performance	Entering the work hours
Status Check Performance	Taking the status report
Effort distribution	PM enters the effort distribution to the project
Starting the Project	Starting the project
Project Support	Project supporter approval
Cost Performance	PM creates the foundation for the project
Change Management Quality	PM organizes the change management plan
Project Ending	Ending the Project
Resource Access	PM asks for resources for the core team
Obtain Human Resources	Resource request and resource assignment
Planning the Project	Planning the Project
Project Plan Quality	Project Plan Execution
Feasibility Report Quality	PM prepares the feasibility report

According to the goals that are listed above, three steps with different questions and metrics are applied. The first step consists of specific questions and measurements for the first project. The second step consists of different questions and measurements for the second project. The last step consists of more general questions. At this last step, measurements of both third and fourth projects are used. After applying these three steps with multiple iterations, a detailed discussion about process improvement is provided.

5.3.1 Analysis and Improvement According to the First Project Data

As mentioned above, for the case study, measurement information from four different project data is collected to be analyzed. At the first step, only the first one of the four project data is usedfor analysis and improvement. For this step, after questions are identified and asked for related goals, metrics are entered. The metrics entered in this step are presented in Table B.1. Questions asked for this step are as follows:

- In order to collect a metric for *Quality Assurance Procedures*, the following question is asked: *How many procedures were applied for quality assurance?*
- In order to collect a metric for *PM organizes the assessment meeting*, the following question is asked: *How many assessment meeting were organized?*
- In order to collect a metric for *Entering the work hours*, the following question is asked: *How many hours were entered for the proect?*
- In order to collect a metric for *Taking the status report*, the following question is asked: *How many status reports were taken during the entire project?*
- In order to collect a metric for *PM enters the effort distribution to the project*, the following question is asked: *Was the effort distribution equitable? Rate between 0-5*
- In order to collect a metric for *Starting the project*, the following question is asked: *How many steps were applied when starting the project?*
- In order to collect a metric for *Project supporter approval*, the following question is asked: *How many different items were approved by the project supporter?*
- In order to collect a metric for *PM creates the foundation for the project*, the following question is asked: *Was the foundation enough for the project? Rate between 0-5*
- In order to collect a metric for *PM organizes the change management plan*, the following question is asked: *How many steps were included in the change management plan?*

- In order to collect a metric for *Ending the Project*, the following question is asked: *How many steps were applied when ending the project?*
- In order to collect a metric for *PM asks for resources for the core team*, the following question is asked: *How many different units are asked for resources?*
- In order to collect a metric for *Resource request and resource assignment*, the following question is asked: *What was the number of resources assigned for the core team?*
- In order to collect a metric for *Planning the Project*, the following question is asked: *How many people are included in the project plannig steps?*
- In order to collect a metric for *Project Plan Execution*, the following question is asked: *How many steps are included in the project plan execution phase?*
- In order to collect a metric for *PM prepares the feasibility report*, the following question is asked: *How many different criteria are used in order to prepare the feasibilty report?*

At this step of the case study, questions asked are mostly related to the quantitative values of process elements. They are usually asking the number of steps or documents. This gives information about the levels of detail of process elements. According to the metrics of those questions, a GoalDAG is obtained. This GoalDAG is presented in Figure 5.4.

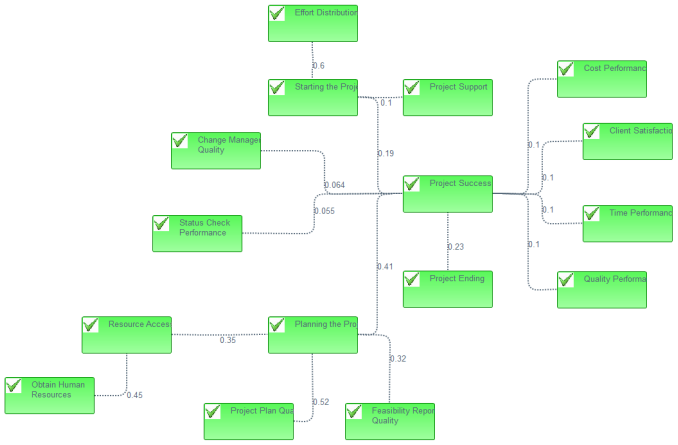

Figure 5.4: Resulting GoalDAG of the first iteration of the first step.

Once the GoalDAG is obtained, the values of the metrics are increased in order to see an improvement. The increased metrics entered in this step are also presented in Table B.1. Since the metrics are mostly representing the levels of detail, this improvement gives an idea about how process can be effected when process elements are more detailed. An improved version of GoalDAG is obtained which is presented in Figure 5.5.

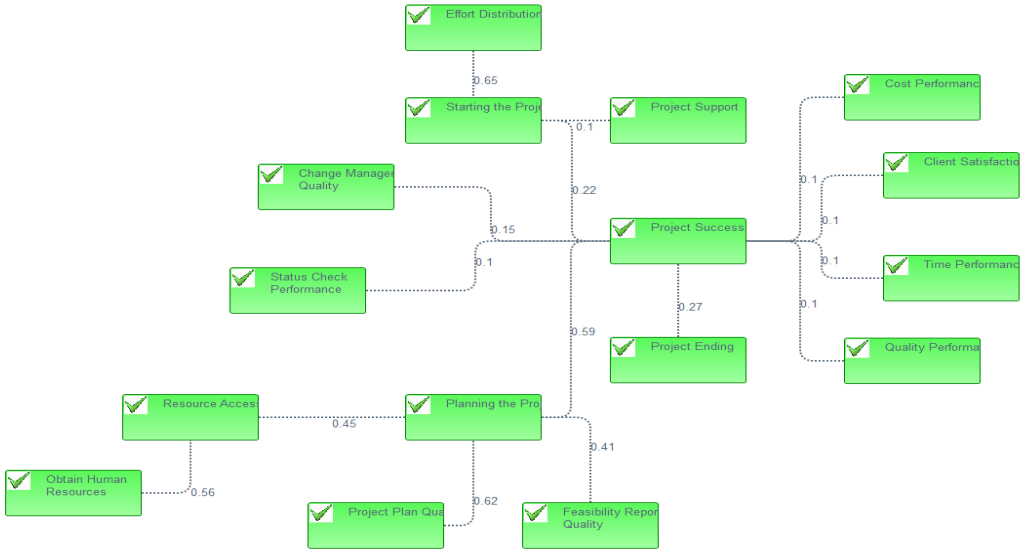

Figure 5.5: Resulting GoalDAG of the second iteration of the first step.

