A SIZE MEASUREMENT METHOD FOR ENTERPRISE APPLICATIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS OF
THE MIDDLE EAST TECHNICAL UNIVERSITY
BY

NESLİHAN KÜÇÜKATEŞ ÖMÜRAL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
THE DEPARTMENT OF INFORMATION SYSTEMS

AUGUST 2022

A SIZE MEASUREMENT METHOD FOR ENTERPRISE APPLICATIONS

Submitted by NESLİHAN KÜÇÜKATEŞ ÖMÜRAL in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in the Department of Information Systems, Middle East Technical University** by,

Prof. Dr. Deniz ZEYREK BOZŞAHİN
Dean, **Graduate School of Informatics** ____________________

Prof. Dr. Sevgi ÖZKAN YILDIRIM
Head of Department, **Information Systems** ____________________

Assoc. Prof. Dr. Altan KOÇYİĞİT
Supervisor, **Information Systems Dept., METU** ____________________

Prof. Dr. Onur DEMİRÖRS
Co-Supervisor, **Computer Engineering Dept., IzTech** ____________________

Examining Committee Members:

Prof. Dr. Ferda Nur ALPASLAN
Computer Engineering Dept., METU ____________________

Assoc. Prof. Dr. Altan KOÇYİĞİT
Information Systems Dept., METU ____________________

Assoc. Prof. Dr. Ayça KOLUKISA TARHAN
Computer Engineering Dept., Hacettepe University ____________________

Asst. Prof. Dr. Özden ÖZCAN TOP
Information Systems Dept., METU ____________________

Assoc. Prof. Dr. Özgür TANRIÖVER
Computer Engineering Dept., Ankara University ____________________

Date: *25.08.2022*

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 : Neslihan Küçükateş Ömüral

Signature : ____________________

ABSTRACT

A SIZE MEASUREMENT METHOD FOR ENTERPRISE APPLICATIONS

Küçükateş Ömüral, Neslihan
Ph.D, Department of Information Systems
Supervisor: Assoc. Prof. Dr. Altan Koçyiğit
Co-Supervisor: Prof. Dr. Onur Demirörs

August 2022, 113 pages

Enterprise Applications are complex, effort-intensive applications that include most of the business processes of the enterprises. They differ from traditional software applications by their high reuse rates. Effort predictions for these applications are usually performed in an ad-hoc fashion and result in large variations. Prediction models utilizing traditional software size measures do not produce accurate results either. In this thesis, we developed a size measurement method considering the unique characteristics of EA projects. This method measures changes where pre-built functionality does not satisfy customer requirements. With the claim that the number of data groups executed for a transaction should be reflected in size, we defined a novel size measurement unit, Data Transaction Point (DTP). We proposed measuring the size of an EA project in terms of DTP in three categories: unchanged, changed, and new. In order to understand the applicability of the method in real life, we evaluated the accuracy of effort estimates based on DTP in multiple cases. The size measurement method has been evaluated through different SAP implementation projects. To assess the effort estimation accuracy, MMRE, MdMRE, and PRED (30) metrics have been used for the projects. The results showed that the accuracy of effort estimates for the projects was in an acceptable range. This method can reduce project schedule and budget overruns with promising effort estimation results. Moreover, it also has the potential to be used in different situations like the evolution and maintenance of software systems where measuring changes has crucial importance.

Keywords: Enterprise Applications, Enterprise Resource Planning, Software Size Measurement, Reuse, Data Transaction Point

ÖZ

KURUMSAL UYGULAMALAR İÇİN BİR BÜYÜKLÜK ÖLÇÜM METODU

Küçükateş Ömüral, Neslihan
Doktora, Bilişim Sistemleri Bölümü
Tez Yöneticisi: Doç. Dr. Altan Koçyiğit
Eş Danışman: Prof. Dr. Onur Demirörs

Ağustos 2022, 113 sayfa

Kurumsal Uygulamalar, iş süreçlerinin çoğunu içeren, karmaşık ve yoğun efor gerektiren uygulamalardır. Bu tarz uygulamalar, sahip oldukları yüksek yeniden kullanım oranları ile geleneksel yazılımlardan farklılaşırlar. Bu uygulamalar için efor tahminleri sıklıkla sistematik olmayan metotlar ile gerçekleştirilir ve büyük farklılıklarla sonuçlanır. Geleneksel yazılım büyüklüğü ölçümlerini kullanan tahmin modelleri de bu tarz uygulamalar için doğru sonuçlar vermez. Bu tez kapsamında, Kurumsal Uygulama Projeleri'nin kendilerine özgü özelliklerini göz önünde bulundurarak bir yazılım büyüklük ölçüm metodu geliştirdik. Bu metot ile uygulamanın sahip olduğu işlevlerin müşteri gereksinimlerini karşılamadığı durumda oluşan değişiklikleri ölçüyoruz. Bir işlem içinde kullanılan veri gruplarının sayısının büyüklüğe yansıtılması gerektiğini düşünerek, yeni bir büyüklük ölçüm birimi olan Veri İşlem Noktası (DTP)'nı tanımladık. Bir Kurumsal Uygulama Projesi'nin büyüklüğünü DTP olarak değişen, değişmeyen ve yeni olmak üzere üç kategoride ölçmeyi önerdik. Metodun alanda uygulanabilirliğini incelemek için, DTP kullanarak yapılan efor tahminlerinin doğruluğunu değerlendirdiğimiz çoklu durum çalışmaları gerçekleştirdik. Büyüklük ölçüm metodunu farklı SAP projeleri üzerinde uyguladık. Efor tahminlerinin doğruluğunu değerlendirmek için MMRE, MdMRE ve PRED (30) metriklerini kullandık. Sonuçlar, projeler için efor tahmin doğruluğunun kabul edilebilir ve güvenilir aralıkta olduğunu gösterdi. Umut veren efor tahmin sonuçları dikkate alındığında, bu metot proje takvimini ve bütçe aşımlarını azaltmaya katkıda bulunabilir. Bu metot, değişiklikleri ölçmenin kritik olduğu yazılım güncelleme ve bakım gibi projeler için de kullanılabilme potansiyeline sahiptir.

Anahtar Sözcükler: Kurumsal Uygulamalar, Kurumsal Kaynak Planlaması, Yazılım Büyüklük Ölçümü, Yeniden Kullanım, Veri İşlem Noktası

to my lovely son, Kerem

ACKNOWLEDGMENTS

I would thank my supervisor Assoc. Prof. Dr. Altan Koçyiğit for his support and important ideas throughout my study.

I would like to express my gratitude to my co-supervisor Prof. Dr. Onur Demirörs, for encouraging me to conduct this thesis study, motivating me when I had challenges and never giving up on his support. I am grateful to him for his patience, enlightening ideas and guidance throughout my study.

Besides my supervisors, I would like to thank Prof. Dr. Ferda Nur Alpaslan and Assoc. Prof. Dr. Aysu Betin Can for their valuable advices during the thesis monitoring meetings.

My sincere thanks go to my father Veysel Küçükateş, my mother Ayfer Küçükateş, and my sisters Aslı and Betül for their love, encouragement and support throughout my educational pursuit.

I would also like to thank my colleague Tuna Hacaloğlu for her unconditional friendship. I cannot forget her support to motivate me in the hardest times.

I owe my most special thanks to my husband Murat Ömural. He was always there to encourage me in the most difficult times of this long process. Thank you for loving me, believing in me and always supporting me.

And my son Kerem, who completely changed my life; I couldn't have walked this long road without his love and energy. We are growing together. Thank you for being in my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ASAP	Accelerated SAP
BI	Business Intelligence
BPR	Business Process Repository
CFP	COSMIC Functional Point
CO	Controlling
CRM	Customer Relationship Management
CS	Continuous Functional Similarity Reflective Size
DS	Discrete Functional Similarity Reflective Size
DTP	Data Transaction Point
EA	Enterprise Applications
ERP	Enterprise Resource Planning
FI	Financial Accounting
FPA	Functional Point Analysis
FSM	Functional Size Measurement
FUR	Functional User Requirements
HCM	Human Capital Management
HRM	Human Resources Management
LOC	Lines of Code
LOOCV	Leave One Out Cross Validation
MdMRE	Median of the Magnitude of Relative Error
MM	Material Management
MMRE	Mean Magnitude of Relative Error
MRE	Mean Relative Error
MRP	Manufacturing Resource Planning
PM	Plant Maintenance
PO	Purchase Order
PP	Production Planning
PP-PI	Production Planning – Process Industries
PS	Project System
RDS	Rapid Deployment Solutions
RRS	Reuse Reflective Size
RICEFW	Reports, Interfaces, Conversions, Enhancements, Forms, Workflows
QM	Quality Management

SaaS	Software as a Service
SCM	Supply Chain Management
SD	Sales & Distribution
SLR	Systematic Literature Review
SRM	Supplier Relationship Management
TRM	Treasury and Risk Management
WM	Warehouse Management

CHAPTER 1

INTRODUCTION

“Simple can be harder than complex: You have to work hard to get your thinking clean to make it simple. But it's worth it in the end because once you get there, you can move mountains.”

Steve Jobs

In this thesis, existing size measurement methods applicable for EA (Enterprise Applications) projects are analyzed, and a novel size measurement method for EA projects is proposed and validated. The remainder of this chapter presents the context of size measurement for EA projects, the problem definition, the purpose of the study, the research strategy, and the organization of the thesis.

1.1. The Context

Enterprise Applications are one of the most complex and effort-intensive information system solutions adopted by various organizations. Enterprise Applications initially emerged as MRP (Manufacturing Resource Planning) Systems, then ERP (Enterprise Resource Planning) Systems to handle different types of transaction-based back-office operations [1][2]. Since then, they have evolved to handle front-office and inter-organizational operations such as SCM (Supply Chain Management) Systems and CRM (Customer Relationship Management) Systems [1][3][4]. “Enterprise Applications” are defined as “commercial software packages that enable the integration of transactions-oriented data and business processes throughout an organization (and perhaps eventually throughout the entire inter-organizational supply chain)” [4].

“Enterprise Software”, “Enterprise Software Applications”, and “Enterprise Systems” terms are widely used to refer to Enterprise Applications (EA). Throughout this thesis, we will use the term “Enterprise Applications”. We define EA projects as the implementation of Enterprise Applications in specific organizations.

Markus and Tanis [4] defined the following key characteristics for Enterprise Applications:

- *Packages:* Enterprise Applications are commercial software packages; purchased or leased from software vendors instead of developed from scratch.

- *Software Life Cycle:* Rather than designing new software to meet the organization's needs, the adopters of Enterprise Applications often try to adjust organization's business processes to the EA package.

- *Best Practices:* Enterprise Applications are designed to handle generic business processes. Organizations redesign their business processes to adopt the EA package's best practices. Business process reengineering is an essential part of EA projects.

- *Integration:* In many cases, EA adopters need to interface the EA package to organization's existing software.

- *Configuration:* During the EA project, the configuration is done by setting software parameters based on the business processes to be implemented. Configuring an EA package for an organization is significantly different from software programming.

- *Evolving*: Enterprise Applications are rapidly changing based on industry expectations.

Enterprise Applications are complex as they include all processes, interactions, and financial transactions to meet the needs of different organizations on the same platform. Daneva and Wieringa [6] listed the following main features that differentiate ERP projects from other types of software projects:

- They cover thousands of business activities.

- They require diverse configuration and modification activities to reflect business requirements.

- They do not necessarily have a master architecture to glue the parts together.

Enterprise Applications' most critical feature is high reuse rates, EA reuse most of the pre-built functionality to fulfill customer requirements. Frakes and Kang [7] define "software reuse" as "using of existing software or software knowledge to construct new software". Sommerville [8] explains reuse in ERP projects as encompassing overall business functionality, with business processes configured for a specific organization. EA projects are known as one of the most successful reuse implementations in Software Engineering, allowing adopters to configure and customize the system to handle variations in the business processes [9].

A commonly used analogy describes the reuse concept in Enterprise Applications. An EA is likened to an oversized suit. When a company decides to implement the EA, it has to tailor the suit to fit its own size. This tailoring is done by “customization”. Determining the suit’s selectable parts, such as buttons, can be considered as “configuration”. Without configuration and customization, that suit would not fit the company.

Configuration is basically setting necessary business-specific parameters in the system to execute the EA package. Configuration does not include any changes to source code. However, customization includes source code changes. Customization in EA is a general term used for modifications made to the software to handle the customer’s requirements that are not supported by standard features of the software [10] [11] [12].

The life cycle of EA projects are mostly similar to the waterfall method. SAP[1], as one of the biggest EA vendors, recommends a project implementation methodology called ASAP (Accelerated SAP) [13], as illustrated in Figure 1.

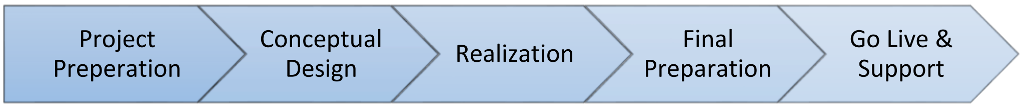


Figure 1. A Sample Project Lifecycle for EA Projects [13]

In the project preparation phase, the “Scope” document of the project is prepared, and in the conceptual design phase, the “Conceptual Design / Business Blueprint” document is prepared. While the modules and scenarios are listed in the "Scope" document, details about the business processes, including configuration and customization requirements, are defined in the “Business Blueprint” document. These documents are the primary source documents that could be used for the size measurement of the EA projects.

Size measurements and effort estimations are mostly performed in an ad-hoc fashion in EA projects, and as a result, they frequently suffer from time and budget overruns. Since the initial study of Stensrud and Myrtveit [14], which argued that traditional size measurement and effort estimation methods are not suitable for complex projects such as ERP projects, many research studies have been undertaken to solve the EA size measurement problem. In these studies, mostly function points based size measurement methods were used, and main source document for the size measurement was stated as “Business Blueprint” document [15].

Software size is a significant asset for effort estimation, project time scheduling, productivity measurement, risk management, quality management, and outsourcing management processes of the projects. Software functional size could also be used as

[1] https://www.sap.com/

a base unit for software acquisition, scope changes, and normalization of base project measures [16]. Several methods have been developed over the years for measuring software size. Functional size measurement methods such as COSMIC, IFPUG, and NESMA are widely used in software engineering [17]. These functional size measurement methods are also considered as potential size measurement methods for EA projects. Although many size measurement methods are tested for EA projects, none of the methods is widely accepted and validated as an appropriate method for EA size measurement.

1.2. The Problem

Enterprise Applications are one of the most complicated systems covering almost all processes of an enterprise. Due to this complexity and size, these kinds of projects frequently suffer from estimation variances [18]. Panorama Consulting Group analyzes the EA field globally and publishes yearly reports since 2008 [19]. A summarized view of last four yearly reports for items related to cost and duration is shown in Table 1.

Table 1. Year-Over-Year Comparison of EA Projects [19]

	2018	*2019*	*2020*	*2021*
Respondent Organizations	241	181	112	140
Organizations Faced Budget Overrun	45%	38%	60%	41%
Average Budget Overrun Rate	24%	66%	33%	*N/A*
Organizations Faced Schedule Overrun	58%	47%	46%	36%
Average Schedule Overrun Rate	11%	33%	20%	*N/A*
Average Effort Underestimate Rate	29%	15%	38%	35%

These analysis results show that organizations have been underestimating effort for years, and the underestimate rate is at least 15%. Since these projects are effort-intensive projects, underestimation of the effort reflects budget overruns. Besides budget overruns, organizations also face project schedule overruns. Both the project schedule and the project effort estimation are determined based on the size measurement data of the project. Thus, we can conclude that the size measurement of EA projects is not performed properly as evidenced by high rates of budget and schedule overruns.

The main factor that differentiates EA projects is the high reuse rates. When the size measurements of these projects are made without considering the reuse rates, it affects not only the effort estimation but also the stages such as budgeting and project planning. The amount of reuse should be clearly measured, and the size measurement should be made by taking this into account. Panaroma Consulting Group also analyzes customization levels of the EA projects, and publish results in their yearly reports [19]. Customization levels in the 2022 report are shown in Figure 2.

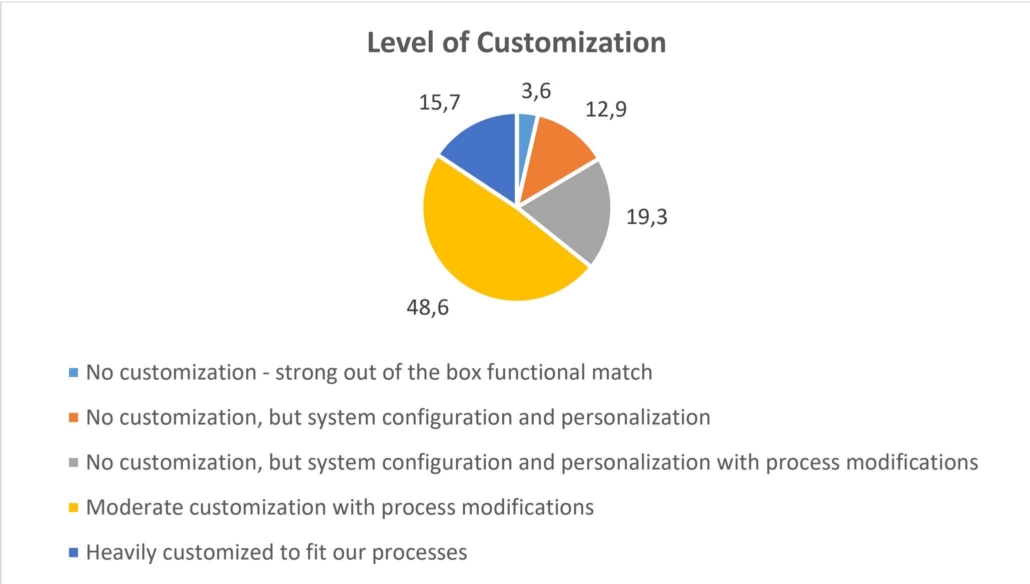


Figure 2. Level of Customization for EA Projects [19]

Customization is implemented by modifying the source code when the standard pre-built functionality is insufficient for the customer's business process requirements. Based on this definition, it is logical to see an inverse relationship between customization and reuse. As seen in Figure 2, only 3.6 % of organizations reuse all pre-built functionality without even performing configuration or personalization. 32.2% of organizations perform configuration and personalization, even process modification during EA implementation. 64.3% of organizations perform customization in EA projects. As the analysis shows, customization seems inevitable in most EA projects.