Contribution weights of these two iterations that are obtained in this scenario are also presented in Table B.2.

5.3.2 Analysis and Improvement According to the Second Project Data

As mentioned above, for the case study, measurement information from four different project data is collected to be analyzed and at the second step, only the second one of the four project data is used for analysis. At this step, different questions are identified and asked for related goals. The metrics entered in this step are presented in Table C.1 Questions asked for this step are as follows:

- In order to collect a metric for *Quality Assurance Procedures*, the following question is asked: *How many days did it take to identify the quality assurance*

procedures?

- In order to collect a metric for *PM organizes the assessment meeting*, the following question is asked: *What was the average duration of assessment meetings?*
- In order to collect a metric for *Entering the work hours*, the following question is asked: *How many days did it take to plan and enter the work hours?*
- In order to collect a metric for *Taking the status report*, the following question is asked: *What was the average duration of preparing a single status report?*
- In order to collect a metric for *PM enters the effort distribution to the project*, the following question is asked: *How many days did it take for the PM to calculate and enter the effort distribution for the project?*
- In order to collect a metric for *Starting the project*, the following question is asked: *How many days did it take to complete the starting project steps?*
- In order to collect a metric for *Project supporter approval*, the following question is asked: *How many days did it take for the project supporter to approve the project?*
- In order to collect a metric for *PM creates the foundation for the project*, the following question is asked: *How many days did it take to create and prepare the foundation?*
- In order to collect a metric for *PM organizes the change management plan*, the following question is asked: *How many days did it take for the PM to organize the change management plan?*
- In order to collect a metric for *Ending the Project*, the following question is asked: *How many days did it take to apply the project ending phases?*
- In order to collect a metric for *PM asks for resources for the core team*, the following question is asked: *How many days did it take for the PM to calculate and ask for resources ?*
- In order to collect a metric for *Resource request and resource assignment*, the following question is asked: *How many days did it take to assign the resources?*

- In order to collect a metric for *Planning the Project*, the following question is asked: *How many days did it take for the planning team to plan the project phases?*
- In order to collect a metric for *Project Plan Execution*, the following question is asked: *How many days did it take for the planning team to inspect the execution of the project phases?*
- In order to collect a metric for *PM prepares the feasibility report*, the following question is asked: *How many days did it take to prepare the feasibilty report?*

At the current step of the case study, questions asked are mostly related to the duration values of process elements. According to the metrics of those questions, a GoalDAG is obtained. This GoalDAG is presented in Figure 5.6.

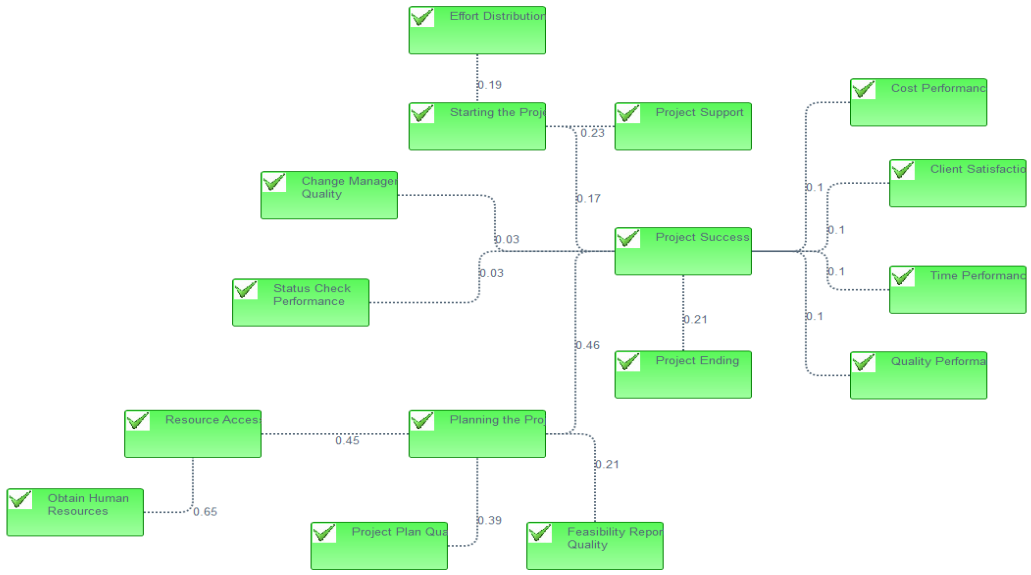

Figure 5.6: Resulting GoalDAG of the first iteration of the second step.

Once the GoalDAG is obtained, the values of the metrics are increased in order to see an improvement. The increased metrics entered in this step are also presented in Table C.1. Since the metrics are mostly representing the duration of process elements, this improvement will give an idea about how process can be affected when completing process elements takes more time. An improved version of GoalDAG is obtained which is presented in Figure 5.7.

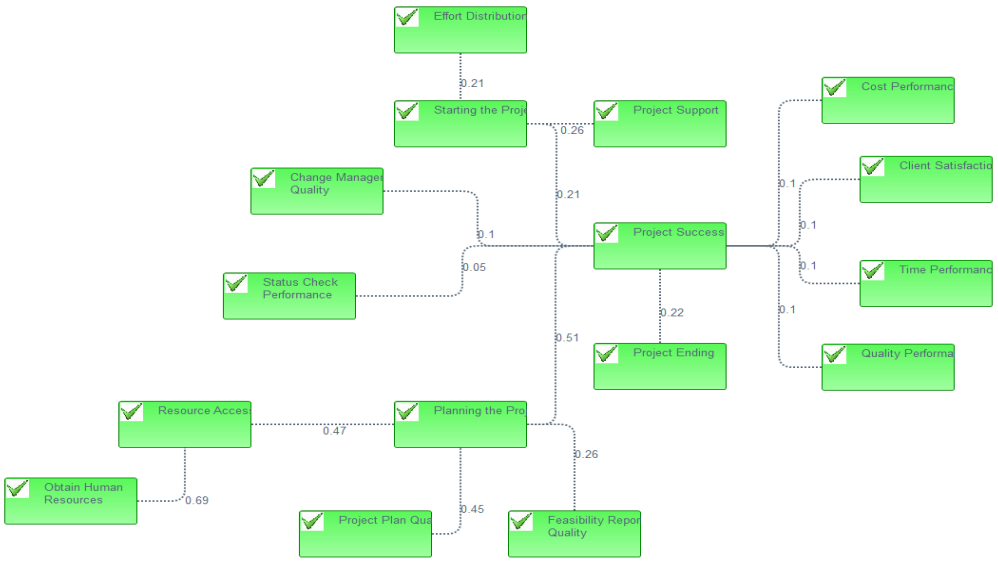

Figure 5.7: Resulting GoalDAG of the second iteration of the second step.

Contribution weights of these two iterations that are obtained in this scenario are also presented in Table C.2.

5.3.3 Analysis and Improvement According to the Third and Fourth Project Data

For the third step of the case study, the third and the fourth project data is used. The metrics entered in this step are presented in Table D.1. All the questions asked in this step are ranged between 0-10 and they are as follows:

- In order to collect a metric for *Quality Assurance Procedures*, the following question is asked: *Rate the quality of quality assurance procedures?*
- In order to collect a metric for *PM organizes the assessment meeting*, the following question is asked: *Rate the efficiency of assessment meetings?*
- In order to collect a metric for *Entering the work hours*, the following question is asked: *Rate the quality of the steps which are followed when entering the work hours?*

- In order to collect a metric for *Taking the status report*, the following question is asked: *Rate the quality of status reports?*
- In order to collect a metric for *PM enters the effort distribution to the project*, the following question is asked: *Rate the efficiency of effort distribution?*
- In order to collect a metric for *Starting the project*, the following question is asked: *Rate the quality of the steps which are taken when starting the project?*
- In order to collect a metric for *Project supporter approval*, the following question is asked: *Rate the approvement performance of project supporter?*
- In order to collect a metric for *PM creates the foundation for the project*, the following question is asked: *Rate the quality of the foundation?*
- In order to collect a metric for *PM organizes the change management plan*, the following question is asked: *Rate the quality of change management plan?*
- In order to collect a metric for *Ending the Project*, the following question is asked: *Rate the quality of the steps which are followed when ending the project?*
- In order to collect a metric for *PM asks for resources for the core team*, the following question is asked: *Rate the quality of asking resources?*
- In order to collect a metric for *Resource request and resource assignment*, the following question is asked: *Rate the efficiency of resource assignment?*
- In order to collect a metric for *Planning the Project*, the following question is asked: *Rate the quality of the steps which are followed during project planning?*
- In order to collect a metric for *Project Plan Execution*, the following question is asked: *Rate the efficiency of project plan execution?*
- In order to collect a metric for *PM prepares the feasibility report*, the following question is asked: *Rate the quality of the feasibility report?*

Questions asked in this step are mostly related to the quality or the efficiency of process elements. According to the metrics of those questions, a GoalDAG is obtained. This GoalDAG is presented in Figure 5.8.

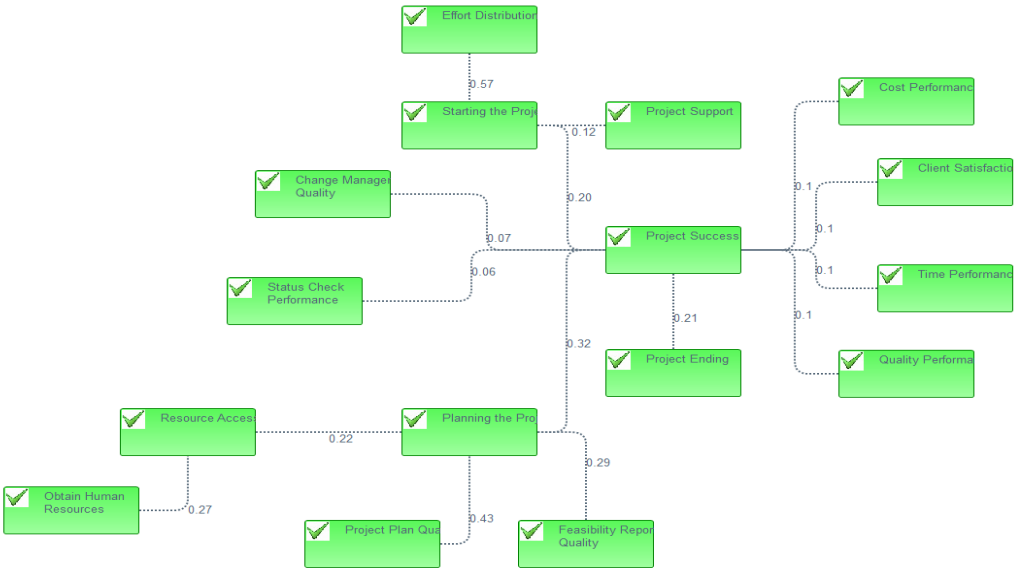

Figure 5.8: Resulting GoalDAG of the first iteration of the third step.

Once the GoalDAG is obtained, the values of the metrics are increased in order to see an improvement. The increased metrics entered in this step are also presented in Table D.1. Since the metrics are mostly representing the quality or efficiency of process elements, this improvement will give an idea about how process can be effected when qualities are increased. An improved version of GoalDAG is obtained which is presented in Figure 5.9.

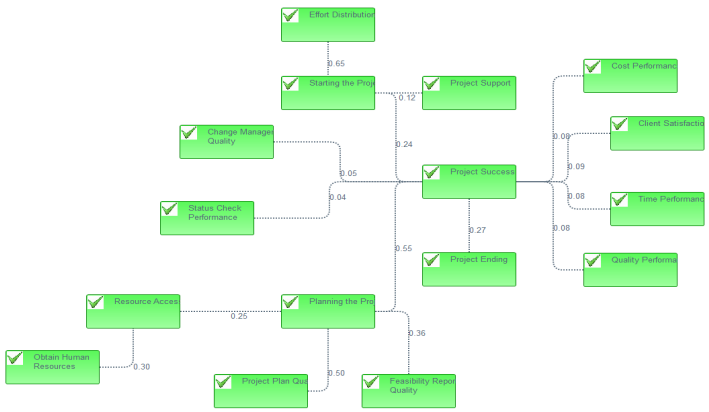

Figure 5.9: Resulting GoalDAG of the second iteration of the second step.