Daneva [20] performed an empirical study to analyze and measure reuse in ERP projects. In this study, she measured the levels of reuse in three SAP projects in the same business sector. Measurement results show that reuse was up to 80% possible at best and reuse levels varied based on modules implemented. Based on the results of this study, she argued that the reusability of the ERP projects should not be overvalued; companies adopting ERP should be ready to the minimum of 20% (in some cases presenting 40%) of pre-built functionality would not fit their needs and customization would be made.

Implications of the study [20] are also critical by showing that companies even in the same business sector face different reuse levels for the same EA packages. Although two similar companies implement the same EA packages, the size of the projects may differ depending on customizations or new reports / developments required. Despite this situation, parameters such as installed packages, number of users, number of employees, company size are frequently used in analogy-based estimation methods for EA projects.

With the development of cloud technology in recent years, EA vendors offer SaaS (Software as a Service) models for EA implementations. With this situation, smaller-scale organizations have also become able to implement EA. Currently, EA has a big share in the software industry; by offering cloud-based implementation models, it is expected to increase its market share. One of the technology research companies, namely Technavio [2], published a study related to expected growth in the Enterprise Application Software Market for 2022-2026 [21]. They stated in the report that with the spread of cloud-based deployment models, EA market share is expected to increase by $104 billion from 2021 to 2026. Another study published by The Business Research Company [22] shares similar insights. In this study, the total global business analytics & enterprise software market size is stated as $352.19 billion in 2021; the expectation for 2022 is stated as $406.68 billion and for 2026, it is stated as $694.07 billion.

The first study in which it was claimed that classical size measurement and effort estimation methods would not be suitable for comprehensive projects such as EA projects was the study of Stensrud [13]. In this study [14], some EA-specific metrics were defined to be used in effort estimation, such as modules, users, software interfaces, sites, business units, EDI interfaces, data conversions, custom-developed reports and modified screens. Later, these EA-specific metrics evolved to RICEFW (Reports, Interfaces, Conversions, Enhancements, Forms, Workflows) objects and were used as parameters for sizing EA projects in studies such as [23] [24].

Many studies exist in the literature in which present size measurement and effort estimation methods are adapted for EA projects. Daneva and Wieringa [6] evaluated existing methods applicability to the EA domain in their study. Daneva [25], then proposed to combine COCOMO II [26] , Monte Carlo Simulation [27] and Portfolio Management methods for EA projects. Erasmus and Daneva [28] explored the applicability of the Expert Judgment method, considering increasing uncertainties and customizations with new technologies. Erasmus and Daneva performed two more studies [29][30] investigating an integrated method with Functional Size Measurement and Expert Judgments. Function Point based size measurement methods were also examined for EA projects in studies such as [31] [32] [33].

Although EA projects are high-budget, long-term projects, the fact that a valid method for measuring the size of the projects has not been developed is one of the most significant shortcomings of this field. The lack of a valid size measurement method for these kinds of projects would result in more budget, and schedule problems as the EA market increases.

[2] https://www.technavio.com/

1.3. Purpose of the Study

The primary purpose of this thesis is to develop a size measurement method for EA projects. Such a size measurement method shall enable project managers to estimate effort, plan the project schedule, and monitor and control the project [16]. We aim for this method to be suitable for the characteristics of EA projects that distinguish them from other software projects.

In order to achieve this purpose, we defined the following research questions to be answered through this thesis:

RQ1. Which size measurement methods have been developed or proposed for EA projects?

RQ1.1. What are the pros and cons of these methods?

RQ1.2. How are these size measurement methods validated, and what is the accuracy?

RQ2. What are the significant characteristics of EA projects that affect size?

RQ3. How can significant characteristics of EA projects be used to formulate a proper size measurement method?

RQ4. How can this size measurement method be used for effort estimations, and what is the accuracy?

1.4. Research Strategy

In order to achieve the research goals of the thesis, we implemented the design science research methodology. This methodology is described in detail in the Chapter 2. We implemented this methodology in three main stages, namely “problem identification”, “solution design”, and “evaluation”. This research methodology including the studies conducted is presented in Figure 3.

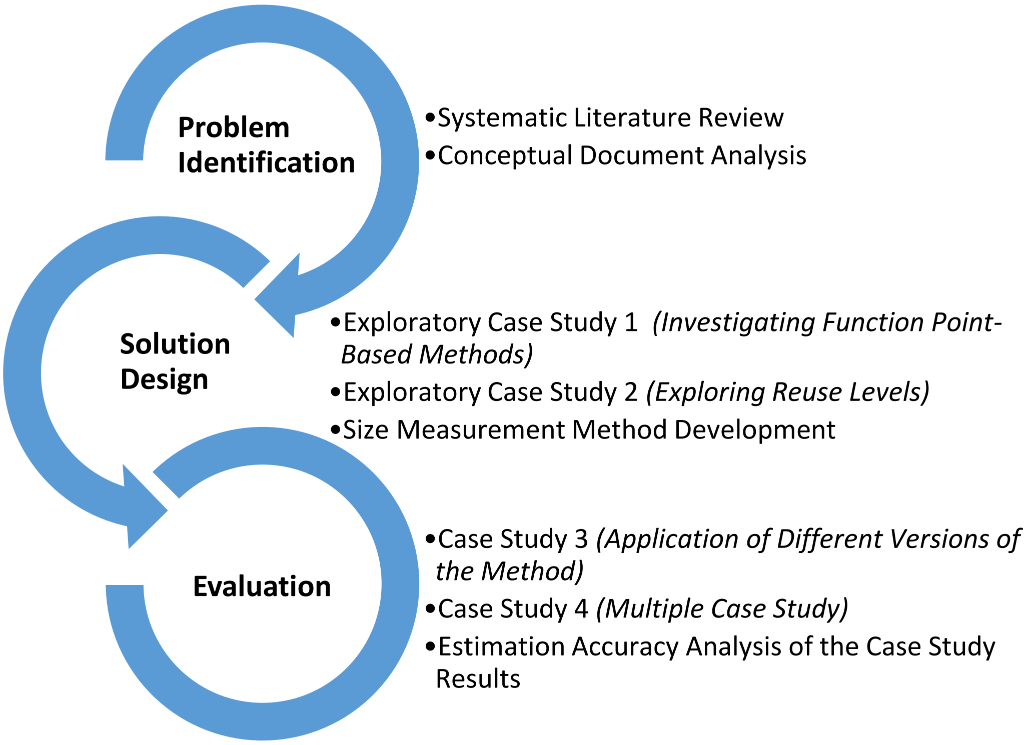


Figure 3. Research Methodology

In the problem identification stage, we performed a Systematic Literature Review (SLR) to answer research questions RQ1, RQ1.1 and RQ1.2. Although EA project is the general term covering different types of enterprise projects, the term ERP was often used in the field. Considering this, we conducted an SLR with a focus of ERP projects in the first place. Afterward, we extended our SLR search for other EA projects, such as SRM, CRM, etc., but we could not find significant studies for these terms. Through this SLR, we aimed to identify size measurement / effort estimation methods used for ERP projects and potential improvement areas of these methods. We used electronic data sources for this study. After performing a database search with defined keywords and two iterations of snowballing, we reached 69 related studies. We reviewed all these studies, excluded 28 studies according to defined inclusion/exclusion criteria. The remaining 41 studies were accepted as primary studies.

"Development of estimation method" was the most common research approach within these primary studies. In most of these studies, "Function Points" based size measurement methods were used for EA projects. "Business Blueprint", covering the requirement specifications of the projects, was the main source for Function Point Analysis (FPA). Validations of these methods were mostly done by using historic project data. Projects used in these validations mostly belonged to similar projects in the same company or industry.

When we examined the studies we had reached through the literature review, we saw that "size measurement / effort estimation" is a significant issue in this field. We found that a consensus could not be reached on how to measure the size or estimate the effort of such projects. We observed that analogy-based methods using old project data as the primary source for effort estimation of new projects are not very accurate for these types of projects. We inferred that the business process requirements of the new project should be taken as a basis for proper size measurement. At this point, the most accurate source for size measurement would be the "Business Blueprint" documents of the projects.

In the solution design stage, we conducted two different exploratory case studies to answer research questions RQ1.1, RQ2 and RQ3. In the first exploratory study, we analyzed existing size measurement methods applicable to EA projects. In the second exploratory case study, we explored the usage of reuse levels for measuring size of EA projects.

In the first case study, we applied three function point based methods published in the literature, namely "COSMIC EPC [33]", "COSMIC-FFP [31]" and "Estimation Strategies based on Function Points and Expert Judgments [29]". In these three methods, function points were used as a size unit for EA projects' effort estimation. We conducted the case study with an SAP Implementation project. In this project, seven different SAP modules were implemented for a mining company. We applied these methods for this project and evaluated these methods concerning their measurement processes and estimation errors. One of our conclusions from this study was that business processes were valuable resources for size measurement and effort estimation. We inferred that COSMIC FPA could be a candidate for size measurement of EA projects. We figured out two critical parameters as "modification" and "reuse" for the size measurement of EA projects. We concluded that, modification and reuse levels should be measured precisely, instead of using conversion factors with wide ranges.

In the second case study, we performed exploratory work to establish an approach to calculating reuse reflective size of EA projects. We explored whether it was possible to measure reuse reflective size for EA projects by using COSMIC function points as a base size unit and defining reuse levels. In this approach, we proposed measuring size in three steps. In the first step, the functional size of business processes was measured as COSMIC function points. In the second step, reuse levels of all business process steps were determined. In the last step, reuse reflective size was calculated by using these two parameters. The main sources for this approach were "Business Blueprint Document" and "Business Process Repository" of the EA. We were able to calculate the size of the project as COSMIC function points by using these business documents. We used four reuse levels defined and validated for EA projects by Daneva [20]. In order to reflect reuse levels in size measurement, we used change impact factors defined in the NESMA FSM method [34], and then calculated reuse reflective size for EA projects.

In order to evaluate our approach in a real-life setting and specify improvement opportunities, we applied this approach in an SAP Implementation project. We conducted three steps of the approach for the case study project and calculated the reuse reflective size of the project. To evaluate the effort estimation accuracy of the approach, we estimated the effort of the project using COSMIC size and reuse reflective size. MRE (Mean Relative Error) is calculated as 0.19 for reuse reflective size whereas it was calculated as 0.39 for COSMIC size where reuse was not considered. We concluded that size measurement, taking into account the reuse level, could lead to more accurate effort estimations. Reuse levels in our approach were valuable for the classification of business process requirements. However, we had concerns about using reuse levels and constant impact factors. Wrong reuse level selection would have led to entirely different size measurement results. Even if we had chosen the correct reuse level, we would not have been able to reflect the varying reuse amounts within the same range into the size measurement. We concluded that instead of determining "reuse level", we need to develop a size measurement method directly measuring "reuse" or "change" amount.

By considering all these inferences, we developed a size measurement method for EA applications. With this method, we proposed measuring "changes" where pre-built functionality was insufficient for the customer requirements. We counted "data groups" in transactions and measured "changes" in the transaction level. We claimed that the number of data groups handled for one transaction should be reflected in size. Change in a transaction with one data group would not have been the same as a change in a transaction with multiple data groups. We defined a new size measurement unit, Data Transaction Point (DTP), for EA projects. We listed all transactions and related data groups based on business processes, and for each business process, we measured size as DTP in the following three categories:

- *Unchanged:* total number of data groups in "no change required / used as-is transactions" are counted

- *Changed:* total number of data groups in "change / customization required transactions" are counted

- *New:* total number of data groups in "newly developed transactions" are counted

We proposed to determine change / customization requirements or new transaction requirements by using the "Business Blueprint Document" of the EA project. This proposed method has five main steps, as shown in Figure 4.

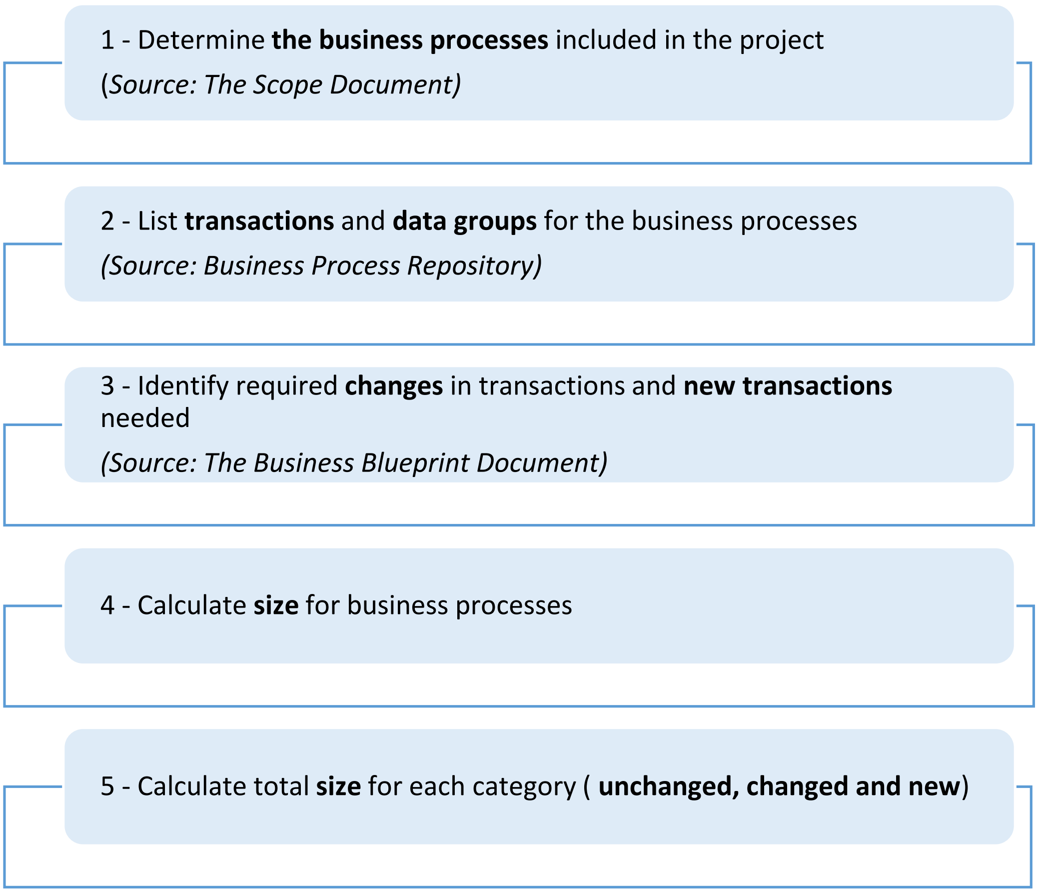


Figure 4. Proposed Size Measurement Method

We proposed three different versions of this method. In the "Quick Size Measurement Method" version, we counted the number of transactions in each category without considering the number of data groups in the transactions. The other two versions mainly differentiated based on how we counted data groups for the category "changed". "Size Measurement Method Version 1" was determined to count all data groups in the changed transactions, and "Size Measurement Method Version 2" was determined to count only the data groups affected by the transaction change.

In the evaluation stage, we conducted two different case studies to mainly answer research questions RQ3 and RQ4. In this third case study of the thesis, we aimed to analyze different versions of the proposed size measurement method. In the last case study of the thesis, we aimed to evaluate our size measurement method for EA projects.

We conducted the third case study to evaluate three versions of the size measurement method. The case project was an SAP Implementation project for a textile & chemistry company. Within the scope of the project, nine different modules of SAP were implemented for the company. Based on three different versions of our size measurement method, we identified the business processes, transactions, and data groups involved in the project, defined the necessary changes and new transactions, and measured the size for each module of the project.

We evaluated the effort estimation accuracy of different versions of the method by performing simple linear regression, multiple linear regression analysis, and the LOOCV (Leave One Out Cross Validation) method. We performed the simple linear regression analysis with the dependent variable "$\text{effort}_{\text{total}}$" and the independent variable "$\text{size}_{\text{total}}$". Then, we performed the multiple linear regression analysis with the independent variables "$\text{size}_{\text{unchanged}}$", "$\text{size}_{\text{changed}}$", and "$\text{size}_{\text{new}}$". We used Pearson's Correlation Coefficient [35] to measure correlation. Both the simple and multiple linear regression results showed that "Size Measurement Method Version 1" had the most predictive relationship, and the level of significance was within the acceptable range. We ran LOOCV for three versions and calculated MRE (Magnitude of Relative Error) with "$\text{effort}_{\text{actual}}$" and "$\text{effort}_{\text{predicted}}$" values. We then calculated MMRE (Mean Magnitude of Relative Error), MdMRE (Median of the Magnitude of Relative Error) and PRED (30). "Size Measurement Method Version 1", with the results respectively "0.21", "0.19" and "0.75", performed better than the other two versions for effort estimation.

These results showed us that the best version of the method for effort estimation was "Size Measurement Version 1", in which we counted all data groups of the changed transactions. This result seems reasonable since when we count only the data groups affected by the change, we could miss a part of the size. By counting all data groups, we measure the impact of change on a transaction basis. "Quick Size Measurement" version of the method may not have given good results compared to others, but it could still be used for early estimation.

We conducted the last case study as a multiple case study to observe the validity of the size measurement method. We aimed to evaluate the accuracy of effort estimations made with this size measurement method. Case study projects were two different SAP implementation projects. In the first project, ten modules of SAP were implemented for an oil & energy company. In the second project, nine modules of SAP were implemented for a manufacturing company. We measured the size of these projects by using our proposed size measurement method, "Size Measurement Method Version 1".

For effort estimation accuracy evaluation, we performed multiple linear regression analysis and LOOCV as in the third case study. Both projects' regression results showed a predictive relationship between size and effort. Linear regression models were acceptable for both projects, as evidenced by the significance level. We calculated MMRE, MdMRE, and PRED (30) values for both projects. For the first project, results were "0.21", "0.17", and "0.90" respectively and for the second project, results were "0.24", "0.19", and "0.78" respectively. Values less than or equal to "0.25" for MMRE and MdMRE are considered as acceptable for effort estimation in software engineering. Thus, for both projects, we had acceptable effort estimation results. PRED (30) is used for measuring estimation accuracy quality, and a value greater than 0.70 is acceptable. For both projects, we had values greater than 0.70, showing the estimation accuracy quality of the method.

In the last two case studies, we have had experience measuring size using “Scope” and “Business Blueprint” documents. These project documents were two valuable sources for our size measurement method. We could determine transactions, data groups, changes in the transactions, and new transactions needed as we defined in the size measurement method. These case studies show that our size measurement method is significantly better for EA projects.

1.5. Organization of the Thesis

Following this Introduction Chapter, the remaining of the thesis is organized as follows:

In Chapter 2, the research methodology used in this thesis, Design Science Research, is explained in detail.

Chapter 3 reviews related research regarding software size measurement, software effort estimation, basic concepts, and presents the current state of the art for size measurement of EA.