Contribution weights of these two iterations that are obtained in this scenario are also presented in Table D.2.

5.3.4 Discussion of Three Scenarios and Possible Improvements

In order to see the possible improvements about Project Management Process, a case study is undertaken which consists of three major steps. In this subsection, a discussion on these steps is provided.

At the first step, questions related to the quantitative values of process elements are asked and a first version of GoalDAG is obtained. At the second iteration of the first step, measurements are given higher values to see an improved version of GoalDAG. Increasing the measurements in this steps means having more detailed process elements. For example, when the number of phases followed during project execution and control is increased from 10 to 12, this leads to more detailed process elements. Also, increasing the values of metrics such as the number of documents, meetings or scenarios leads to increase in the level of detail of related process elements.

Comparison of the resulting GoalDAG of two iterations gives us some idea about how those changes affect the process performance. Initially, it can be observed that almost all the contribution weights have increased. This means, when the level of detail in the process increases, the effects of the elements to the project success increases as well. Additionally, it can be observed that the increase in the contributions of project planning phases is more dramatical than the remaining ones. This demonstrates that, increasing the steps of project planning phases have more impact on the project than the other phases such as starting or ending. Users can compare the effects of different process elements to the main goal and choose to increase some or all of them depending on their resources and needs.

At the second scenario, questions related to durations of process elements are asked and a first version of GoalDAG is obtained. Measurements values are then increased to see an improved version of GoalDAG. In this context, increasing a value means spending more time on the corresponding process element.

Comparison of the two iterations gives us some idea about the possible changes in the process performance. As more time is spent on the tasks, almost every contribution weight increases. This means, increasing the duration of the tasks can lead to a process improvement. Also it can be observed that, contribution of sub-goals to

the main goal have not escalated dramatically. It can be concluded that time values do not have substantially large effects on process improvement. In addition, it can be observed that only the contribution of *Change Management Quality* has increased dramatically. This shows to the user that spending more time on this task is critical for process improvement.

At the last scenario, questions related to quality and efficiency are asked and a first version of GoalDAG is obtained. Similarly with previous cases, measurement values are increased to see an improved version of GoalDAG.

Comparison of the two iterations gives us an idea about the effects of quality and efficiency on process improvement. It can be observed that, increasing the qualities have increased the contributions of *Project Start, Project Ending* and *Project Planning* on the main goal. It should also be noted that, contributions of small goals such as *Change Management Quality, Status Check Performance, Cost Performance, Quality Performance* and *Client Satisfaction* have decreased although their qualities were also increased. Evidently, the quality of such simple goals does not have considerable effects on the entire process. On the other hand, the quality of more complex tasks are critical for the process. Users can perform improvements by changing the qualities of some tasks depending on their needs.

Those three scenarios show us that, different types of metrics can affect the project success in different ways. For example, once the values of the metrics in the first scenario have increased, contributon of the Project Planning goals increased substantially. However, such substantial changes were not observed in the second scenario. It can also be observed that increasing the quality of some goals leads to a decrease in their contribution weights in some cases. Different scenarios with different types of metrics can also be applied and results can be displayed depending on the users needs.

CHAPTER 6

CONCLUSION AND FUTURE WORK

This thesis provides tool support for approaches and methodologies in the field of business process improvement. Business process management is a hot topic and business process improvement is a popular branch of it with growing interest.

There are various types of approaches in the field of business process improvement and the notable works are mentioned in this thesis. This work, however, supports a methodology for goal oriented business process improvement.

For goal oriented business process improvement, a previously defined goal hierarchy, named GoalDAG, is introduced. This goal hierarchy is planned to be correlated with business processes. For a generic usage, BPMN is chosen for business process notation. In order to correlate the goal hierarchy with BPMN processes, a new Eclipse extension was developed.

A transformation technique has been developed for strategic analysis. Tetrad was chosen as an analysis tool and, a transformation is implemented for the goal hierarchy that is sketched with the Eclipse extension.

In order to be able to perform a business process improvement, Goal Question Metric approach is followed. The extension has been able to support GQM approach in order to correlate measures with process elements.

There are many advantages of the approaches that are described in this thesis. First of all, a tool is developed that helps to combine the goal concept with business processes which makes it useful in the field of business process improvement. Second, a

strategic analysis methodology is presented for process analysis, which is also a new approach in that field. Finally, concepts of Goal Question Metrics and BPMN are also correlated within the extension and this combination is used for business process improvement. Combining these two concepts for the aim of business process improvement is what makes this thesis unique in the literature.

This thesis also provides a detailed case study which is undertaken with project management process. First, goals are identified and process is analyzed according to the GoalDAG. At the second part, data from four different projects are used and three different scenarios are applied with different questions and metrics. Process is analyzed according to three different perspectives and effects of these perspectives on process improvement are discussed.

A potential drawback of the work presented in this thesis is related with Goal Question Metric approach. In the supported Goal Question Metric method, the metrics are entered to the system with normalized values because the system combines those values and gives them to Tetrad at the background for analysis. As a future direction, this feature can be improved and the extension can support different ranges for each metric. However, this improvement is out of scope for this thesis since it includes complex statistical methods.

REFERENCES

[1] H. Nash, "Harvey Nash 2013 CIO Survey," Tech. Rep., 2013, [last accessed on 14-Aug-2014]. [Online]. Available: http://www.harveynash.com/group/mediacentre/2013CIO_survey.pdf

[2] The Economist, "The Strategic CIO," Tech. Rep., 2013, [last accessed on 14-Aug-2014].

[3] CIO, "2013 State of the CIO," Tech. Rep. January, 2013, [last accessed on 14-Aug-2014]. [Online]. Available: http://www.cio.com/documents/pdfs/2013StateoftheCIOExecSummary.pdf

[4] S. White, "Introduction to BPMN," Tech. Rep., 2004. [Online]. Available: http://www.bpmn.org/Documents/Introduction%20to%20BPMN.pdf

[5] OMG, *Business Process Model and Notation (BPMN), Version 2.0*, Object Management Group Std., Rev. 2.0, January 2011. [Online]. Available: http://www.omg.org/spec/BPMN/2.0

[6] M. Chinosi and A. Trombetta, "BPMN: An introduction to the standard." *Computer Standards & Interfaces*, vol. 34, no. 1, pp. 124–134, 2012. [Online]. Available: http://dblp.uni-trier.de/db/journals/csi/csi34.html#ChinosiT12

[7] S. White, "Business Process Modeling Notation (BPMN) Version 1.0," BPMI.org, Tech. Rep., 2004. [Online]. Available: http://www.bpmn.org/Documents/BPMN%20V1-0%20May%203%202004.pdf

[8] (2013) BPMN2 Modeller. Eclipse. [last accessed on 14-Aug-2014]. [Online]. Available: http://eclipse.org/bpmn2-modeler/

[9] OMG. (2013) BPMN 2.0 Metamodel. [last accessed on 14-Aug-2014]. [Online]. Available: http://www.omg.org/bpmn/Documents/BPMN_2-0_RFP_07-06-05.pdf

[10] "Eclipse BPMN2 Modeler User Guide," Tech. Rep., 2004.

[11] (2014) Eclipse Graphiti. Eclipse. [last accessed on 14-Aug-2014]. [Online]. Available: http://www.eclipse.org/graphiti/

[12] (2014) Eclipse Plugin Development Environment. Eclipse. [last accessed on 14-Aug-2014]. [Online]. Available: http://www.eclipse.org/pde/

[13] (2013) Eclipse Plug-in Development Environment Guide. Eclipse. [Online]. Available: http://help.eclipse.org/kepler/index.jsp?nav=%2F4

[14] (2013) The Tetrad Project. Carnagie Melon University. [last accessed on 14-Aug-2014]. [Online]. Available: http://www.phil.cmu.edu/projects/tetrad/

[15] J. Pearl, *Causality: Models, reasoning and inference*. Cambridge, UK: Cambridge University Press, 2000.

[16] R. K. Ko, S. S. Lee, and E. W. Lee, "Business Process Management (BPM) standards: A survey," *Business Process Management journal*, vol. 15, no. 5, 2009.

[17] M. H. Jansen-vullers and M. Netjes, "Business process simulation – a tool survey," in *In Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN*, 2006.

[18] B. Andersson, I. Bider, P. Johannesson, and E. Perjons, "Towards a formal definition of goal-oriented business process patterns." *Business Proc. Manag. Journal*, vol. 11, no. 6, pp. 650–662, 2005.

[19] K. Tumay, "Business process simulation," in *Proceedings of the 27th Conference on Winter Simulation*, ser. WSC '95. Washington, DC, USA: IEEE Computer Society, 1995, pp. 55–60.

[20] C. Güngör and H. Oğuztüzün, "GoalDAG - Archimate Integration." in *Proceedings of 20th International Conference on Information and Software Technologies (ICIST)*, 2014, pp. 194–210.

[21] R. W. Robinson, *Combinatorial Mathematics V*. Springer Berlin / Heidelberg, 1977, vol. 622, ch. Counting unlabeled acyclic digraphs, pp. 28–43.

[22] R. S. Kaplan and D. P. Norton, "The Balanced Scorecard: Measures that Drive Performance," *Harvard Business Review*, vol. 70, pp. 71–79, 1992.

[23] R. P. Kaplan and D. P. Norton, "Linking the balanced scorecard to strategy," *California Management Review*, vol. 39, pp. 53–79, 1996.

[24] S. J. Bleistein, K. Cox, and J. Verner, "Validating strategic alignment of organizational it requirements using goal modeling and problem diagrams," *Journal of Systems and Software*, vol. 79, no. 3, pp. 362 – 378, 2006.

[25] E. Yu, L. Liu, and Y. Li, "Modelling strategic actor relationships to support intellectual property management," in *Conceptual Modeling - ER 2001*, ser. Lecture Notes in Computer Science, H. S.Kunii, S. Jajodia, and A. Sølvberg, Eds. Springer Berlin / Heidelberg, 2001, vol. 2224, pp. 164–178, 10.1007/3-540-45581-7.

[26] J. Gordijn and J. Akkermans, "Value-based requirements engineering: exploring innovative e-commerce ideas," *Requirements Engineering*, vol. 8, pp. 114–134, 2003, 10.1007/s00766-003-0169-x.

[27] E. Santos, J. Castro, J. Sanchez, and O. Pastor, "A Goal Oriented Approach for Variability in BPMN." in *WER*, G. D. S. Hadad, O. Dieste, and J. P. Carvallo, Eds., 2010.

[28] B. Korherr and B. List, "Extending the epc and the bpmn with business process goals and performance measures." in *ICEIS (3)*, J. Cardoso, J. Cordeiro, and J. Filipe, Eds., 2007, pp. 287–294.

[29] B. Burmeister, M. Arnold, F. Copaciu, and G. Rimassa, "Bdi-agents for agile goal-oriented business processes." in *AAMAS (Industry Track)*, M. Berger, B. Burg, and S. Nishiyama, Eds. IFAAMAS, 2008, pp. 37–44.

[30] G. Koliadis and A. Ghose, "Relating business process models to goal-oriented requirements models in kaos." in *PKAW*, ser. Lecture Notes in Computer Science, A. G. Hoffmann, B. H. Kang, D. Richards, and S. Tsumoto, Eds., vol. 4303. Springer, 2006, pp. 25–39.

[31] K. Decreus and G. Poels, "A goal-oriented requirements engineering method for business processes." in *CAiSE Forum*, ser. Lecture Notes in Business Information Processing, P. Soffer and E. Proper, Eds., vol. 72. Springer, 2010, pp. 29–43.

[32] L. Braubach, A. Pokahr, K. Jander, W. Lamersdorf, and B. Burmeister, "Go4flex: Goal-oriented process modelling." in *IDC*, M. Essaaidi, M. Malgeri, and C. Badica, Eds., vol. 315, 2010, pp. 77–87.

[33] V. R. Basili, "Software modeling and measurement: the Goal/Question/Metric paradigm," Techreport UMIACS TR-92-96, University of Maryland at College Park, College Park, MD, USA, Tech. Rep., 1992.

[34] V. R. Basili, G. Caldiera, and D. H. Rombach, *The Goal Question Metric Approach*. John Wiley & Sons, 1994, vol. I.

[35] V. R. Basili, "Applying the Goal/Question/Metric Paradigm in the Experience Factory," Techreport UMIACS TR-92-96, University of Maryland at College Park, College Park, MD, USA, Tech. Rep., 1993.