Chapter 4 presents two exploratory case studies that were performed to develop a size measurement method and discusses the results of the case studies.

Chapter 5 describes the size measurement method that we developed for EA projects. Different versions of the method and a sample size measurement are presented in this chapter.

In Chapter 6, two case studies that were performed to evaluate the size measurement method are presented. The results of these case studies are discussed in this chapter.

Chapter 7 summarizes the overall findings, their implications, limitations, and possible future work.

CHAPTER 2

RESEARCH METHODOLOGY

The primary purpose of this thesis is to develop a size measurement method for EA projects. Depending on this purpose, “Design Science Research” is determined as the main research methodology of the study. This chapter reviews the “Design Science Research Methodology”. In Section 2.1, the methodology is briefly explained. Section 2.2 presents the “Problem Identification Phase” of the methodology. In Section 2.3, the “Solution Design Phase” is explained. “Evaluation Phase” of the methodology is explained in Section 2.4.

2.1. Design Science Research

Two fundamental paradigms exist in Information Systems Discipline, namely “Behavioral Science Paradigm” and “Design Science Paradigm”. While the Behavioral Paradigm is more focused on developing and verifying theories to explain or predict human and organizational behavior, the Design Science Paradigm aims to expand the competencies of people and institutions by developing innovative and significant artifacts [36].

Hevner et al. [36] state that the Design Science Paradigm is a proactive paradigm from a technology standpoint, that focuses on creating and evaluating an Information Technologies artifact that aims to solve an important organizational problem. Offerman et al. [37] list the IT artifacts that are most frequently referenced when using a design science research methodology in software engineering as systems designs, methods, notations, algorithms, guidelines, requirements, patterns, and metrics. In this context, the size measurement method that is aimed to be developed within the scope of this thesis is compatible with the main output of the Design Science Research Paradigm.

Offerman et al. [38] formalized a detailed research process for Design Science Research consisting of three main steps: “Problem Identification”, “Solution Design”, and “Evaluation”. The problem identification step includes defining the problems that are practically relevant to the environment. The solution design step covers designing the required IT artifact. In the last step, the designed IT artifact is evaluated with the knowledge base or the environment. During Design Science Research activities, feedback loops frequently occur between these three steps, and in most of cases, “Solution Design” and “Evaluation” cycles are repeated many times until the needed solution is reached. We adopted the process defined by Offerman et al. [38] for our thesis study, as presented in Figure 5.

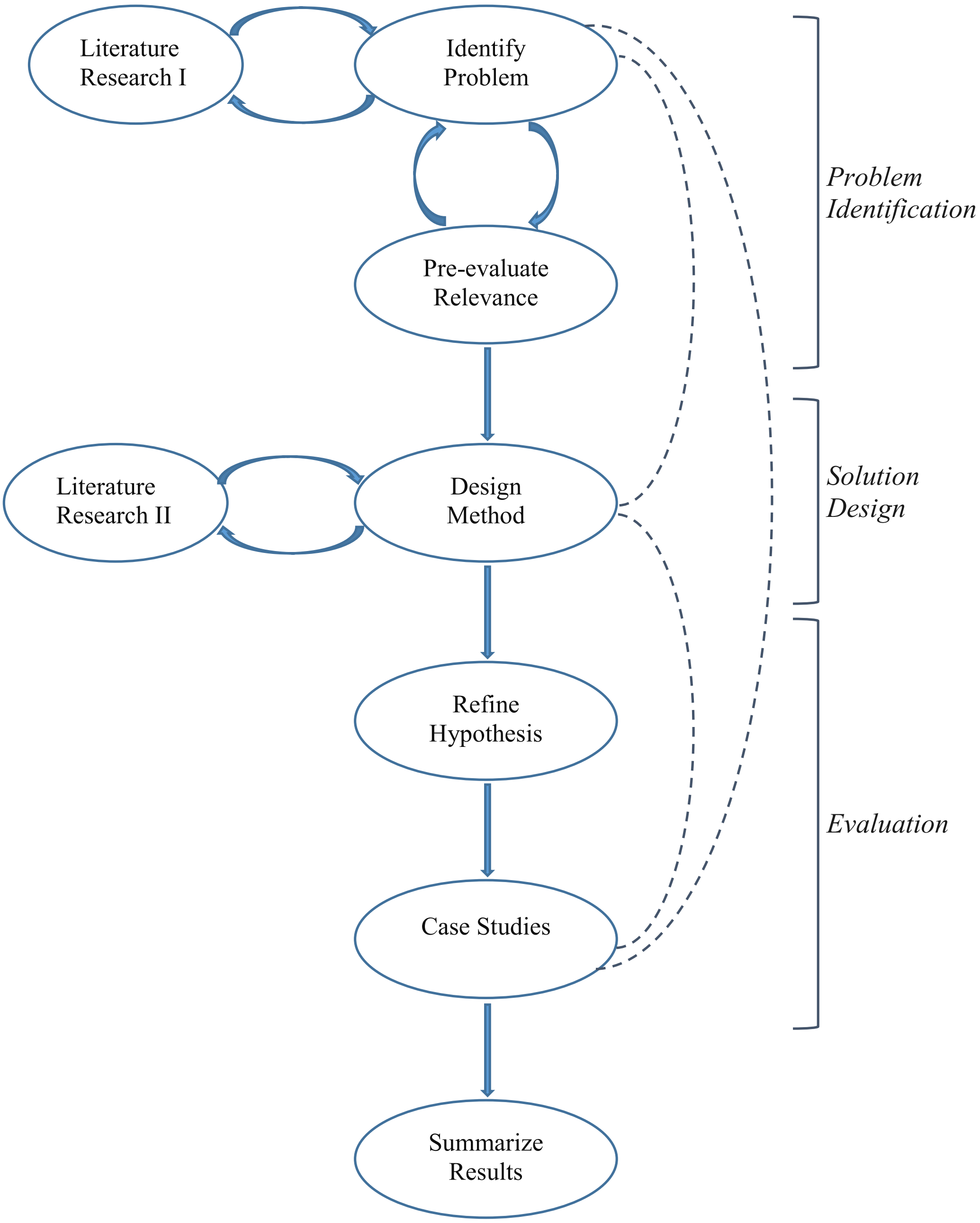


Figure 5. Design Science Research Process (Adapted from Offerman et al [38])

1.2. Problem Identification Phase

In this first phase of the research process, the main aim is to define a research question and evaluate its practical relevance. The most common tools used in this phase are literature reviews, expert interviews, and conceptual document analysis. In this thesis, we used conceptual document analysis and literature reviews for the problem identification phase.

Within the scope of conceptual document analysis, the software and documents of the system, project technical documents, other related documents, and online repositories can be examined. In this context, we examined documents for EA projects such as project management methodologies, project plans, project schedules, scope, and blueprint documents. We also analyzed project management tools and repositories provided by major EA vendors.

We conducted an SLR (Systematic Literature Review) to understand the size measurement methods utilized for EA projects and identify the problems and potential areas of improvement. With SLR, all the related studies done in the field were compiled systematically and the problems mentioned in the related literature would be revealed. We performed the review process according to the SLR method defined by Kitchenham [39].

Kitchenham [39] defines three main phases for an SLR, namely “planning the review”, “conducting the review” and “reporting the review”. In this study, the following main stages are listed as distributed over phases.

1. The need for a review is identified.

2. The review protocol (research questions, search terms, resources, study selection criteria, study quality assessment procedures, and data extraction strategy) is developed.

3. The research strategy is identified.

4. Primary studies are selected based on selection (inclusion & exclusion) criteria.

5. The quality of primary studies is assessed.

6. Data extraction forms are designed and recorded.

7. The results of the included primary studies are consolidated and summarized.

8. The review is reported mostly in a technical report, journal or conference paper.

For the research strategy, Wohlin [40] suggests forward and backward snowballing searches besides the main database search. In the main database search, a search is conducted with defined keywords. In the backward snowballing search, new studies are found by checking the reference list of the primary studies. In the forward snowballing search, studies that cited the primary studies are found. In our SLR, we used these three kinds of search iteratively until no new study was found.

1.3. Solution Design Phase

The main activities conducted in the solution design phase are developing the artifacts, exploring possible solutions in a controlled environment, and exploring the knowledge base for possible solutions. As stated by Offerman et al. [38], artifact design is not a standard process; it is a creative engineering process without so much guidance.

Literature reviews and case studies are processes that support the solution design. Within the scope of this thesis, we searched the literature and performed exploratory case studies iteratively to develop the size measurement method.

Runeson and Höst [41] state that a case study is a well-suited research methodology for software engineering research, as it examines modern phenomena in their natural context. They define the key characteristics of a case study as:

1. It is a flexible type to cope with the complex and dynamic features of real-world events.
2. Its conclusions are based on a clear chain of evidence, qualitative or quantitative, collected from multiple sources in a plan and consistency.
3. It improves existing knowledge by basing it on a pre-established theory, if any, or by building a new theory.

Yin [42] describes a typical case study process as shown in Figure 6.

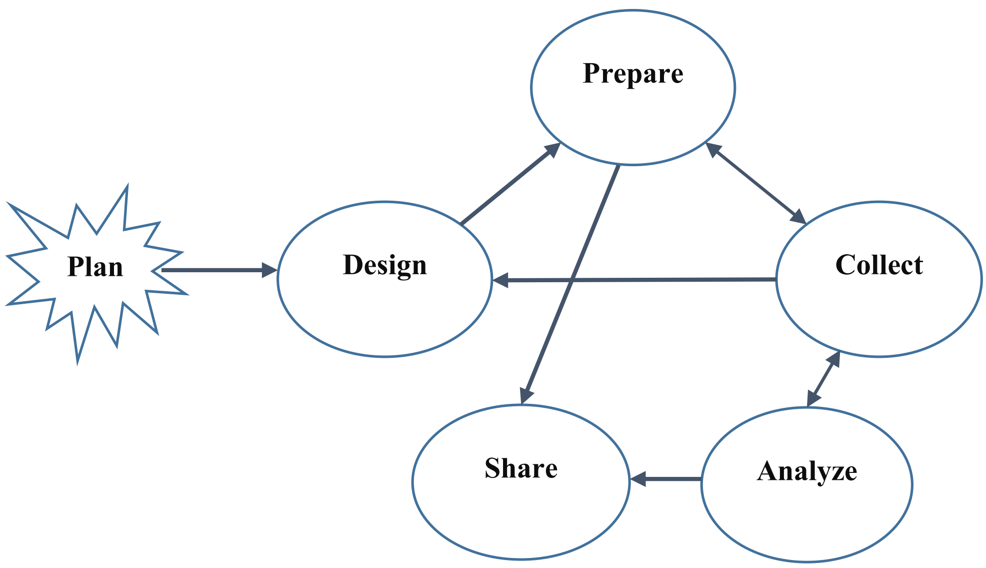


Figure 6. Case Study Method (Yin, [42])

As shown in the figure, Yin [42] defines the case study process as a linear but also iterative process. In this study, the steps of the defined case study process are summarized as follows:

- *The planning step*

  A decision is made on whether to implement the case study. The rationale for conducting the case study is clarified, and the suitable research questions are determined.

- *The design step*

  The cases to be studied are determined. The design of the case study (single or multiple, holistic or embedded) is made. A set of procedures are defined to ensure the quality of the case study.

- *The preparation step*

  Necessary training is taken to perform the case study. The case study protocol, mainly including the purpose of the protocol, data collection procedure, and an outline for the case study report, is developed. Pilot studies are carried out, and relevant approvals are taken.

- *The collection step*

  The case study protocol developed in the preparation step is applied. By using the evidence obtained from different sources, the case study database is created. The evidence chain is maintained.

- *The analysis step*

  Collected data is analyzed based on theoretical propositions and other strategies. Using qualitative and/or quantitative data, analytical techniques are applied, and the data is interpreted.

- *The sharing step*

  The target audience for the presentation of the results is determined. Texts and visual materials are created. The findings are presented with sufficient evidence for the reader to make their own deductions.

In the context of software engineering, case study research methodology can take the form of different research methodologies with different purposes. Runeson and Höst [41] claim that case study methodology can be used for exploratory, descriptive, explanatory and improving purposes in the software engineering discipline. Robson [43] defines these four types of research purposes as follows:

- *Exploratory research* is used to uncover what is going on, search for new insights, and generate ideas and hypotheses for new research.

- *Descriptive research* is used to describe a situation or phenomenon.

- *Explanatory research* is used to seek an explanation for a situation or a problem. In most of cases, these explanations are in the form of cause and effect relationships.

- *Improving research* is used to develop a particular aspect of the phenomenon being studied.

In addition to these types, as Yin [42] states, case studies can be conducted as multiple case studies rather than a single case study. Yin [42] explains that multiple case studies create a stronger research design with repetition, and with this feature, they can be applied to increase the reliability of the case study.

Within the scope of this thesis, we both used exploratory and improving case studies. Exploratory case studies were conducted during the development of the size measurement method in the solution design phase. After we developed the size measurement method, we used an improving case study to evaluate different versions of the method and identify potential improvement opportunities. Finally, we conducted a multiple case study to validate the size measurement method we developed.

1.4. Evaluation Phase

The main activities conducted in the evaluation phase are evaluating possible solutions in a controlled environment, evaluating the proposed solution, and contributing the solution to the knowledge base. Evaluation can be performed by tools such as case study, action search, expert survey, laboratory experiments, and simulations [38]. In this context, we conducted two different case studies in the evaluation phase.

By using the size measurement results of these case studies, we performed simple linear regression and multiple linear regression analysis. In the simple linear regression analysis, we used "$\text{effort}_{\text{total}}$" as the dependent variable and "$\text{size}_{\text{total}}$" as the independent variable. In the multiple linear regression analysis, three different size categories as, "$\text{size}_{\text{unchanged}}$", "$\text{size}_{\text{changed}}$" and "$\text{size}_{\text{new}}$" were used as independent variables. De Santo et al. [44] defines "linear regression" as a statistical technique used to explain or predict how a dependent variable behaves. Single and multiple linear regression equations are presented in Figure 7.

Simple Linear Regression

$$y = \beta_0 + \beta_1 x$$

Multiple Linear Regression

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots\cdots + \beta_k x_k$$

y = dependent variable

x = independent variable

β_0 = intercept of the line/ hyperplane

β_i= slope coefficient for the relevant independent variable

Figure 7. Linear Regression Equations

In regression analysis, the coefficient of determination (R^2 or R-Squared) is used to measure the strength of the relationship between the regression model and the dependent variable. Humphrey [45] proposes the following criteria for regression models in software engineering for planning purposes:

- $0.9 \leqslant R^2$; a predictive relationship
- $0.7 \leqslant R^2 < 0.9$; a strong relationship
- $0.5 \leqslant R^2 < 0.7$; an adequate relationship
- $R^2 < 0.5$; not reliable for planning

After assessing the coefficient of determination, we should determine if the result is significant; the significance level shows the probability of finding the correlation by chance [45] [46]. If the significance level is less than 0.5, models are mostly admitted in an acceptable significance range.

For justification of the applicability of the linear regression models, we used residual plots. These plots show the difference between the estimated and actual values of the dependent variable. If the data points in a residual plot are randomly scattered, the linear regression analysis is suitable for the data set; otherwise, a non-linear model would be used [47].

In order to evaluate the estimation accuracy of the size measurement method, two different data sets are needed, namely “training” and “test (validation)”. Results of the “training” data set are used to estimate the “test” data set. Due to the structure of EA projects, we have limited number of measurement data. Assessing the size measurement method is complicated in these conditions. Kocaguneli and Menzies [48] propose using LOOCV (Leave One Out Cross Validation) to ensure consistency and repeatability for this kind of small sample size. In LOOCV, a single observation from the data set is used as "test" data, and the remaining n-1 observations are used as "training" data. In each iteration, the test data is changed in sequence. Iterations are done as many as the number of cases in the data set.

Estimation results obtained from LOOCV are used to evaluate the estimation accuracy of the model. MRE (Magnitude of Relative Error) is the base unit for evaluations. MMRE (Mean Magnitude of Relative Error), MdMRE (Median of the Magnitude of Relative Error) and PRED(n) are the most used estimation accuracy evaluation metrics in software engineering discipline [49]. Equations to calculate these metrics are presented in Figure 8.

$$\mathrm{MRE} = \frac{|\mathrm{EA} - \mathrm{EP}|}{\mathrm{EA}}$$

$$\mathrm{MMRE} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{MRE}_i$$

$$\mathrm{MdMRE} = \mathrm{median}\ (\mathrm{MRE}_i)$$

$$\mathrm{PRED}\ (x) = k/n$$

MRE = Magnitude of Relative Error

MMRE = Mean Magnitude of Relative Error

MdMRE = Median Magnitude of Relative Error

EA = Effort $_{actual}$ *EP = Effort* $_{predicted}$

k = Number of estimations where MRE is less than or equal to x.

n = Total number of estimations

Figure 8. Estimation Accuracy Evaluation Equations

Some threshold values are recommended in the literature for these estimation accuracy evaluation metrics. In one the main studies in software engineering, 0.25 was defined as a threshold value for MMRE and MdMRE; values less than or equal to 0.25 were accepted as successful estimations [50]. Regarding PRED, "PRED (30)⩾ 0.60" is defined as good estimation accuracy [51]. In the studies [52] [53], "PRED (30)⩾ 0.70" is proposed as a more reliable accuracy threshold. We used these threshold values for the estimation accuracy evaluation of the case studies.

CHAPTER 3

RELATED RESEARCH

This chapter reviews the literature on size measurement of Enterprise Applications (EA). Section 3.1 presents current software size measurement methodologies, their weaknesses, and strengths. Section 3.2 presents the Systematic Literature Review we performed for size measurement/effort estimation methods used for ERP projects. Section 3.3 presents a brief discussion of the current state-of-the-art for size measurement of EA.

3.1. Software Size Measurement

Size measurement is one of the most critical processes of software project management. Ozkan et al. stated that functional size is the primary input for effort, cost, and schedule prediction, and also project scope changes can be measured using functional size [16]. They also figured out many other contributions of functional size to software project management, as being used as a software acquisition unit and normalizing base project measures such as performance and quality [16].

The Lines of Code (LOC) is the first measure used for software size. As stated in the name, size measurement is done by counting the lines of source codes of the software. It was the only software size measure in the 1970s and is still in use. It is easily applied and can be automated, but it is not suitable for size measurement at the requirements elicitation phase of the project.

In order to solve size measurement based project management problems, several size measurement/estimation approaches have been developed over the years [17]. As stated in the study [17], approaches based on measuring functionality for software size measurement have been widely used.