[36] L. Lazic and N. Mastorakis, "Cost effective software test metrics," *W. Trans. on Comp.*, vol. 7, no. 6, pp. 599–619, Jun. 2008.

[37] P. Berander and P. Jönsson, "A goal question metric based approach for efficient measurement framework definition," in *Proceedings of the 2006 ACM/IEEE International Symposium on Empirical Software Engineering*, ser. ISESE '06. New York, NY, USA: ACM, 2006, pp. 316–325.

[38] L. Sánchez-González, F. García, F. Ruiz, and M. P. Velthuis, "Measurement in business processes: a systematic review." *Business Proc. Manag. Journal*, vol. 16, no. 1, pp. 114–134, 2010.

[39] R. E. Park, *Goal-driven software measurement: A guidebook (Handbook / Carnegie Mellon University. Software Engineering Institute)*. Carnegie Mellon University, Software Engineering Institute.

[40] S. M. Powell, S.G. and C. Trimble, "Measurement and control of business processes," *System Dynamic Review*, vol. 17, no. 1, pp. 63–91, 2001.

[41] J. Cardoso and W. Van Der Aalst, *Handbook of Research on Business Process Modeling*. Hershey, PA: Information Science Reference - Imprint of: IGI Publishing, 2009.

[42] J. Cardoso, "How to measure the control-flow complexity of web processes and workflows," in *Workflow Handbook 2005*, L. Fischer, Ed. Lighthouse Point, FL, USA: Future Strategies Inc., 2005, pp. 199–212.

[43] S. Cardoso, "Evaluating the process control-flow complexity measure," in *Proceedings of the IEEE International Conference on Web Services*, ser. ICWS '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 803–804.

[44] E. R. Aguilar, F. Ruiz, F. Garcia, and M. Piattini, "Applying software metrics to evaluate business process models." *CLEI Electron. J.*, vol. 9, no. 1, 2006.

[45] E. R. Aguilar, "An exploratory experiment to validate measures for business process models." in *RCIS*, C. Rolland, O. Pastor, and J.-L. Cavarero, Eds., 2007, pp. 271–280.

[46] E. R. Aguilar, F. Garcia, F. Ruiz, M. Piattini, C. A. Visaggio, and G. Canfora, "Evaluation of bpmn models quality - a family of experiments." in *ENASE*, C. Gonzalez-Perez and S. Jablonski, Eds. INSTICC Press, 2008, pp. 56–63.

[47] E. R. Aguilar, J. Cardoso, F. Garcia, F. Ruiz, and M. Piattini, "Analysis and validation of control-flow complexity measures with bpmn process models." in *BMMDS/EMMSAD*, ser. Lecture Notes in Business Information Processing, T. A. Halpin, J. Krogstie, S. Nurcan, E. Proper, R. Schmidt, P. Soffer, and R. Ukor, Eds., vol. 29. Springer, 2009, pp. 58–70.

[48] PMI, Ed., *A Guide to the Project Management Body of Knowledge (PMBOK Guide): An American National Standard ANSI/PMI 99-001-2008*, 4th ed. Newtown Square, PA: Project Management Institute, 2008.

APPENDIX A

PROJECT MANAGEMENT PROCESS

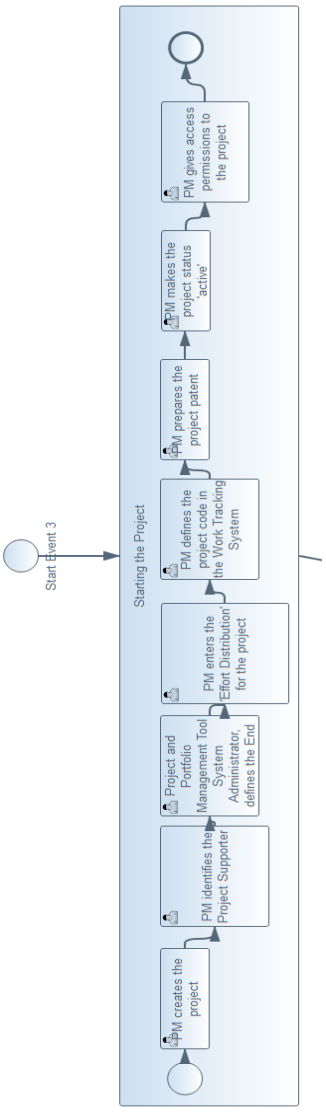

Figure A.1: Project Management Process - Part 1

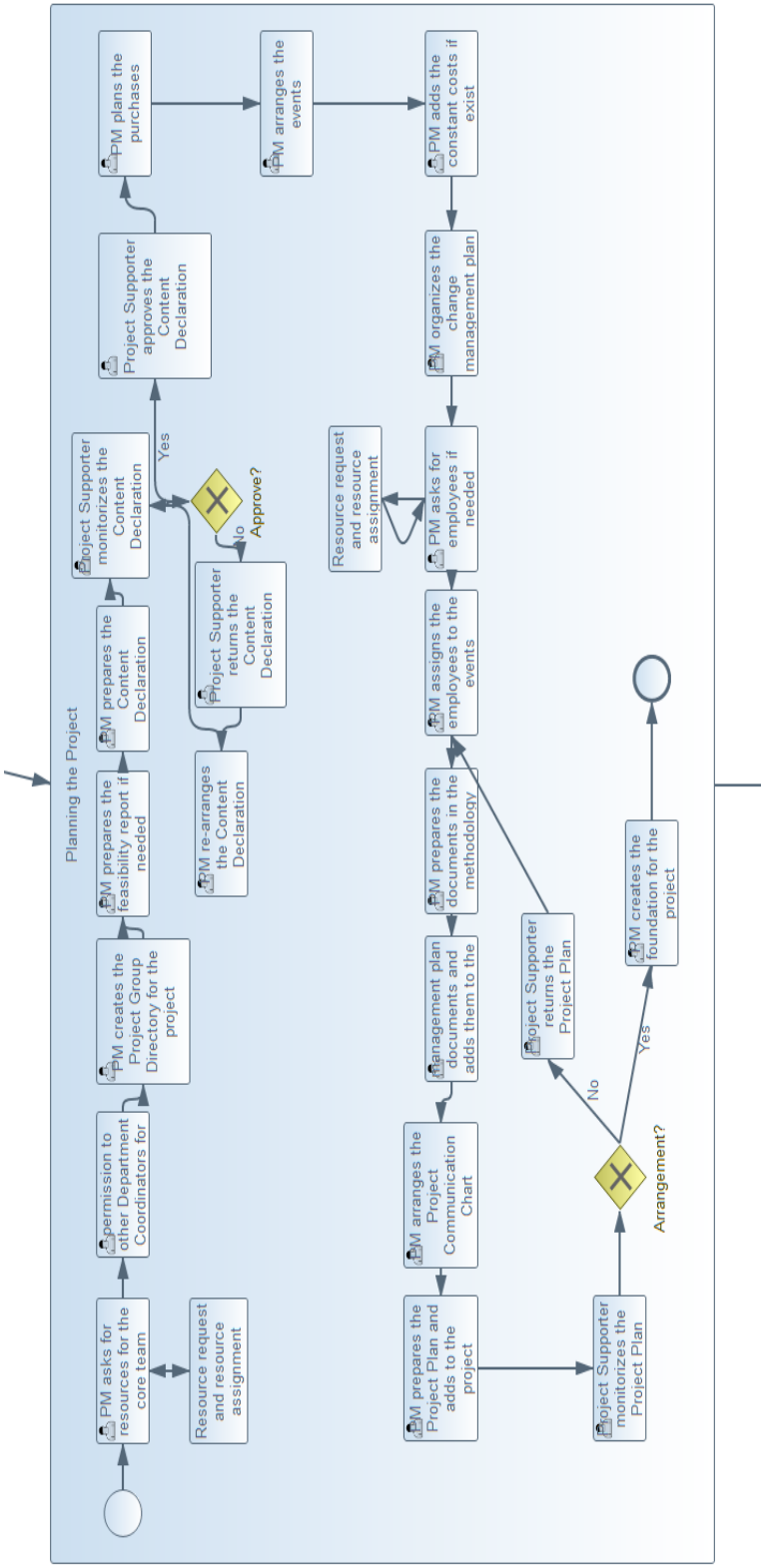

Figure A.2: Project Management Process - Part 2

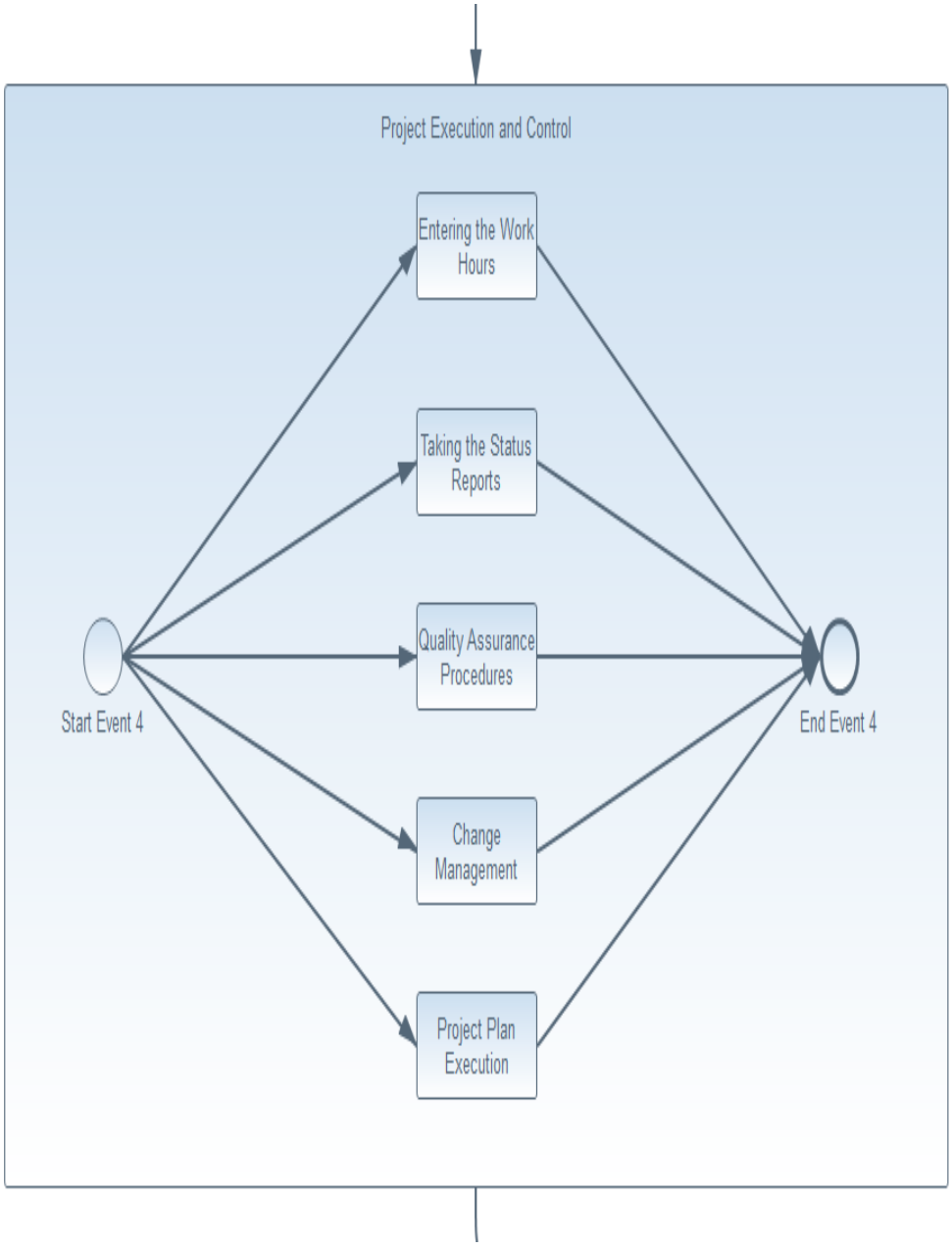

Figure A.3: Project Management Process - Part 3

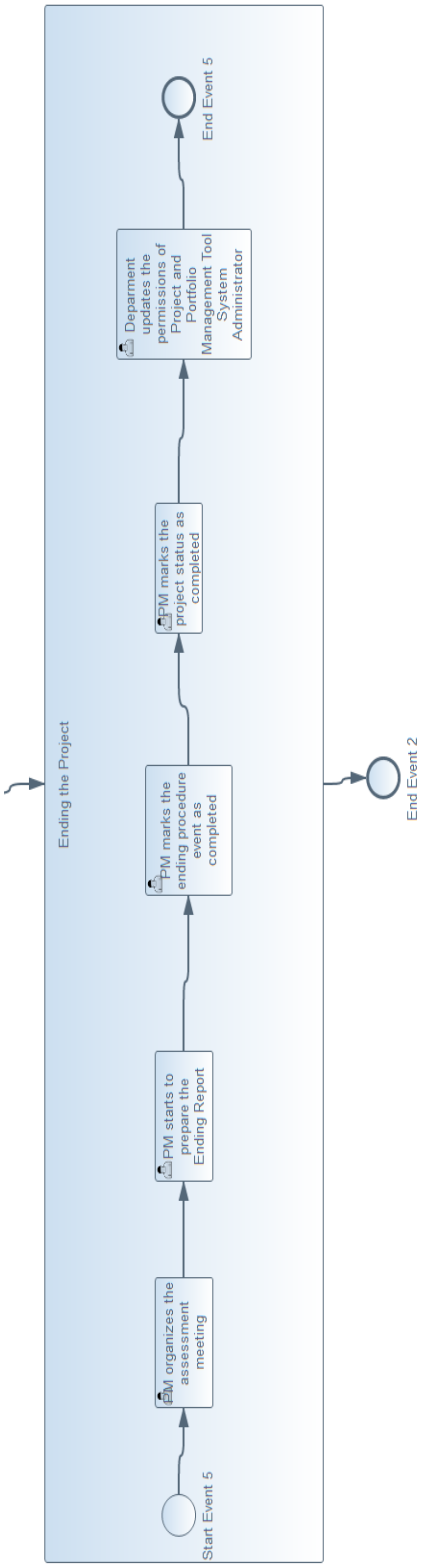

Figure A.4: Project Management Process - Part 4

APPENDIX B

LIST OF METRICS AND CONTRIBUTION WEIGHTS IN SCENARIO 1

Table B.1: Metrics that are used in the iterations in scenario 1

List of Metrics		
Task Name	Iteration 1	Iteration 2
Quality Assurance Procedures	6	7
PM organizes the assessment meeting	5	7
Entering the work hours	2400	2700
Taking the status report	14	16
PM enters the effort distribution to the project	3	4
Starting the project	22	24
Project supporter approval	5	6
PM creates the foundation for the project	3	4
PM organizes the change management plan	13	15
Ending the Project	14	16
PM asks for resources for the core team	3	4
Resource request and resource assignment	30	32
Planning the Project	7	8
Project Plan Execution	40	42
PM prepares the feasibility report	18	20

Table B.2: Contribution weights of two iterations in scenario 1

Contribution Weights		
Goal Name	Iteration 1	Iteration 2
Quality Performance	0.1	0.1
Client Satisfaction	0.1	0.1
Time Performance	0.1	0.1
Status Check Performance	0.055	0.1
Effort distribution	0.6	0.65
Starting the Project	0.19	0.22
Project Support	0.1	0.1
Cost Performance	0.1	0.1
Change Management Quality	0.064	0.15
Project Ending	0.23	0.27
Resource Access	0.35	0.45
Obtain Human Resources	0.45	0.56
Planning the Project	0.41	0.59
Project Plan Quality	0.52	0.62
Feasibility Report Quality	0.32	0.41

APPENDIX C

LIST OF METRICS AND CONTRIBUTION WEIGHTS IN SCENARIO 2

Table C.1: Metrics that are used in the iterations in scenario 2

List of Metrics		
Task Name	Iteration 1	Iteration 2
Quality Assurance Procedures	2	3
PM organizes the assessment meeting	1	1.5
Entering the work hours	4	5
Taking the status report	16	20
PM enters the effort distribution to the project	8	9
Starting the project	25	29
Project supporter approval	2	2.5