In 1979, Albrecht made the first definition of Function Points(FP) and Function Point Analysis(FPA); the main idea was to measure the software size as a unit of "functionality" provided to the user [54]. After Albrecht, different kinds of functional size measurement methods have been developed based on this idea. Today, IFPUG FPA (ISO/IEC 20926:2009,[55]), COSMIC FPA (ISO/IEC 19761:2011, [56]), NESMA FSM (ISO/IEC 24570:2018,[57]) and MkII FPA (ISO/IEC 20968:2002, [58]) are widely used FPA methods which are certified as an international standard by ISO. All these FSM methods measure size as FP using FUR (functional user requirements), but they differ by measurement style and measured units [17]. A brief history of functional size measurement methods is presented in Figure 9.

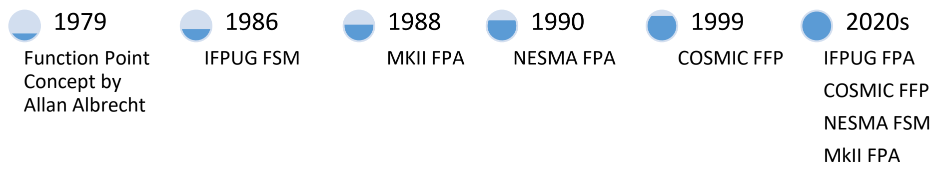


Figure 9. History of Functional Size Measurement Methods

These functional size measurement methods can also be used for size measurement of software maintenance, i.e., estimation of software changes. In the study [59], it was reported that COSMIC FPA could be used to measure software changes. They achieved an effort estimation accuracy of 10 to 20 percent for software changes of a client's project by measuring size as COSMIC FP. In the study [60], a model using both algorithmic and non-algorithmic approaches to estimate the effort of an EA upgrade project was proposed. This model was validated with three different types of EA, namely ERP, CRM, and HRM.

3.2. Systematic Literature Review

ERP projects constitute a major part of EA projects, and although the inclusive term is EA, the term ERP is often used in the field. Taking this into account, we conducted an SLR for ERP projects in the first place. Afterwards, we searched the literature for other EA projects, such as SRM, CRM etc., but we could not find significant studies.

We performed the Systematic Literature Review (SLR) to gain an understanding of size measurement / effort estimation methods utilized for ERP projects and to identify the potential areas of improvement. Results of this SLR was presented at "Euromicro SEAA 2017 Conference" [15].

The review process was performed according to the SLR method defined by Kitchenham [39]. We defined a research protocol including research questions, data source, and a main method to perform the review.

3.2.1. Research Questions

We performed SLR to figure out all existing studies in the ERP size measurement / effort estimation domain, their limitations, and open issues for future studies. Accordingly, we used the following research questions in the SLR:

RQ1. What type of research approaches are applied for ERP size measurement / effort estimation studies?

RQ2. Which size measurement / effort estimation topics were analyzed for ERP projects?

RQ3. Which size measurement / effort estimation methods were developed or suggested for ERP projects?

RQ4. What inputs, outputs and prediction models are used in these effort estimation methods?

RQ5. How are these effort estimation methods validated?

3.2.2. Data Source

We have used an electronic data source for this study. Electronic database used as the data source is listed in Appendix-A. Furthermore, we used both forward and backward snowballing to extend the main database search as Wohlin suggested [61].

Search strings were constructed firstly by breaking down the research topic "ERP Size Measurement and Effort Estimation". Since SAP is the most widely used ERP worldwide, SAP was used interchangeably with ERP. Words in ERP project management terminology such as RICE (report, interface, conversion, extension) Objects and Business Blueprint were also used as search strings. Synonym words were also used in constructed search strings. Full list of search strings is presented in Appendix-A.

After performing the database search, abstracts of the articles were read, and included articles were identified. After two iterations of snowballing, no new articles were found to add the research result articles' list. Distribution of papers based on the search method is presented in Table 2.

Table 2. SLR Search Method

Search Method	**# Articles**
Database Search	31
Backward Snowballing	4
Forward Snowballing	6

3.2.3. Study Selection

In order to assess potential primary studies, we used inclusion and exclusion criteria for study selection. The main inclusion criterion of our review was that paper described "research on size measurement / effort estimation for ERP projects". Studies related to complexity, and size/cost factors were only included if the main purpose of the study was to improve ERP size measurement/effort estimation.

According to our research purpose, papers not directly related to ERP size measurement/effort estimation and overlapping papers describing the same study were excluded. Furthermore, papers having missing information in the definition of the methods or validation process were also excluded.

69 studies were reviewed, and 28 were excluded according to defined inclusion/exclusion criteria. The remaining 41 studies were accepted as primary studies.

3.2.4. Data Extraction

In order to answer research questions, two different kinds of characteristics were used for data extraction from papers. Firstly, characteristics applicable to all papers, as listed in Table 3, were extracted. Characteristic "research approach" was used relying on the work presented in the study [40].

"Research approach" and "estimation topic" characteristics may have multiple values since a study may investigate/apply more than one estimation topic or may be performed based on more than one research approach.

Table 3. Classification of the Papers

Characteristic	**Definition / Categories**
Research Approach	Theory, Survey, Experiment, Case Study, Development of Estimation Method, History-based Evaluation, Own Experience, Real-life Evaluation, Review, Simulation, Other
Estimation Topic	Main estimation topic/topics addressed in the paper

This classification of papers gave a general view of ERP research studies. The second classification was performed for a deep investigation of corresponding studies. For each study that applies the "development of estimation method" as a research approach, characteristics listed in Table 4 were also extracted.

Table 4. Classification of the Estimation Papers

Characteristic	**Definition/Categories**
Inputs	Input parameters, such as function points and line of code and project characteristics used in estimation
Outputs	Output of the estimation method, such as person-hour, person-month
Prediction Model	Prediction model applied to input parameters for effort estimation
Data Set	Data set used for estimation method validation
Descriptive Statistics	Quantitative analysis of data for studies applied "history-based evaluation" research approach

3.2.5. Results of the SLR

We found those 41 primary studies in journals, conference proceedings, and as also technical reports.

Distribution of papers based on research approaches is illustrated in Figure 10. According to this distribution, the research approach "development of estimation method" was the most common research approach for this research topic. In this research approach, estimation methods were developed/suggested and evaluated. Novelty of method is not essential in this research approach; the suggested method may be a combination of existing estimation methods. This result shows that most of the studies in this review aimed to develop an appropriate size measurement / effort estimation method for ERP projects. Analysis of existing methods seems to be mostly performed by applying the research approach "history-based evaluation".

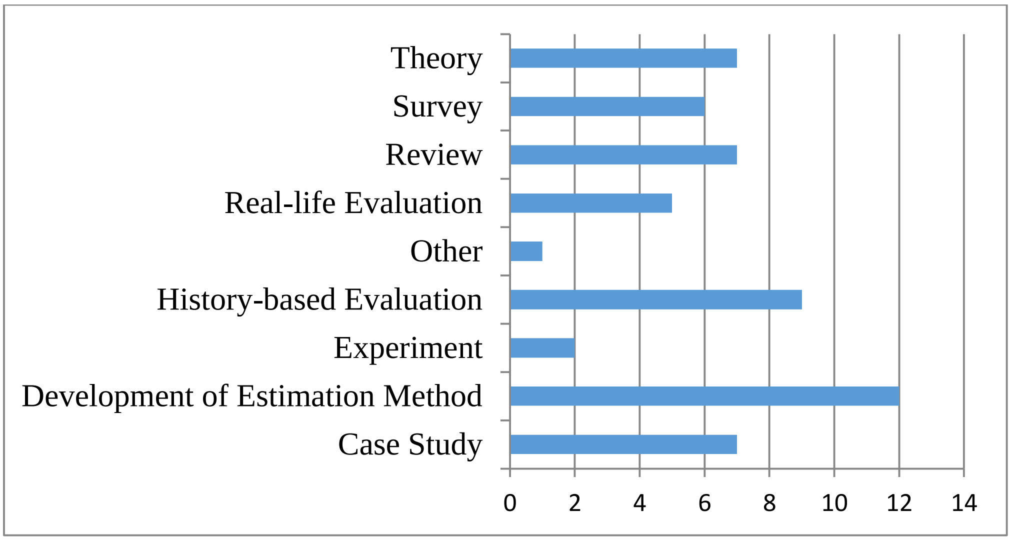


Figure 10. Distribution of Papers based on Research Approaches

All primary studies were listed based on estimation methods they applied as listed in Appendix-A. According to this distribution, most of the studies seemed to be performed for analysis and validation of "cost drivers" in ERP projects.

12 studies that applied the "development of estimation method" research approach were analyzed in more detail based on defined characteristics in Table 4. 10 of these estimation studies were directly explaining ERP effort estimation, and remaining 2 studies were explaining mainly ERP size estimation.

The distribution of inputs of these studies is shown in Appendix-A. According to this distribution, "Function Points" seems to be the most common input parameter applied in these estimation studies. None of the papers use the same prediction model for effort estimation. Prediction models used in these studies and descriptive statistics for their prediction models are listed in Appendix-A.

Based on these descriptive statistics, studies [62][33] seem to be candidate effort estimation methods for ERP projects. In the study [62], validation was done using historical data for 39 ERP projects. This historical data was gathered via questionnaires filled out by Austrian companies that already implementing ERP. Within this data set, a considerable amount of enterprise was from the same industry. This approach depends on historical data for similar projects. This low MMRE value could result from making estimation for similar projects. As the authors stated in [62], the estimation performance of this social choice method heavily depends on the similarity of the new project to be estimated to the projects in the data set. If the new project is an outlier, effort estimation would not be within target.

Study [33] also has a low MMRE value. In this study, effort estimation was done by using a new functional size measurement method called “COSMIC EPC”. Validation of the method was performed using 9 SAP projects. The author claimed that if only the business processes were available for the project, this method performed better than other function point-based estimation methods, as confirmed by MMRE values.

3.2.6. Discussions of the Studies

The first study discussing that traditional effort estimation methods were not appropriate for industrial projects such as ERP projects was Stensrud [14]; he argued that ERP projects should be handled differently since they included a large variety of metrics to estimate the cost. They made an empirical study using the analogy tool ANGEL and linear regression analysis. Their conclusion of this study was that analogy tools and multiple linear regression analysis were appropriate effort estimation methods for these kinds of projects, especially because of their flexibility in choosing input parameters.

In 2001, Stensrud [14] stated that ERP projects should be estimated using multi-dimensional project size measures, not only function points or LOC. In this study, possible ERP size measures were stated as the number of ERP modules, users, software interfaces, sites, business units, EDI interfaces, data conversions, custom-developed reports, and modified screen. Existing effort estimation methods were analyzed based on applicability to ERP projects, and added value to human users.

After Stensrud’s study [14], 40 different studies were performed to solve ERP effort estimation problem. Daneva, Maya, and Roel Wieringa [6] and Daneva, Maya [63] performed similar studies to Stensrud [14] by analyzing existing effort estimation methods’ applicability to the ERP domain. The study [64] was also another review study investigating function points, size and cost estimators.

Most of the studies were interested in cost factors for ERP projects. An extensive survey was conducted for Swiss SMEs implementing ERP to figure out ERP cost drivers [65]. More than 4000 Swiss SMEs were contacted during one year for applying survey. Survey results were analyzed using descriptive statistics and correlation analysis. This analysis validated that the “number of modules to be implemented” was a significant cost driver. A new cost driver, “consultant experience” was found based on analysis. However, the “number of users”, which was assumed to be a major cost driver in those days, could not have been validated as a cost driver by this study.

Daneva made qualitative research to figure out existing state-of-the-art for size and effort estimation processes, especially for cross-organizational ERP projects [63]. In this research, an empirical study was performed with asynchronous focus groups, including representatives from ERP vendors, consultancy firms, advisory firms, and adopter companies. Based on the results, it was concluded that there was a confusion within the representatives for the definition and measurement of the size. However, if the size is used in the project, it was mostly measured as functional size.

Daneva [25] continued her studies by developing a solution approach for the ERP effort estimation problem. In this study, she suggested using a portfolio management perspective for effort estimation. Proposed solution was a combination of existing effort models as COCOMO II, Monte Carlo simulation, and a portfolio management model. This ERP effort estimation solution approach was validated by a case study performed in a company that had implemented 13 ERP projects. It was concluded that this approach achieved an increased probability of success, especially for projects with high uncertainty. 4 other studies [66] [67] [68] [69] were performed to investigate the suggested COCOMO II, Monte Carlo simulation, and Portfolio Management Model combination for ERP projects’ effort estimation.

In another study, Wilson proposed to use RICE objects for ERP size and effort estimation [23]. She developed a statistical model for size and effort estimation by using RICE objects. Linear regression analysis performed for validation of the models showed a strong correlation between RICE objects and ERP software development effort. The study [31] was another study in which RICE objects were again statically analyzed.

Erasmus and Daneva continued their study on ERP effort estimation by analyzing “Expert Judgment” method [28]. They claimed that with new database technologies such as SAP HANA (in-memory database), it is possible to construct ERP Suite including BI (Business Intelligence) software in one platform. They believed that this all-in-one platform solution resulted in more uncertainties at the early levels of the ERP projects. Thus, these uncertainties and resulting customizations could not be estimated by one parametric approach. In their study, the “Expert Judgments” method was analyzed by action research. Main conclusion of this research was to introduce multiple checkpoints to avoid Expert Judgments’ estimation bias.

In 2015, Erasmus, Pierre, and Maya Daneva performed two studies [29][30], suggesting that a combination of Function Points and Expert Judgments would be the most proper effort estimation method for evolving ERP domain.

Five more studies directly addressed Function Point usage for ERP projects’ effort estimation. Vogelezang made an experiment on COSMIC-FFP for ERP project size and effort estimation [31]. He used the “refined approximate COSMIC-FFP” method to make size estimation in the early stages of the ERP project. This size value is used for effort estimation by using historical productivity rates.

Martín Téllez, Francisco [32] proposed a new function point approach as “the eEPC-COSMIC approach” in his thesis report. In this study, a methodology was developed to calculate COSMIC data movements from extended EPC diagrams. Erasmus [33] also performed a comprehensive study that proposes a new ERP effort estimation method called “COSMIC EPC”, using business process models for defining COSMIC function points.

Daneva investigated the reuse measurement process for function point calculation in her projects [20] [70], which were based on IFPUG for function point calculation. Last study addressing Function Points was Kuijpers [71]; he suggested using system components in a reusable way to calculate IFPUG function points automatically.

3.3. A Brief Discussion of the State of the Art

Based on the analysis we performed in SLR, in most of the studies, cost factors specific to the ERP domain were analyzed and their correlation were validated with historical data sets. None of the studies with the estimation approach “development of estimation method” suggested the same estimation method. However, three of these studies used function points as input to a prediction model. Validations of these effort estimation methods were mostly done by a history-based validation research approach. Datasets used in this validations mostly belonged to similar projects within same company or industry.

Enterprise Applications are evolving based on customer and industry expectations year by year. Thus, size measurement of Enterprise Applications should be done using the customer requirements of each new project. Function point approaches based on blueprints were widely used for size estimation in this domain. Since Enterprise Applications evolve rapidly, size measurement method in this domain should be easy enough to be applied by even a novel user. Size measurement methods developed for this domain should take into account high reuse rates of these projects.

CHAPTER 4

DEVELOPMENT OF A SIZE MEASUREMENT METHOD (EXPLORATORY CASE STUDIES)

This chapter presents two exploratory case studies we performed to answer our research questions. In Section 4.1, the case study analyzing function point-based effort estimation methods is presented. Section 4.2 presents the second case study that was performed to explore reuse reflective size for EA projects. In Section 4.3, the implications of these two case studies are discussed.

4.1. Case Study 1 – Measurement based on Function Points

The main objective of this research study is to analyze studies in the literature in which EA effort estimation methods using function points, as size input, are developed. To achieve this objective, we conducted a case study applying three main function point-based EA effort estimation methods which we found out in our SLR study [15]:

- The first method is the effort estimation method called “COSMIC EPC”, which was proposed by Erasmus [33]. This method uses business process models for defining COSMIC function points. In this study, an add-on functionality for this calculation was also developed for SAP project management tools.

- The second method is COSMIC-FFP. Vogelezang did an experiment on COSMIC-FFP for projects’ size and effort estimation [31]. He used the “refined approximate COSMIC-FFP” method to estimate size in early stages of an EA project. In this method, processes are categorized based on their complexity, and total COSMIC function points are calculated using pre-defined function points of categories. This size value is then used for effort estimation using historical productivity rates.

- The third method is a combination of Functional Size Measurement and Expert Judgments. Erasmus and Daneva performed a study [29] suggesting that instead of using specific effort estimation methods for every situation, integrating different estimation strategies based on Function Points and Expert Judgments would be more appropriate for the projects. They emphasized that this will leverage the strengths of both methods.

We presented the results of this case study at “IWSM MENSURA 2017” [72].

4.1.1. Description of the Case

This case project was an SAP Implementation Project. The project was started in April 2015 and completed in March 2016. Go-Live of the project occurred on 01.01.2016; the last 3 months of the project was for operational support and maintenance.

This project consisted implementation of SAP modules that are listed in Table 5.

Table 5. Case Study 1- SAP Modules Implemented

SAP Module Implemented	**Module Description**
MM	Material Management
SD	Sales & Distribution
PM	Plant Maintenance
FI	Financial Accounting
CO	Controlling
HR	Human Resource
PP-PI	Production Planning – Process Industries

The project staff consisted of 1 Project Manager, 7 Senior SAP Consultant, and 5 Junior SAP Consultants from SAP Consultancy Company; 1 Project Manager, 3 Process Analysts, and 7 SAP Key-Users from SAP Adopter Company.

The case study company implementing SAP is a Turkish mining company with three mine sites located in different cities of Turkey. Total number of employees for the company is almost 1200; at the beginning of the project, total number of SAP users was predicted as 110.

SAP Adopter (Consultancy) Company performing this project, with over 1500 employees, is one of the leading IT companies in Germany. It is also a partner of SAP AG since 2000. This project was conducted by the Turkey office of this SAP Adopter Company.

Business Blueprint documents defining requirement specifications were obtained from the SAP Adopter (SAP Consultancy) Company. Business Blueprint Documents have three main sections as “Organizational Units”, “Master Data” and “Business Processes”. Business processes of the project were figured out by analyzing these Blueprint documents.

SAP Adopter Company collects effort data on a daily basis at SAP CATS Time Sheet Module. A project document, including realized effort values based on project phases, modules, and consultant experience levels, was also obtained from the company. The efforts utilized for this project are presented based on the project phase and modules in Table 6.

Table 6. Case Study 1- Efforts Utilized

Project Phase	**Effort (Person-Hours)**
Business Blueprint &Infrastructure	*616*
SAP Basis	108,5
Controlling	226
Finance	78
Human Resources	59
Logistics	59,5
Sales & Distribution	32
Service & Energy App.	53
Customization & Development	*1230,5*
SAP Basis	85,5
Mobility	85
Controlling	263
Finance	130
Human Resources	373,5
Logistics	90,5
Sales & Distribution	62
Service & Energy App.	141
Integration & Go Live	*470,75*
SAP Basis	16,25
Mobility	5,5
Controlling	131,5
Finance	77
Human Resources	144
Logistics	74,5
Sales & Distribution	8
Service & Energy App.	14
Operation & Support	*683,75*
SAP Basis	37
Mobility	7,5
Controlling	272
Finance	46,75
Human Resources	196
Logistics	29,5
Sales & Distribution	8
Service & Energy App.	87
Total	***3001***

4.1.2. Application of the Method 1 – The Cosmic EPC Method

The COSMIC EPC method converts business processes to COSMIC function points, then to effort using conversion factors. Erasmus [33] defines the method in detail by also illustrating with a sample SAP business scenario.

Business Blueprints for MM, SD and PM Modules were read to figure out which business scenarios & processes were applied in this project. Business processes were searched in the Business Process Repository of SAP Solution Manager. Related business processes (level 3- business process steps) exported to Microsoft Excel to make effort calculations. A sample view of the business processes' list is in Figure 11.

- Procurement Scenario
  - Supplier Master Data
    - Create Supplier
    - Extend Supplier master data
    - Display Supplier master Record
  - Contract Processing in ERP
    - Create contract
    - Maintain authorized business partner
    - Create contract items
    - Maintain target and estimated values
    - Determine and maintain texts
    - Perform credit check
    - Determine and process message output
    - Monitor contract fulfillment

Figure 11. Case Study 1 - Business Process List Sample

The data movements (Entry, Exit, Read and Write) were counted on the lowest level of detail, on the process step level (Level3). Basically; the total functional size for a business process is calculated by summing up all related data movements. This size value is then updated using parameters as "Reuse" and "Modify". All parameters used in the effort calculation are described in Table 7.

Table 7. Case Study 1 - Calculation Parameters [33]

Parameter	**Description**
Level	It shows the process detail level: ▪ Level 1 = Business Scenario ▪ Level 2 = Business Process ▪ Level 3 = Process Step
Module	It shows the ERP module that the process belongs to.
Include	It is used to indicate whether to include the process step or not, "#SubPr" tab is filled accordingly: ▪ "YES" => # SubPr = 1 ▪ "NO" => # SubPr = 0

Parameter	Description
Modify	It is used to indicate that either business process requires customization or not. If "YES" is selected for any one of the process steps of a business process: ▪ Modify (business process) = "YES" ▪ Data movements value for business process is multiplied by #included process steps
Reuse	It shows if the process step is already implemented in another business process. If "YES" is selected, all the related data movement fields for that process step will be reset to value "0".

All these parameters were stated for process steps in the calculation sheet. A sample view from the calculation sheet is in Figure 12.

Business Process	Level	Module	Include	Reuse	# SubPr	Modify	Entry	Exit	Read	Write	Total
2.2.8. Monitor contract fulfillment	3	MM	YES	YES	1	NO	1	1	1		0
2.3. Processing purchase requisitions	2				4	YES					36
2.3.1. Create purchase requisitions	3	MM	YES	NO	1	YES	1			1	2
2.3.2. Release purchase requisitions	3	MM	YES	NO	1	NO	1			1	2
2.3.3. Assign source of supply to purchase requisitions	3	MM	YES	NO	1	NO	1			1	2
2.3.4. Generate versions of purchase requisitions	3	MM	NO	NO	0	NO	1			1	0
2.3.5. Monitor list display of purchase requisitions	3	MM	YES	NO	1	YES	1	1	1		3
2.4. Processing purchase orders	2				6	YES					90
2.4.1. Create purchase orders	3	MM	YES	NO	1	YES	1			1	2
2.4.2. Release purchasing documents	3	MM	YES	NO	1	NO	1			1	2
2.4.3. Find new source of supply	3	MM	YES	NO	1	NO	1			1	2
2.4.4. Compare quotations	3	MM	YES	NO	1	NO	1	1	1		3

Figure 12. Case Study 1 - Calculation Sample

Based on this calculation, the total size of the project for MM, SD, and PM modules was calculated as 886 Cfs (COSMIC functional size) points. In this method, effort (scenario implementation time) calculation is conducted based on the equation presented in Figure 13. In this calculation, the EA vendor provides scenario template time, representing the standard functionality without customization. The conversion factor is determined based on the complexity of the scenario.

$$SI\_Time = ST\_Time \times SUC\_CF \times (Cfs / ST\_fs)$$

SI_Time = Scenario Implementation Time

ST_Time = Scenario Template Time

SUC_CF = Standard Unit Cost Conversion factor

Cfs = COSMIC functional size of the project data

ST_fs = Scenario Template functional size

Figure 13. Case Study 1 - COSMIC EPC Effort Formula [33]

Template values for “time to implement” and “total functional size” were calculated without taking into account “include”, “reuse”, and “modify” values effect.

- ST_fs = 247 Cfs
- ST_time = 165 person-hours

The conversion factor was decided as 0.9 based on the complexity of the project. Related conversion factors are shown in Table 8.

Table 8. Case Study 1 - Conversion Factors

Conversion Factors for Implementation Engineer					
Level	***Level1***	***Level2***	***Level3***	***Level4***	***Level5***
Conversion Factor	1,6	1,1	0,9	0,75	0,6

By considering these values, implementation time (effort) calculation is as follows:

$$SI\_Time = 165 \times 0.9 \times (886/247) = 532 \text{ person-hours}$$

Thus, the total effort for MM, SD, and PM modules implementation for this project is calculated based on COSMIC EPC as 532 person-hours.

Realized value for these modules was 378.5 person-hours; with this values MRE (Mean Relative Error) is calculated as 0.40.

4.1.3. Application of the Method 2 – COSMIC-FFP

This method emphasizes that usage of process-chains during the implementation of the project will influence effort value. A process-chain is defined as a set of business functions handling specific events of business processes. In our case project, all project is handled in one process chain; thus, we made calculations accordingly.

For making effort estimation in the early stages of the project, an approximation method, “approximate COSMIC-FFP” is developed in this study. The effort is calculated based on both “COSMIC-FFP” and “approximate COSMIC-FFP”.

Based on “approximate COSMIC-FFP”, processes should be classified in one of the categories in Table 9.

Table 9. Case Study 1 - Categories for “Approximate COSMIC-FFP” [31]

Category	***Cfsu***	***Description***
Small	4	retrieval of information about a single object of interest
Medium	7	storage of a single object of interest with some checks

Category	*Cfsu*	*Description*
Large	11	retrieval of information about multiple objects of interest
Complex	24	

This method is then called “refined approximate COSMIC-FFP”. It is stated in the study that the COSMIC function units in this table are taken from Measurement Manual and thought to be 20-30% precise, although they are environment dependent.

Based on these definitions, the total size of the project for MM, SD and PM modules is calculated as follows:

COSMIC-FFP:

- Size = 247 Cfsu

Refined Approximate COSMIC-FFP:

- Size = 384 Cfsu

In this study, the time to delivery of a project is correlated exponentially with the size of the project. According to this method, total effort (in months) correlates exponentially with the project’s size. The effort formula is defined as shown in Figure 14.

$$\mathbf{Time}_{\text{Delivery}} = \frac{\mathbf{Size}^{\mathbf{Power}}}{\mathbf{PL}}$$

where:

Time	= Time to delivery of the process-chain in months
Size	= Functional size in Cfsu
Power	= 0,20 for a single production line and 0,37 for two production lines
PL	= Number of deployed production lines (1 or 2)

Figure 14. Case Study 1 - COSMIC FFP Time Calculation Formula [31]

Realized effort for these modules was 2.15 person-months (considering 8 hours of working daily) for our case study project. Effort and MRE based on this realized effort are calculated as follows:

Based on COSMIC-FFP:

- $\text{Time}_{\text{Delivery}} = \frac{247^{0,2}}{1}$ = 3 person-months
- MRE = 0.39

Based on Refined Approximate COSMIC-FFP:

- Time $_{Delivery} = \frac{384^{0,2}}{1}$ = 3.3 person-months
- MRE = 0.53

4.1.4. Application of the Method 3 – ERP Services Effort Estimation Strategies

This study emphasizes that no one estimation strategy is appropriate for every situation. Instead of selecting one strategy for the estimation, it is suggested to integrate different strategies considering the topic, resources, and situation of the project.

SAP projects are mainly divided into two different types “Innovation & Ramp-up Projects” and “RDS (Rapid Deployment Solutions)”. Innovation & Ramp-up projects are projects where the solution is modified based on customer requirements. However, in RDS projects, solutions are pre-engineered and implemented with a small customization.

According to this definition, our project in this case study is considered “Innovation & Ramp-up Project”.

In this study, three main estimation strategies are defined as “Baseline”, “Configurable”, and “Tailored”. Those requirements and activities representing the implementation of core functionalities of ERP are counted as “Baseline”, and estimation could be based on FSM data. In “Configurable” estimation strategy, the estimator could rely on configurable estimates or rules of thumb since these kinds of customization scenarios are repeatable customizations requiring fine tuning to a certain degree. “Tailored” estimation strategy should be used mainly for activities related to customizations unique to customer-specific processes. It is emphasized that these strategies could also be used both for first estimations and validating estimations.

An ERP Services Estimation Strategy Matrix, as shown in Table V, is developed to assist the estimator in figuring out which estimation strategy, methods, and material to use according to components and project. For each matrix column, the parameters chosen for our case study are colored blue, as shown in Table 10.

Table 10. Case Study 1 - The ERP Services Estimation Strategy Matrix [29]

1	Estimation Strategy	Baseline	Configurable	Tailored
2	Project Type	Time & Material (Innovation & Ramp-Up Projects)		
		Rapid Deployment Solutions (Engineered Projects)		
3	Implementation	Baseline	Customization	
4	Configuration Component	Core	Optional	Specific
5	Estimation Component	Activities and Sub-activities	Pre-configured Scenarios	Expert Judgment
6	Knowledge Base	Explicit		Tacit
7	Effort Estimation	Function Point & FSM Estimation	Expert Judgment	
8	Estimation Reference	Work Breakdown Structure Template	Rules of thumb	Experience & Expertise

After defining the estimation strategy, a validation process is recommended to be applied for the estimations. According to the filled Estimation Strategy Matrix for our case study, effort estimation could be done based on both FSM Estimation and Expert Judgment.

A sample validation method is explained in the study for three scenarios having historical project estimates. In this validation method, effort based on the COSMIC FP is calculated based on COSMIC EPC method defined by Erasmus [33]. It is recommended to use the historical rule of thumb if we could not relay on FSM since it has a frequency count of less than 3.

We could rely on FSM to implement MM, SD and PM modules since these are modules reused and repeated frequently. In our case study project, there is only one scenario seldom implemented, which is related to the PP-PI module specific to the mining industry. For this kind of scenario, strategy tailored is used and validation is done based on the rule of thumb, as illustrated in Table 11.

Table 11. Case Study 1 – Validation [29]

	FSM (COSMIC FP – CFP)			**Rule of thumb (Expert Judgment)**		
Scenario	***CFP&Effort***	***Frequency Count***	***Effort Variance***	***Rule of thumb (hours)***	***Frequency Count***	***Effort Variance***
Batch Management	null	null	null	Range (70-120 Hours)	2 (seldom used)	null

SAP Adopter Company implemented this scenario two times; thus, we filled the frequency field as 2. Expert of this module estimates implementation of this scenario has an effort range of 70-100 hours. Realized effort for this scenario was 87 person-hours, which is in the rule of thumb estimation range.

4.1.4. Discussions of the Results

The COSMIC EPC method requires business processes that could be easily obtained from the SAP Solution Manager Business Process Repository. However, deciding on the parameters requires a high level of module knowledge, and thus could not be performed by novel consultants.

In COSMIC EPC, reuse is not related to previous projects' developments; it counts reuse only within the same project. This may result in an overestimation of effort in case of a high level of reuse. Another problematic point in this method is the complexity parameter. The complexity of the project is only evaluated by deciding conversion factor with a range of 0.6 to 1.6. A faulty decision on this parameter will completely change the effort value.

Modification of one process step and multiple process steps have the same effect on calculation, the sum of data movements will be multiplied by the total active process steps in that business process. The logic behind this is explained as modification of any process step will affect the whole business process. However, modifying one business step and whole business steps of a business process will not result in the same effort. Thus, the modification level does not seem to reflect effort value properly in COSMIC EPC.

Based on case study calculations, approximate COSMIC-FFP size estimations seem to be a good alternative, especially when no detailed business process data is available. This method could be applied for very early estimates of EA projects especially during signing contracts phase. In the COSMIC-FFP method only parameter affecting effort value is the number of production lines. Two projects using the same number of production lines seem to have same productivity based on this method. Effort estimation is not influenced by the project's reuse, modification, or complexity rates.

Service Estimation Strategy seems to be critical for especially projects where solutions are tailored based on customer requirements. These kinds of projects do not have historical productivity rates, so Expert Judgment is suggested to be used for effort estimation. By using estimation strategies, it is claimed that expert bias could be reduced.

In our case study, three modules were all reused and repeated SAP modules and scenarios. Thus, we could apply the estimation strategy concept defined in [29] for only one scenario specific to the mining industry. The rule of thumb effort value for this scenario was in a wide range. So, relying on rule of thumb would result in high estimation errors as in our case study. FPA-based effort estimation methods should be enhanced to objectively estimate these kinds of not reused, not repeated scenarios.

4.2. Case Study 2 – Exploring Reuse Levels

Enterprise Applications have pre-built functionalities that mostly fulfill customer requirements. From a software development perspective, it can be considered as reusing most of the pre-built functionality. We assume that as reuse dominates the development, reuse rates should be reflected size measurement for proper results. Based on this assumption, we targeted to develop a size measurement approach for EA projects using reuse reflective size of the projects. To establish such an approach, we first need to calculate the size of the project and then calculate the amount of reuse.

As a first step, we conducted a case study to understand if COSMIC function points can be used as a size measure and if we can calculate reuse levels by analyzing available documents as a base of such an approach. We conducted a case study on a completed SAP Implementation project to explore the potential. Using the business blueprints, we calculated COSMIC functional size and reuse reflective size for this project. We also used the calculated size in an existing effort estimation method and compared the results with the actual effort. We presented the results of this case study at "Euromicro SEAA Conference 2018" [73].

4.2.1. The Approach

Enterprise Applications are in a growing industry with new products, and solutions according to customers' expectations. In this evolving environment, producing solutions that cost millions of dollars, it does not seem feasible to rely on analogy-based size measurement methods. Instead, we need to work on more precise size measurement methods. Based on our systematic literature review study [15], "Function Points" was the most common input parameter used in ERP effort estimation studies. FPA can be measured once the requirements are known, even though there are estimation methods to be used in earlier phases and based on business processes. Furthermore, FP has many uses in the software project management processes such as monitoring scope change, early size estimation, normalizing performance, and quality measures [16]. Thus, we assumed FPA could be the most likely size measure for a potential EA size measurement method.

Reuse is one of the most critical parameters that should be taken into account for EA effort estimation. In an EA project, most of the customer requirements can be fulfilled by EA pre-built functionalities; in some cases, reuse rates of 80% are possible [20]. Thus, we might figure out the relation between reuse levels and size, and design a size measurement method that accounts for the relative cost of reuse.

Therefore, we suggest that an approach can be built primarily on two variables: first, we need the FP count, and secondly, we need to calculate the amount of reuse. Business Blueprint Documents could be the primary source for FPA calculation since it contains all details related to customer requirements. A Business Blueprint Document has mainly three parts “Organizational Units”, “Master Data” and “Business Scenarios”. Especially by analyzing the “Business Scenarios” part, one can list business scenarios and processes that will be implemented in the project and requirements related to changes in business processes.

In EA project management tools, “business scenarios” are defined as including related business processes and process steps. For example, for an SAP Implementation Project, one can define all included business processes and process steps for a project by analyzing Business Blueprint Document and SAP Solution Manager Business Process Repository. We assume that we can calculate the size of the project as COSMIC function points by using these business documents.

We used reuse levels defined and empirically validated by Daneva [20]. She classified reuse in four levels in her study:

- *Level 3:* Processes and data components are completely reused in this reuse level. There is no need to perform any kind of change in this reuse level.
- *Level 2:* It refers to minor enhancements applied to process and data components. “Minor enhancement” means changing a certain parameter of a business process or data component. This parameter change should not cause a change in the process logic.
- *Level 1:* This category refers to major enhancements applied to processes and data components. “Major enhancement” means conceptual level changes applied in process logic; the definition of the process or data component is modified at the code level.
- *No Reuse*: In fact, this is not a real reuse level; this level refers to completely new developments, and reuse is not applied at all.

With this reuse level categorization, we can define reuse levels at process step levels based on the Business Blueprint Documents. However, we need a method to calculate the functional size of the project by taking into account these reuse levels, and we need to reflect the reuse rate in the size calculation. In studies [74] [75], a method is proposed to calculate similarity reflective functional size, which could be used for effort estimates. This similarity reflective functional size methodology could be applicable for EA projects’ size measurement; the main difference is that we do not need to calculate internal (within the project) reuse, instead, we need to calculate external reuse, i.e., how much pre-built functionality of an EA will be used in the project.

In these studies [74] [75], two types of functional similarity reflective size calculation methods are introduced; continuous functional similarity reflective size (CS) and discrete functional similarity reflective size (DS). The discrete similarity reflective functional sizes are calculated by using constant functional similarity percentage values, which were extracted from the software enhancement approach of NESMA FSM method [57]. In NESMA, functional size is multiplied by an impact factor depending on the amount of change. For DS calculation, the amount of change of NESMA is considered as the amount of functional similarity and impact factors as functional similarity percentage constants. We adapted this reflective size estimation method to EA projects using reuse levels Daneva [20] proposed. We defined functional similarity percentage constants for calculation as shown in Table 12.

Table 12. Case Study 2 - Similarity Percentage Constants

Reuse Level	Reuse Level Description	Highest Similarity Percentage Value	Functional Similarity Percentage Constants
Level1	major enhancements applied to reference processes and data components	max<=34 %	0,25
Level2	minor enhancements applied to reference processes and data components	0,67<max<1,0	0,75
Level3	process and data components that were reused without any changes	max=1,0	0,1
no-reuse	new development	max=0	1

Note: In Nesma, 0,34 < max <= 0,67 similarity value is assigned to “0,50” as functional similarity percentage constant. Since we have 4 levels of reuse definition in the EA field, and these definitions do not fit with this similarity range, we did not use this value for reuse assignment.