PM creates the foundation for the project	6	8
PM organizes the change management plan	5	6
Ending the Project	17	19
PM asks for resources for the core team	1	1.5
Resource request and resource assignment	3	4
Planning the Project	13	15
Project Plan Execution	12	14
PM prepares the feasibility report	5	7

Table C.2: Contribution weights of two iterations in scenario 2

Contribution Weights		
Goal Name	Iteration 1	Iteration 2
Quality Performance	0.1	0.1
Client Satisfaction	0.1	0.1
Time Performance	0.1	0.1
Status Check Performance	0.03	0.05
Effort distributiont	0.19	0.21
Starting the Project	0.17	0.21
Project Support	0.23	0.26
Cost Performance	0.1	0.1
Change Management Quality	0.03	0.1
Project Ending	0.21	0.22
Resource Access	0.45	0.47
Obtain Human Resources	0.65	0.69
Planning the Project	0.46	0.51
Project Plan Quality	0.39	0.45
Feasibility Report Quality	0.21	0.26

APPENDIX D

LIST OF METRICS AND CONTRIBUTION WEIGHTS IN SCENARIO 3

Table D.1: Metrics that are used in the iterations in scenario 3

List of Metrics		
Task Name	Iteration 1	Iteration 2
Quality Assurance Procedures	3	5
PM organizes the assessment meeting	4	5
Entering the work hours	6	8
Taking the status report	4	6
PM enters the effort distribution to the project	7	8
Starting the project	7	8
Project supporter approval	5	6
PM creates the foundation for the project	3	4
PM organizes the change management plan	6	7
Ending the Project	5	6
PM asks for resources for the core team	3	5
Resource request and resource assignment	3	4
Planning the Project	8	9
Project Plan Execution	7	9
PM prepares the feasibility report	5	7

Table D.2: Contribution weights of two iterations in scenario 3

Contribution Weights		
Goal Name	Iteration 1	Iteration 2
Quality Performance	0.1	0.08
Client Satisfaction	0.1	0.09
Time Performance	0.1	0.08
Status Check Performance	0.06	0.04
Effort distributiont	0.57	0.65
Starting the Project	0.20	0.24
Project Support	0.12	0.12
Cost Performance	0.1	0.08
Change Management Quality	0.07	0.05
Project Ending	0.21	0.27
Resource Access	0.22	0.25
Obtain Human Resources	0.27	0.30
Planning the Project	0.32	0.55
Project Plan Quality	0.43	0.50
Feasibility Report Quality	0.29	0.36