We calculate reuse reflective size of an EA project in three steps as shown in Figure 15.

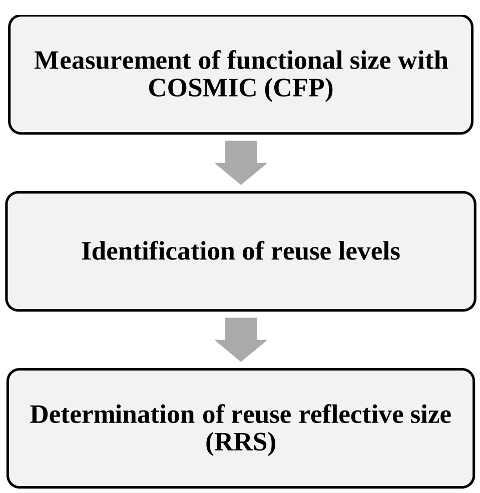


Figure 15. Case Study 2 - Reuse Reflective Size Measurement

We perform these three steps for all business processes in business process structure of the project. Reuse Reflective Size calculation formulas, which are mainly adapted from DS, are presented in Table 13.

Table 13. Case Study 2 - Reuse Reflective Size Measurement

Reuse Level	***Reuse Reflective Size Measurement Formula***
Reuse Level 3	RRS = CFP*reuse overhead
Reuse Level 2	RRS = (CFP*0.25) + (CFP*0.75*reuse overhead)
Reuse Level 1	RRS = (CFP*0.75) + (CFP*0.25*reuse overhead)
No-reuse	RRS = CFP

To calculate the reuse reflective size, we also need to identify "reuse overhead" value for EA projects. There are no studies for identifying reuse overheads in EA projects. As a first attempt, we use reuse overhead as "0.10" as given in the DS methodology.

After calculating reuse reflective size for all business processes, we sum up the total reuse reflective size of the project. This reuse reflective size could be used as a primary input for effort estimation of EA projects.

4.2.2. Case Study Design

To explore the applicability of our reuse reflective size measurement approach in a real-life setting and to figure out potential improvement opportunities for the approach, we have conducted a case study with an SAP Implementation project. We measured the COSMIC functional size of the project, defined reuse levels, calculated the reuse reflective size of the project, and used this size value in an existing effort estimation method. This case study was designed to answer three questions:

- How can COSMIC function points be used as a size measure for EA projects?
- How can reuse levels be calculated for EA projects?
- Is reuse a valuable input for EA effort estimation? How does reuse affect EA effort estimation?

We used the same SAP Implementation Project that we analyzed in Case Study 1. We measured the size of MM, SD, and PM modules of the project.

4.2.3. Application of the Approach

The Business Blueprint Document is the main source for requirement analysis in this case study. By reading the “Business Processes” part and using SAP Solution Manager Business Process Repository, we could list all business scenarios & processes & process steps applied in this project.

After creating the business process structure, we defined the COSMIC data movements (Entry, Exit, Read and Write) on the process step level. Basically, the total functional size for a business process can be calculated by summing up all data movements of included business steps. A sample view for calculation is shown in Figure 16.

Business Process	Module	Include	Reuse Level	Func. Similarity Constant	Total CFP	Reuse Reflective Size
1. Sales Scenario		YES				
1.1. Master Data Governance for Customer		YES				**2,575**
1.1.1. Create customer	SD	YES	Level2	0,75	2	0,65
1.1.2. Search for customer	SD	YES	Level3	0,1	3	0,3
1.1.3. Display customer	SD	YES	Level2	0,75	3	0,975
1.1.4. Process customer via change request	SD	NO				0
1.1.5. Approve changes	SD	NO				0
1.1.6. Change customer	SD	YES	Level2	0,75	2	0,65
1.2. Contract Processing in ERP						**8,55**
1.2.1. Create contract	SD	YES	Level2	0,75	2	0,65
1.2.2. Create contract items	SD	YES	Level2	0,75	2	0,65
1.2.4. Display contract	SD	YES	Level2	0,75	3	0,975
1.2.4. Maintain target and estimated values	SD	YES	Level3	0,1	2	0,2
1.2.5. Determine and maintain texts	SD	YES	Level1	0,25	2	1,55
1.2.6. Perform credit check	SD	YES	Level2	0,75	2	0,65
1.2.7. Determine and process message output	SD	YES	Level1	0,25	2	1,55
1.2.8. Monitor contract fulfillment	SD	YES	Level1	0,25	3	2,325

Figure 16. Case Study 2 - Reuse Reflective Size Measurement Sample

Based on this calculation, the reuse reflective size of the project for MM, SD, and PM modules was calculated as 112.55 COSMIC function points, whereas total size was calculated as 247 COSMIC function points without taking into account functional similarity.

4.2.4. Effort Estimation with Reuse Reflective Size

By applying the approach, the reuse reflective size value is almost half of the standard size. Is this a valid result? To understand this, we used this reuse reflective size in an existing effort estimation method suggested by Vogelezang [31]. According to this method, total effort (in months) correlates exponentially with the size of the project. The effort formula is defined as shown in Figure 17.

$$\text{Time}_{\textbf{Delivery}} = \frac{Size^{\text{Power}}}{\text{PL}}$$

$\text{Time}_{\text{Delivery}}$= Time to delivery of the process-chain in months

Size = Functional size in Cfsu

Power = 0,20 for a single production line; 0,37 for two production lines

PL= Number of deployed production lines (1 or 2)

Figure 17. Case Study 2 - COSMIC-FFP Effort Formula [31]

This method emphasizes that usage of production lines during the implementation of the project will influence effort value. A "production line" is defined as a set of business functions handling business processes' specific events. In our case project, all project is handled in one production line; thus we used PL as "1" and power as "0.20".

The realized effort of these modules for our case study project was 2.15 person-months (considering 8 hours of working day). We calculated effort and corresponding MRE for "Size without considering reuse" and "Reuse Reflective Size":

Based on COSMIC FP (without considering reuse):

Time Delivery = $[247^{0.2}]$ /1 = 3 person-months

MRE = 0.39

Based on Reuse Reflective Size:

Time Delivery = $[112.55^{0.2}]$ /1 = 2.57 person-months

MRE = 0.19

4.2.5. Discussions of the Results

The first objective of the case study was to evaluate the applicability of COSMIC FPA for EA projects. We observed that Business Blueprint Document is a proper source for function point analysis. We could calculate COSMIC function points for the SAP Implementation project by analyzing the Business Blueprint Document and using the SAP Solution Manager Business Process Repository. SAP Solution Manager provides content for all SAP applications in the form of business scenarios, business processes, and process steps. Instead of creating the Business Blueprint from scratch, it can be created in SAP Solution Manager by using Business Process Repository. This will shorten the total time required for CFP calculation.

The second objective was to explore reuse level calculation based on Business Blueprint Document. This document includes all necessary details to understand if minor or major enhancements are required for related business processes. Reuse level definitions are clear for enhancements; we could easily decide on reuse levels for the SAP Implementation Project.

Reuse levels used in our approach are valuable, especially for classification of requirements. We used these levels to decide on functional similarity constants that will be used in the effort estimation formula. Instead of using reuse levels and constant rates, we could analyze if it would be possible to measure reuse rate more precisely for EA projects.

Reuse overhead is used as 0.10 for reuse reflective size calculation in this case study. For an accurate result, reuse rates for EA projects should be calculated for a set of projects. Reuse rate could differ based on module or business scenarios. Further empirical studies are required to explore the effects of the reuse overhead.

For the last objective of the case study, the effort estimation result by using reuse reflective size is much better than the results obtained by using standard size as CFP. As depicted, the MRE for effort estimation with Reuse Reflective Size is much better.

We observe that making more reliable effort estimates could be possible by calculating size using COSMIC, defining reuse overhead value, calculating productivity for the EA domain, and measuring reuse rate precisely.

4.3. A Brief Discussion of the Case Studies

In Case Study 1, we analyzed three research papers that all validated the developed effort estimation method for a specific EA vendor, SAP. Our case study is also performed for an SAP project. SAP Business Blueprint document is a good source for figuring out business processes. We conclude that the SAP Business Blueprint document can be used for size measurement purposes.

All three studies suggest using COSMIC FPA for size estimation. Requirements based on business processes are analyzed to define COSMIC data movements. COSMIC EPC [33] takes into account also “reuse” and “modification” parameters to calculate the final size value. This size value is then converted to effort using conversion factors based on complexity. Approximate COSMIC-FFP [31] also uses COSMIC function points for size estimation and converts this size value by conversion factors based on the number of production lines. Service Estimation Strategy [29] has a different approach; it suggests COSMIC EPC for function point based estimation, but if business scenarios are not reused and repeated frequently, Expert Judgment based on Rule of Thumb is also suggested. This study does not directly offer one effort estimation method for all types of business scenarios; instead, it provides an estimation matrix that presents estimation strategies based on situation, historical data availability, and knowledge base.

These three studies show that business processes are valuable resources for the size measurement. COSMIC FPA could be a good candidate for size measurement of EA projects. EA project tools such as SAP Solution Manager are currently used also for Business Blueprint document generation. These tools could be used for automatically calculating the size and effort of an EA project based on Business Blueprint documents.

Based on our case study, it can be concluded that the size estimation by the COSMIC-FFP method is very rough. The number of production lines, the only parameter affecting effort value, was 1 for our case study. Thus, the productivity rate for any project with the same number of production lines would be same as our case study. COSMIC-EPC has a detailed effort estimation method with modification, reuse, and complexity parameters. This method counts reuse if only reuse is within the same project. Reusing the pre-built functionality of the EA is not taken into account in this method. The main difficulty we faced during applying this method was selecting the complexity conversion factor. This conversion factor has a range between 0.6 to 1.6; thus, the decision of this factor affects effort value entirely.

ERP Service Estimation Strategy matrix is valuable for integrating different strategies considering the project's topic, resources, and situation. However, it is not easy to obtain the proper rule-of-thumb estimations, especially for new customer-specific requirements. In our case study, the rule-of-thumb value for the scenario was in a wide range which could result in high estimation errors.

We concluded from the first case study that critical parameters for function point-based EA size measurement methods would be "modification" and "reuse levels". Modification and reuse levels should be defined precisely. Instead of using conversion factors with wide ranges, methods should rely on exact measurements.

In Case Study 2, we introduced an exploratory approach for "calculating reuse reflective size of EA projects", which could be used for effort estimation. We applied this approach in the same SAP Implementation project to evaluate it in a real-life setting and identify improvement opportunities.

Based on the approach, the reuse reflective size of an EA project is calculated in three steps; COSMIC function points' calculation, reuse levels definition, and reuse reflective size calculation. For calculating COSMIC function points, we used the Business Blueprint Document of the project and the SAP Solution Manager Business Process Repository. Reuse levels are calculated by analyzing the "Business Processes" part of the Business Blueprint. After calculating reuse reflective size, this size value is used for effort calculation. We calculated reuse reflective size for our case study project. We used this size in an effort estimation method suggested for EA projects in the literature. MRE was calculated as 0.19, whereas it was calculated as 0.39 when reuse was not taken into account for size calculation.

Based on our second case study, it can be concluded that COSMIC size measurement and reuse reflective size could be a valuable input for effort estimation. We observed that the size calculation could be applied to EA projects. Since EA project tools such as SAP Solution Manager are capable of generating Business Blueprint documents, these tools could also be used for size measurement of an EA project automatically.

Although this exploratory case study shows promising success, we had some concerns about using rates for reuse levels. Choosing the wrong reuse level would result in a completely incorrect measurement. We concluded that we need to develop a size measurement method that directly measures “reuse” for EA projects.

CHAPTER 5

THE SIZE MEASUREMENT METHOD

This chapter presents the size measurement method that we developed for EA projects. In Section 5.1, the size measurement method is described in detail. In Section 5.2, three different versions of the developed method are explained. In Section 5.3, sample size measurement of a business process with the proposed method is presented in detail.

5.1. The Method

Based on systematic literature review and case studies' conclusions, we attempted to develop a size measurement method for EA projects. The critical difference of these projects, which would affect the size measurement approach, is "high reuse rates". "Having high reuse rates" means most of the customer requirements are provided by pre-built functionality of an EA. Thus, we need to measure "changes" where pre-built functionality is insufficient for the customer requirements.

For measuring "changes", we propose to count "data groups" used in each "transaction". We defined a special size unit, namely ***"Data Transaction Point (DTP)"*** as a measure of EA projects' size. DTP is a size unit showing the number of *data groups handled by transactions*; in other words, it shows data points of a transaction. A change in a transaction containing one data group does not have the same effect on size, as a change in a transaction containing multiple data groups. We assume that if a transaction has a large number of data groups, it will have a greater impact on the size as the change in that transaction will likely be reflected across all data groups.

We list data groups and transactions based on business processes. For each business process, we measure the size for three categories:

- *Unchanged:* For the business process, data groups in "no change required / used as is transactions" are counted

- *Changed:* For the business process, data groups in "change / customization required transactions" are counted

- *New:* For the business process, data groups in "new developed transactions" are counted

For all these categories, data groups based on transactions are summed up to reach the total size of the business process as DTP.

This proposed size measurement method has five main steps, which are presented in Figure 18.

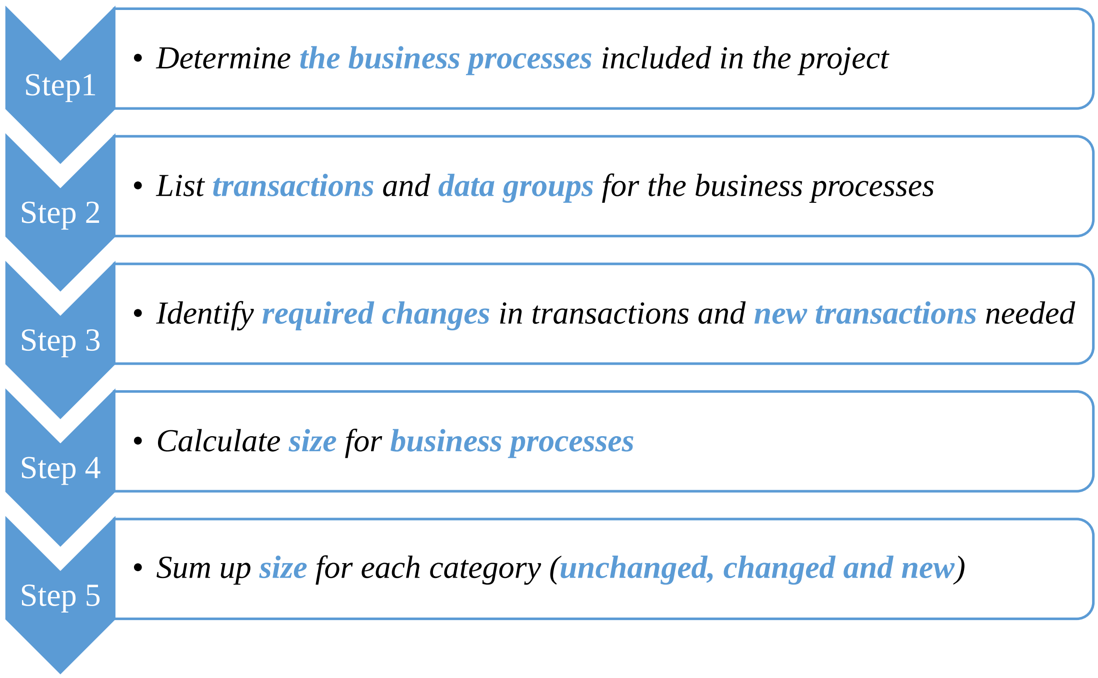


Figure 18. Steps of the Size Measurement Method

5.1.1. Step 1 - Determining Business Processes

In an EA project, a number of business scenarios are applied to meet the customer requirements. A business scenario contains many business processes. A business process is defined as a collection of operations taking a specific business input and converting it into a valuable business output via a number of transactions [76].

Business scenarios and related business processes that will be applied in the project are decided in the scope determination phase of these kinds of projects. As the first step of size measurement, the *"Project Scope Document"* is reviewed and *"Business Scenarios"* and *"Business Processes"* are listed for the project as presented in Figure 19.

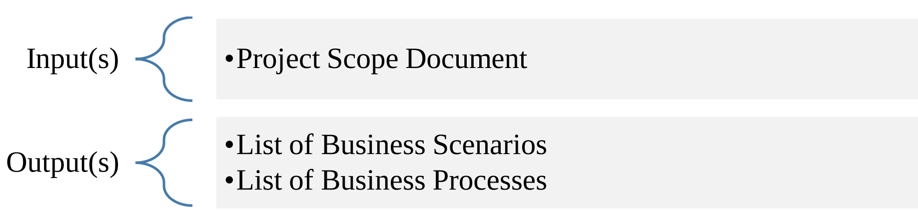


Figure 19. Determining Business Processes

5.1.2. Step 2 - Defining Transactions and Data Groups

The second step of the size measurement is listing *"Transactions"* and *"Data Groups"* included in the business processes. For this step, the *"Business Process Repository"* of the EA could be used as the resource. Inputs and outputs of this step are presented in Figure 20.

Figure 20. Defining Transactions and Data Groups

In EA terminology, a *"transaction"* is basically defined as executing a program; transactions can be called via system-defined / user-specific transaction codes or can be invoked by other programs [77]. While many programs in EA are run by users with transaction codes, some programs are run by periodic jobs, and some programs are called by programs that provide integration with other systems. For example, a report created in the system can be sent to users via e-mail by a program periodically, or data can be retrieved or transmitted via integrated systems through a program. Since in all of these programs, data groups are processed to meet business process requirements, we consider them as transactions, regardless of whether they are run with a transaction code by users.

In EA, each transaction uses a number of data groups to perform the business request. The general definition for a "*data group"* is "a unique, non-empty, non-ordered group of data attributes, explaining the same one object of interest"[56]. EA handles data in two main categories: master data and transactional data. Master data is permanent data, containing information for customers, suppliers, materials etc., whereas, transactional data is temporary data containing information for sales orders, invoices etc.[76]. We define the *"data group"* for the EA as covering both data categories.

We can explain the data group concept for EA through an example. In the "sales order processing" business process, there are basically transaction codes such as "create sales order", "change sales order", "display sales order". Four main data groups as "Customer", "Material", "Pricing" and "Sales Order" can be defined for these transactions. "Customer", "Material" and "Pricing" are master data that contain necessary information to create a sales order. "Sales Order" is the transactional data that occurred during the execution of the transactions. Consider a business requirement such as "If the customer is a foreign customer and the price of the material is below 100 $, set the status of the sales order as released". In order to apply the necessary customization for this requirement, these four different data groups would be handled. Thus, we count all of these data groups in the size calculation.

5.1.3. Step 3 - Identifying Required Changes and New Transactions

In an EA project, based on customer requirements, some transactions are used as is and some transactions are used by applying customization or code changes. If customer requirements cannot be met by pre-built functionality of the EA, new transactions can be developed using programming language of the EA.

In this step of the size measurement, *change required transactions*, *new transaction needs* and *related data groups* should be figured out. Main input of this step is “The Business Blueprint / Functional Requirement Document” of the EA project. By examining in detail the Business Blueprint / Functional Requirement Document of the project, it should be determined which category (unchanged, changed, and new) each transaction belongs to. Inputs and outputs of this step are presented in Figure 21.

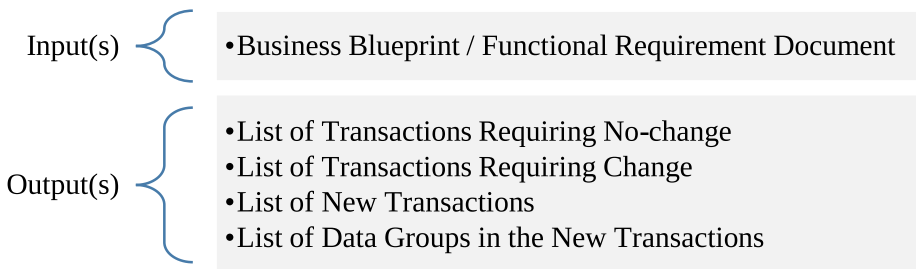


Figure 21. Identifying Required Changes and New Transactions

5.1.4. Step 4 - Calculating Size for Business Processes

For each business process, the size for each category is calculated as DTP by counting data groups in the transactions of that business process. Calculation formula differs based on the logic of counting data groups. Two different versions of this calculation formula are represented in Figure 24 and Figure 25 in the next section.

5.1.5. Step 5 - Calculating Total Size

Total size of an EA project is the total size of the business processes included in that project. Using the formula shown in Figure 22, total size of the project should be calculated in this step.

Project / Module - Total Size Measurement

$$Size_{unchanged} = \sum_{i=1}^{n} Size_{unchanged}\ (BP_i)$$

$$Size_{changed} = \sum_{i=1}^{n} Size_{changed}\ (BP_i)$$

$$Size_{new} = \sum_{i=1}^{n} Size_{new}\ (BP_i)$$

BP = Business Process

Figure 22. Total Size Measurement Formula

5.2. Different Versions of the Method

For size measurement of EA projects, we proposed three different size measurement formula namely as "Quick Size Measurement", "Size Measurement Version 1" and "Size Measurement Version 2".

In Quick Size Measurement, the number of transactions in each category are counted without considering the number of data groups in those transactions. This Quick Size Measurement Formula is presented in Figure 23.

Quick Size Measurement

$$Size_{unchanged} = \sum_{i=1}^{n} UT_i$$

$$Size_{changed} = \sum_{i=1}^{n} CT_i$$

$$Size_{new} = \sum_{i=1}^{n} NT_i$$

UT_i = the number of unchanged transactions in the business process i

CT_i = the number of changed transactions in the business process i

NT_i = the number of new transactions in the business process i

Figure 23. Quick Size Measurement Formula

For size measurement using both transactions and data groups in those transactions, data groups for category “changed” can be counted in two different ways:

1. Counting all data groups in changed transactions

2. Counting only data groups affected by the change in the transactions

We developed two versions of the Size Measurement Method based on this distinction. In Size Measurement Version 1, all data groups are counted for the category “changed” as represented in Figure 24.

Business Process – Size Measurement Version 1

$$Size_{unchanged} = \sum_{i=1}^{N} u_i \cdot DG_i$$

$$Size_{changed} = \sum_{i=1}^{N} c_i \cdot DG_i$$

$$Size_{new} = \sum_{i=1}^{N} n_i \cdot DG_i$$

$$\boldsymbol{u_i} = \begin{cases} \mathbf{1}, & if\ transaction\ i\ is\ an\ unchanged\ transaction \\ \mathbf{0}, & otherwise \end{cases}$$

$$\boldsymbol{c_i} = \begin{cases} \mathbf{1}, & if\ transaction\ i\ is\ a\ changed\ transaction \\ \mathbf{0}, & otherwise \end{cases}$$

$$\boldsymbol{n_i} = \begin{cases} \mathbf{1}, & if\ transaction\ i\ is\ a\ new\ transaction \\ \mathbf{0}, & otherwise \end{cases}$$

$\boldsymbol{DG_i}$ *=total number of data groups in transaction i*

Figure 24. Size Measurement Formula for Business Process - Version 1

In Size Measurement Version 2, only data groups affected by the change of the transaction are counted in the formula of the category “changed”. This size measurement formula is represented in Figure 25.

Business Process – Size Measurement Version 2

$$Size_{unchanged} = \sum_{i=1}^{N} u_i \cdot DG_i$$

$$Size_{changed} = \sum_{i=1}^{N} c_i \cdot DGA_i$$

$$Size_{new} = \sum_{i=1}^{N} n_i \cdot DG_i$$

$$\boldsymbol{u_i} = \begin{cases} \mathbf{1}, \ if \ transaction \ i \ is \ an \ unchanged \ transaction \\ \mathbf{0}, \ otherwise \end{cases}$$

$$\boldsymbol{c_i} = \begin{cases} \mathbf{1}, \ if \ transaction \ i \ is \ a \ changed \ transaction \\ \mathbf{0}, \ otherwise \end{cases}$$

$$\boldsymbol{n_i} = \begin{cases} \mathbf{1}, \ if \ transaction \ i \ is \ a \ new \ transaction \\ \mathbf{0}, \ otherwise \end{cases}$$

DGi *= total number of data groups in transaction i*

DGAi *= total number of affected data groups in transaction i*

Figure 25. Size Measurement Formula for Business Process - Version 2

5.3. Sample Size Measurement

For a better understanding of the method, we applied the method on a well-known business process, the “Purchase Order” process. We used “Size Measurement Method Version 1” with an SAP module for this practice.

A “Purchase Order (PO)” is an official document showing the description, quantity, price, and purchase conditions of the ordered products or services. It is created by the buyer and forwarded to the vendor to officially start the purchasing process. Purchase order process is included in the MM (Material Management) Module of SAP. Main transactions run for this process are presented in Figure 28 in Appendix B.

In this process, the PO is created, changed, displayed, reported, and in most of the cases an approval (release) step is applied. A sample view for the PO is presented in Figure 29 in Appendix B.

5.3.2. Transactions and Data Groups

We use the Project Scope Document to understand that a process would be implemented within the scope of that project. A sample section from the Scope Document showing that the PO process is implemented in the project could be as in Figure 26.

> *The purchasing process will be carried out through the system for both investment and non-investment purchases. MRP will be carried out through the system, minimum stock quantities for materials will be defined, and Purchase Requests will be opened automatically according to the needs. Purchase Orders will be created for purchase requisitions, and the approval process for both purchase requisitions and purchase orders will be carried out through the system. The printout of the purchase orders can be taken over the system or, if desired, sent to the seller by e-mail.*

Figure 26. The Scope Definition for Purchase Order Process

In order to list "Transactions" and "Data Groups" included in the business process, we use the "Business Process Repository (BPR)" of the EA and EA itself as the resource. As described in the method, we consider both master data and transactional data when determining Data Groups.

We firstly checked the master data list for the MM module defined in BPR of SAP, as presented in Table 46 in Appendix B, and determined master data of PO process as "Buyer", "Vendor", "Material", "Conditions", "Supplement" and "Release Strategy". "Purchase Order" is the main transactional data of this process. As described in the scope, PO is created depending on the Purchase Requisition, so we determined "Purchase Requisition" as another transactional data.

We listed related "transactions" and "data groups" of PO process as presented in Table 14.

Table 14. List of Data Groups and Transactions for Purchase Order Process

Business Process	***Process Step***	***Data Groups***	***Transactions***
Purchase Order Processing	Maintain Purchase Order	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	Create, Change, Maintain Supplement, Mass Maintenance, Display
Purchase Order Processing	Release Purchase Order	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	Individual Release, Collective Release

Business Process	***Process Step***	***Data Groups***	***Transactions***
Purchase Order Processing	Report Purchase Order	Purchase Order, Buyer, Vendor, Material, Conditions	Report By Vendor, Report By Material, Report General, Report By Purchase Order

5.3.3. Required Changes and New Transactions

After listing “transactions” and “data groups”; change required transactions, new transaction needs and related data groups should be figured out. For this purpose, we use the “Business Blueprint Document” as the resource. The “Business Blueprint Document” includes both configuration and customization details of the project and new transaction requirements are also defined in this document.

For this sample calculation, we listed “customer requirements”, “related transactions”, “new transactions and their data groups” for the PO process in Table 15.

Table 15. List of the Change Requests for Purchase Order Process

Purchase Order Processing – Required Changes & New Transactions	
CR.1	*“The purchase order will be only created by entering a purchase requisition as a reference document. If the entered purchase requisition’s approvals are not complete, the system will not allow the purchase order creation, and an error message will return stating that approvals are not complete”*
	change in purchase order creation *-transactions (create)*
CR.2	*“When a user tries to change a PO, the purchase order, the system will check the PO approval status. If any approval is given for the PO, a pop-up will appear saying "PO is approved, if you save changes, SAS approvals will be initialized". If the user approves, the change will be saved and approvals will be initialized”*
	change in purchase order change *-transactions (change, mass maintenance)*
CR.3	*“When an approval for the purchase order is given, the next approver will be notified by e-mail”*
	new transaction for PO release, same data groups as individual & mass release *-transactions (e-mail)*

Purchase Order Processing – Required Changes & New Transactions	
CR.4	*"A new PO good receipt report is required for PO showing PO details, good receipt document, batch numbers"*
	new transaction for PO good receipt report *-transactions (report PO good receipt)* *-data groups (purchase order, buyer, vendor, material, conditions, good receipt document, batch)*

5.3.4. Calculating Size

For the business process, we calculated the size of each category by using the size calculation formula presented Figure 24. Calculation sheet is presented in Table 16.

Table 16. Calculation for Purchase Order Process

Transaction	***Data Groups***	***#Data Groups***	u_i	c_i	n_i	***Size** unchanged **(DTP)***	***Size** changed **(DTP)***	***Size** new **(DTP)***
Create	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	0	1	0	0	7	0
Change	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	0	1	0	0	7	0
Maintain Supplement	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	1	0	0	7	0	0
Mass Maintenance	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	0	1	0	0	7	0
Display	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	1	0	0	7	0	0

Transaction	***Data Groups***	***#Data Groups***	u_i	c_i	n_i	***Size** unchanged **(DTP)***	***Size** changed **(DTP)***	***Size** new **(DTP)***
Individual Release	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	6	1	0	0	6	0	0
Collective Release	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	6	1	0	0	6	0	0
E-mail	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	6	0	0	1	0	0	6
Report by Vendor	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0
Report by Material	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0
Report General	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0
Report by Purchase Order	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0
Report PO GR	Purchase Order, Buyer, Vendor, Material, Conditions, Good Receipt Document, Batch	7	0	0	1	0	0	7
						46	**21**	**13**

Based on these calculations, size of the PO process is as follows:

$\text{Size}_{\text{unchanged}}$ = 46 DTP

$\text{Size}_{\text{changed}}$ = 21 DTP

Size_{new} = 13 DTP

CHAPTER 6

APPLICATION OF THE METHOD

This chapter presents two case studies we performed to evaluate our size measurement method. In Section 6.1, the case study analyzing different versions of our size measurement method is presented. Section 6.2 presents the multiple case study that was performed to evaluate our size measurement method for EA projects.

6.1. Case Study 3 – Application of Different Versions

To explore the applicability of our size measurement method in a real-life setting and evaluate different versions of the method, we conducted a case study with an SAP Implementation project.

6.1.1. Case Study Design

We conducted this case study to see our size measurement method's applicability in a real-life setting and to figure out potential improvement opportunities. We determined business processes, transactions, data groups included in the project, defined required changes and new transactions, and measured the size of each module of the project based on three different versions of our size measurement method.

This case study was designed to answer the following questions:

- How can the size measurement method we developed be used for measuring the size of Enterprise Applications?

- Which version of the size measurement method we developed is better for effort estimation of Enterprise Applications?

6.1.2. Description of the Case

The case project is an SAP Implementation Project. Company implementing SAP is a textile & chemistry company located in Turkey.

This project consisted implementation of 9 SAP modules that are listed in Table 17.

Table 17. Case Study 3 - SAP Modules Implemented

SAP Module Implemented	Module Description
FI	Financial Accounting
CO	Controlling
MM	Materials Management
QM	Quality Management
WM	Warehouse Management
SD	Sales and Distribution
PP	Production Planning
PM	Plant Maintenance
HCM	Human Capital Management

SAP Consultancy Company performing the project is the Turkey branch of one of the leading IT companies in Germany. SAP Consultancy Company performed SAP Implementation Projects based on “Accelerated SAP (ASAP) methodology”. Relying on ASAP, they prepare “Business Blueprint Documents” in the requirement elicitation phase of the project. Business Blueprint Document for the project was taken from the company for this case study.

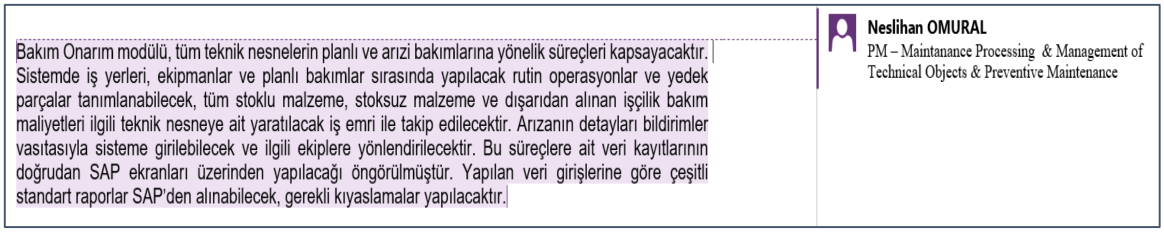
SAP Consultancy Company uses SAP CATS Time Sheet Module for recording working times and tasks. Company provided us “actual effort values” of each SAP module, separately for configuration and programming. Total actual effort of this project is 8035 person-hours.

6.1.3. Application of the Method

We applied three versions of our size measurement method for this project.

- *Step 1 - Determining Business Processes*

Firstly, by reading “Scope” document of the project, we figured out which business processes were applied in that module. A sample view of the PM Module for this step is shown in Figure 27.



Bakım Onarım modülü, tüm teknik nesnelerin planlı ve arızi bakımlarına yönelik süreçleri kapsayacaktır. Sistemde iş yerleri, ekipmanlar ve planlı bakımlar sırasında yapılacak rutin operasyonlar ve yedek parçalar tanımlanabilecek, tüm stoklu malzeme, stoksuz malzeme ve dışarıdan alınan işçilik bakım maliyetleri ilgili teknik nesneye ait yaratılacak iş emri ile takip edilecektir. Arızanın detayları bildirimler vasıtasıyla sisteme girilebilecek ve ilgili ekiplere yönlendirilecektir. Bu süreçlere ait veri kayıtlarının doğrudan SAP ekranları üzerinden yapılacağı öngörülmüştür. Yapılan veri girişlerine göre çeşitli standart raporlar SAP'den alınabilecek, gerekli kıyaslamalar yapılacaktır.

Neslihan OMURAL
PM – Maintanance Processing & Management of Technical Objects & Preventive Maintenance

Figure 27. Case Study 3 - A Sample View for Determining Business Processes

- *Step 2 - Defining Transactions and Data Groups*

By using the Business Process Repository of SAP and our "Scope" comments, we listed transactions and data groups included in that module of this project. As a sample view, the list of transactions and data groups for "Internal Warehouse" business process of the WM module of the project is presented in Table 18.

Table 18. Case Study 3 - A Sample View for Defining Transactions and Data Groups

Business Process	***Process Step***	***Data Groups***	***Transactions***
Maintenance Processing	Notification	Notification, Equipment, Material, Serial Number, Work Center, Responsible	Create, Change, Display, List editing
Maintenance Processing	Order	Equipment, Material, Serial Number, Order, Work Center, Operation, Personnel	Create, Change, Display, Print
Maintenance Processing	Order List	Equipment, Material, Serial Number, Order	Change, Display
Maintenance Processing	Operation List	Equipment, Material, Serial Number, Order, Operation	Change, Display
Maintenance Processing	Component List	Equipment, Material, Serial Number, Order, Component	Change, Display
Maintenance Processing	Permit List	Equipment, Material, Serial Number, Order, Permit	Change, Display
Maintenance Processing	Budget	Equipment, Material ,Order, Budget	Change, Display
Maintenance Processing	Supplement	Equipment, Material, Order, Supplement	Change, Display
Maintenance Processing	Return	Equipment, Material, Order, Return	Change, Display
Maintenance Processing	Availability Control	Equipment, Material, Order, Availability, Component	Activate, Deactivate
Maintenance Processing	Year-End Closing	Equipment, Material, Order, Budget	Carry budget

- *Step 3 - Identifying Required Changes and New Transactions*

We analyzed the "Business Blueprint Document" of modules, we defined the transactions that need to be customized and we determined the new transaction requirements and related data groups. As a sample view, the list of the related change requests for the WM module of the project is presented in Table 19.

Table 19. Case Study 3 - A Sample List of Change Requests

PM Module – Required Changes & New Transactions	
CR.1	change in maintenance order *-transactions (create, change)*
CR.2	change in maintenance notifications *-transactions (create, change)*
CR.3	new transaction for maintenance notifications *-transactions (e-mail)* *- data groups (notification, equipment, material, serial number, work center, responsible)*
CR.4	new transaction for maintenance order component *-transactions (post)* *- data groups (equipment, material, serial number, order, component)*
CR.5	change in maintenance notifications *-transactions (create)*
CR.6	change in maintenance order *-transactions (print)*

- *Step 4 - Calculating Size for Business Processes*

For each business process, we calculated #transactions and #data groups of each category; and calculated the size as DTP. A sample view of the maintenance processing business process for this step is shown in Table 20.

Table 20. Case Study 3 - A Sample Size Calculation for Maintenance Processing

Process Step	***Transaction***	***#Data Groups***	***#Data Groups Affected***	u_i	c_i	n_i	***Size** unchanged (DTP)*	***Size** changed (DTP) – Ver1*	***Size** changed (DTP) – Ver2*	***Size** new (DTP)*
Notification	Create	6	3	0	1	0	0	6	3	0
Notification	Change	6	3	0	1	0	0	6	3	0
Notification	Display	6	0	1	0	0	6	0	0	0
Notification	List editing	6	0	1	0	0	6	0	0	0
Notification	E-mail	6	0	0	0	1	0	0	0	6
Order	Create	7	5	0	1	0	0	7	5	0
Order	Change	7	5	0	1	0	0	7	5	0
Order	Display	7	0	1	0	0	7	0	0	0
Order	Print	7	5	0	1	0	0	7	5	0
Order List	Change	4	0	1	0	0	4	0	0	0
Order List	Display	4	0	1	0	0	4	0	0	0
Operation List	Change	5	0	1	0	0	5	0	0	0
Operation List	Display	5	0	1	0	0	5	0	0	0
Component List	Change	5	0	1	0	0	5	0	0	0
Component List	Display	5	0	1	0	0	5	0	0	0
Permit List	Change	5	0	1	0	0	5	0	0	0
Permit List	Display	5	0	1	0	0	5	0	0	0
Budget	Change	4	0	1	0	0	4	0	0	0
Budget	Display	4	0	1	0	0	4	0	0	0
Supplement	Change	4	0	1	0	0	4	0	0	0
Supplement	Display	4	0	1	0	0	4	0	0	0
Return	Change	4	0	1	0	0	4	0	0	0
Return	Display	4	0	1	0	0	4	0	0	0
Availability Control	Activate	5	0	1	0	0	5	0	0	0
Availability Control	Deactivate	5	0	1	0	0	5	0	0	0
Year-End Closing	Carry budget	4	0	1	0	0	4	0	0	0

Process Step	***Transaction***	***#Data Groups***	***#Data Groups Affected***	u_i	c_i	n_i	***Size unchanged (DTP)***	***Size changed (DTP) – Ver1***	***Size changed (DTP) – Ver2***	***Size new (DTP)***
						Business Process Total Size (DTP)	**95**	**33**	**21**	**6**

➢ *Step 5 - Calculating Total Size*

After figuring out #transactions and #datagroups for each business process, we calculated the size of these 9 modules based on three versions of the method. Size measurement of all modules of this project is shown in Table 1Table 21.

Table 21. Case Study 3 - Size Measurement Results

Module	***Quick Size***			***Size (DTP)-Version 1***			***Size (DTP)-Version 2***			***Realized Effort (person-hours)***		
	Size unchanged	*Size changed*	*Size new*	*Size unchanged*	*Size changed*	*Size new*	*Size unchanged*	*Size changed*	*Size new*	*Configuration*	*Programming*	*Total*
FI	280	15	28	965	68	153	965	25	153	929	296	1225
CO	282	4	0	1259	21	0	1259	11	0	1432	0	1432
MM	242	35	49	850	194	248	850	105	248	1120	206	1326
WM	37	2	16	191	12	117	191	4	117	276	141	417
QM	211	6	9	350	15	43	238	6	43	448	39	487
PP	222	18	21	961	51	118	961	30	118	882	616	1498
PM	68	5	2	279	33	11	279	21	11	345	73	418
SD	121	20	49	537	96	228	414	35	228	848	329	1177
HCM	76	6	0	388	45	0	388	18	0	55	0	55
											Total	**8035**

6.1.4. Estimation Accuracy Evaluation

In order to evaluate the effort estimation accuracy of different versions of our size measurement method, we performed simple and multiple linear regression analysis, and leave one out cross validation for the project.

➢ *Regression Analysis*

We performed regression analysis to analyze the effectiveness of the size measurement method in estimating effort.

First, we performed regression analysis for the case that the dependent variable is "total actual effort" and the independent variable is "total size". We performed this analysis for all three versions of the size measurement method.

We then performed multiple linear regression analysis for three versions of the method with the following variables:

Independent variables:

- $Size_{unchanged}$
- $Size_{change}$
- $Size_{new}$

Dependent variable:

- $Effort_{actual}$

Regression analysis results for three versions of the size measurement method are presented in Table 22.

Table 22. Case Study 3 - Regression Analysis Results

		Quick Size Measurement	*Size Measurement Ver1*	*Size Measurement Ver2*
Simple Linear Regression	*Intercept*	41.08	-51.18	37.90
	Size $_{total}$	4.20	1.17	1.14
	R-squared	0.7007	0.8674	0.8253
	Significance F	0.0048	0.0002	0.0006
Multiple Linear Regression	*Intercept*	15.98	-89.80	-31.97
	Size $_{unchanged}$	3.98	1.22	1.13
	Size $_{changed}$	-3.73	-2.33	-3.41
	Size $_{new}$	12.57	3.30	3.16
	R-squared	0.7350	0.9256	0.8908
	Significance F	0.0662	0.0030	0.0077

Regression results for a regression model are evaluated by both assessing "R-squared (coefficient of determination)" and "Significance F" values. For justification of the regression model, "R-squared (coefficient of determination)" is used. "R-squared" shows how well the model fits the data. As the "R-squared" value increases, the model fits the data much better. If "R-squared" is greater than 0.90, it is examined as a predictive relationship; if "R-squared" is between 0.70 and 0.90, it is examined as a strong relationship; if "R-squared" is between 0.50 and 0.70, it is examined as an adequate relationship and if "R-squared" is less than 0.50, it is examined that the model is not reliable [78]. "Significance F" value shows the probability of the regression model is wrong. As the "Significance F" value decreases, the model is much more significant. The significance level of less than or equal to 0.05 is considered acceptable for tests [79].

Based on both simple and multiple linear regression, "Size Measurement Version 1" shows the most predictive relationship and significance level is in an acceptable range. When we analyze the regression line of all versions, HCM (Human Capital Management) module seems significantly falling outside the regression line, as an outlier. We think, this could be occurred due to a data entry error. To be sure about our conclusions of "Size Measurement Version 1", we decided to renew the regression analysis after removing the outlier.

Renewed regression analysis results for three versions of the size measurement method are presented in Table 23.

Table 23. Case Study 3 - Regression Analysis Results Renewed

		Quick Size Measurement	*Size Measurement Ver1*	*Size Measurement Ver2*
Simple Linear Regression	*Intercept*	209.32	106.71	194.06
	Size $_{total}$	3.61	1.05	1.02
	R-squared	0.6450	0.9189	0.8954
	Significance F	0.0164	0.0001	0.0004
Multiple Linear Regression	*Intercept*	200.01	56.79	126.13
	Size $_{unchanged}$	3.19	1.11	1.04
	Size $_{changed}$	8.22	-0.70	-1.43
	Size $_{new}$	4.84	2.05	2.15
	R-squared	0.6685	0.9454	0.9189
	Significance F	0.1815	0.0055	0.0120

Based on both renewed simple and multiple linear regression, "Size Measurement Version 1" again shows the most predictive relationship and the significance level is again in an acceptable range.

- *Leave One Out Cross Validation*

In order to understand how accurately different versions of the size measurement method predicts effort; we performed the leave one out cross validation (LOOCV) method for this case study project.

We run 8 tests based on the size measurement results of three different versions. Results of these tests are shown in Table 24, Table 25, and Table 26.

Table 24. Case Study 3 - LOOCV Results of Quick Size Measurement

Iteration	**Module**	**Size** unchanged	**Size** changed	**Size** new	**Effort** predicted (person-hour)	**Effort** actual (person-hour)	**MRE**
Iteration 1	FI	280	15	28	1447	1225	0.18
Iteration 2	CO	282	4	0	814	1432	0.43
Iteration 3	MM	242	35	49	1986	1326	0.50
Iteration 4	WM	37	2	16	406	417	0.03
Iteration 5	QM	211	6	9	1110	487	1.28
Iteration 6	PP	222	18	21	1042	1498	0.30
Iteration 7	PM	68	5	2	567	418	0.36
Iteration 8	SD	121	20	49	664	1177	0.44

Table 25. Case Study 3 - LOOCV Results of Size Measurement Version 1

Iteration	**Module**	**Size** unchanged (DTP)	**Size** changed (DTP)	**Size** new (DTP)	**Effort** predicted (person-hour)	**Effort** actual (person-hour)	**MRE**
Iteration 1	FI	965	68	153	1466	1225	0.20
Iteration 2	CO	1259	21	0	1461	1432	0.02
Iteration 3	MM	850	194	248	1691	1326	0.27
Iteration 4	WM	191	12	117	586	417	0.40
Iteration 5	QM	350	15	43	539	487	0.11
Iteration 6	PP	961	51	118	1269	1498	0.15
Iteration 7	PM	279	33	11	273	418	0.35
Iteration 8	SD	537	96	228	965	1177	0.18

Table 26. Case Study 3 - LOOCV Results of Size Measurement Version 2

Iteration	**Module**	**Size** unchanged (DTP)	**Size** changed (DTP)	**Size** new (DTP)	**Effort** predicted (person-hour)	**Effort** actual (person-hour)	**MRE**
Iteration 1	FI	965	25	153	1528	1225	0.25
Iteration 2	CO	1259	11	0	1375	1432	0.04
Iteration 3	MM	850	105	248	2102	1326	0.59
Iteration 4	WM	191	4	117	668	417	0.60
Iteration 5	QM	238	6	43	440	487	0.10
Iteration 6	PP	961	30	118	1284	1498	0.14
Iteration 7	PM	279	21	11	395	418	0.05
Iteration 8	SD	414	35	228	820	1177	0.30

Using the equations listed in Figure 8, we calculated MMRE, MdMRE, and PRED(30) values as shown in Table 27.

Table 27. Case Study 3 - Estimation Accuracy Evaluation Results

Project	*MMRE*	*MdMRE*	*PRED (30)*
Quick Size Measurement	*0.44*	*0.39*	*0.38*
Size Measurement Ver1	*0.21*	*0.19*	*0.75*
Size Measurement Ver2	*0.26*	*0.20*	*0.75*

MMRE, MdMRE, and PRED (30) values all show that "Size Measurement Version 1" performs better than the other two versions for effort estimation.

6.1.7. Discussions of the Results

The main contribution of our size measurement method is evaluating business requirements in different categories as "unchanged", "changed" and "new" and measuring size based upon these categories. As a unit of size measurement, we counted data groups processed by transactions and developed two different versions of the method based on the counting logic of data groups in the category "changed". While in "Size Measurement Method Version 1" we count all the data groups handled by the transaction, in "Size Measurement Method Version 2" we only count the data groups affected by the change. We also proposed another version as "Quick Size Measurement Method", in which we do not take into account data groups and count only transactions based upon categories.

The first objective of the case study was to evaluate the applicability of different versions of our size measurement method for EA projects. We measured the size of the case project by using "Scope" and "Business Blueprint" documents. Using these documents, we could categorize business requirements and distribute transaction codes and data groups into these categories as defined in the method. Requirements in the blueprint document were clear enough to make this categorization. After distributing transactions and data groups, it was clear how to calculate the total size of business processes and modules in DTP size units.

The second objective of the case study was to determine which version of the size measurement method is better for the effort estimation of EA projects. When we examined the results, we concluded that "Size Measurement Version 1" is the best version of the method for effort estimation of EA projects. This was not an unexpected result for us, since when we count only data groups affected by the change, we would miss the real effect of the change on size. To better explain the situation, we can think through an example. Suppose there are two transactions that need to be changed according to business requirements, one handling 3 data groups, other one handling10 data groups, and both having 2 data groups affected by the change. In Version 2, both transactions will have the same size, while in Version 1 we will measure the first transaction as 3 DTP and the second transaction as 10 DTP. When we consider the customization to be made for the change, making changes in a transaction with 10 data groups and making changes in a transaction with 3 data groups will not be the same task. Considering that the transaction processing more data groups will be more complex, the work required for customization will also be greater. According to this inference, when we count all data groups, we can measure the effect of the change on the basis of transaction size.

The "Quick Size Measurement" version had worse estimation accuracy than the other versions, but it still doesn't mean that this version won't be used at all. Since we do not count data groups here, the measurement time is shorter, so it can still be used for early estimation. When we compare the "Quick Size Measurement" with the versions that we count data groups, we see that the versions we count data groups give better results in all cases. This shows that the idea of counting data groups in addition to transactions is much more convenient for size measurement.

6.2. Case Study 4 – Multiple Case Study

In order to observe the validity of our size measurement method and increase the generalizability, we repeated Case Study 3 with new projects. We conducted a multiple case study with two different SAP Implementation projects.

6.2.1. Case Study Design

We conducted this case study to observe the size measurement method a in real-life setting and to figure out potential improvement opportunities. We determined business processes, transactions, and data groups included in the projects, defined required changes and new transactions, and measured size for each module of the project based on the Size Measurement Version1 Method.

This case study was designed to answer the following research questions:

RQ1. How can the size measurement method be used for measuring the size of Enterprise Applications?

RQ2. How accurate does the size measurement method perform for effort estimation of Enterprise Applications?

We determined our case selection strategy to select cases from different domains, sizes, and organizations to enable us to evaluate the applicability of the method in different situations. In addition, the accessibility and integrity of the project documents and the availability of the realized effort data were among our case study selection criteria. Relying on our project management experience and network on SAP, we contacted SAP Consultancy Companies, explained our study and described the required project documents for the study. Among the project documents, we chose the two most comprehensive projects with complete effort data, including different types of modules that were implemented in different organizations and domains.

The dataset for the case studies includes the main project documents regarding requirements elicitation and realized effort data of each module. Regarding RQ1, we planned to measure the size of all modules of the projects using the proposed size measurement method. Our validation strategy to answer RQ2 was to use multiple linear regression analysis and LOOCV. To evaluate the accuracy of effort estimations, we planned to calculate MMRE, MdMRE and PRED(n) metrics using LOOCV results and compare them with threshold values accepted in the software engineering discipline.

During the design phase of the study, we identified the main threats to internal and external validity of the study, and planned mitigation actions for them. The main threat to internal validity was the quality and reliability of documents and effort data obtained from organizations. To mitigate this threat, we chose the projects where effort data was properly collected on the basis of modules, and project documents were clear and complete. Another factor that increased the reliability of the effort data was that this data had been confirmed with the customer. In these EA projects, the effort data was shared with the customer, and customer approval was obtained for billing.

Another threat to internal validity is the interpretation of the collected project requirements documents. Business processes, related changes and new transaction needs are extracted from these Scope and Business Blueprint documents. In order to reduce interpretation variance, project managers were contacted for issues that were complex or unclear in these documents.

The main threat to validity was the representation of the cases for EA projects. To mitigate this threat, we chose EA projects, including different types of modules, and implemented in different business domains. Another validity threat is the validity of the measures. Variances may occur in the measurement depending on the measurer. To reduce this variance and ensure reliability, the size measurement method was clearly defined in steps, and measuring size of the projects was explained through a sample business process. Validation procedures were also explained in detail. In this way, other researchers who want to repeat this research can reach similar results using these procedures.

6.2.2. Description of the Cases

The first case project is an SAP Implementation Project. Company implementing SAP is an oil & energy company located in Turkey. This project consisted implementation of 10 SAP modules that are listed in

Table 28.

Table 28. Case Study 4 - SAP Modules Implemented in the 1st Project

SAP Module Implemented	**Module Description**
FI	Financial Accounting
TRM	Treasury and Risk Management
CO	Controlling
PS	Project System
MM	Materials Management
QM	Quality Management
WM	Warehouse Management
SD	Sales and Distribution
PM	Plant Maintenance
HCM	Human Capital Management

The second case project is also an SAP Implementation Project. Company implementing SAP is a manufacture company located in Turkey. This project consisted implementation of 9 SAP modules that are listed in Table 29.

Table 29. Case Study 4 - SAP Modules Implemented in the 2nd Project

SAP Module Implemented	**Module Description**
CO	Controlling
PS	Project System
FI	Financial Accounting
MM	Materials Management
WM	Warehouse Management
PP	Production Planning
QM	Quality Management
SD	Sales and Distribution
HCM	Human Capital Management

These two projects were both carried out by the same SAP Consultancy Company. This Consultancy Company is one of the gold partners of SAP AG and one of the leading IT companies in Europe with over 2000 employees. These projects were performed by the Turkey branch of the company.

For both projects, “Accelerated SAP (ASAP) methodology” were used as the project implementation methodology. “Business Blueprint Documents” were prepared in line with

ASAP during the requirements elicitation phase of the project. Within the scope of the case study, “Scope” and “Business Blueprint” documents were provided for both projects.

The Consultancy Company uses SAP CATS Time Sheet to capture the projects’ efforts. “Realized effort” values are gathered for each module of the projects. The total effort of the first project was reported as 9232.5 person-hours, and the total effort of the second project as 3303 person-hours.

6.2.3. Application of the Method

We applied all steps of the size measurement method for both projects.

- *Step 1 - Determining Business Processes*

In the first step, we read the “Scope” document of the projects and figured out which *business scenarios* and related *business processes* were applied in each module. As a sample view, business processes’ list of the QM module of the 1st project is presented in Table 30.

Table 30. Case Study 4 - A Sample Business Processes List

QM Module - Business Processes
Quality planning
Quality inspection
Quality control
Test equipment management

- *Step 2 - Defining Transactions and Data Groups*

In the second step, we listed *the transactions* and *the data groups* in each module of the project by using the SAP Business Process Repository resource. As a sample view, the list of transactions and data groups for “Quality Inspection” business process of QM module of the 1st project is presented in Table 31.

Table 31. Case Study 4 - A Sample List of Data Groups and Transactions

Business Process	***Process Step***	***Data Groups***	***Transactions***
Quality Inspection	Source Inspection Lot	Inspection Lot, Material	Trigger, Plan, Overview
Quality Inspection	Deadline Monitoring Lot	Inspection Lot, Material	Trigger, Plan, Overview

Business Process	***Process Step***	***Data Groups***	***Transactions***
Quality Inspection	Inspection	Inspection, Material	Change, Display, Record
Quality Inspection	Maintain Quality Control Card	Quality Control Card, Material	Create, Change, Display, Print
Quality Inspection	Usage Decision	Usage Decision, Inspection, Material	Trigger, Plan, Overview

- *Step 3 - Identifying Required Changes and New Transactions*

In the third step, by reading the "Business Blueprint Document" of the modules, we defined the transactions that need to be customized, and determined the new transaction requirements and related data groups. As a sample view, the list of the related change requests for the QM module of the 1st project is presented in Table 32.

Table 32. Case Study 4 - A Sample List of the Change Requests

QM Module – Required Changes & New Transactions	
CR.1	change in source inspection lot *-transactions (trigger, plan)*
CR.2	change in material inspection lots *-transactions (overview, overview of quantities, quantities)*
CR.3	change in usage decision *-transactions (trigger, plan)*
CR.4	change in customer quality inspection lot *-transactions (overview, overview of quantities, quantities)*
CR.5	new transactions for quality inspection lot cockpit *-transactions (maintain, summary)* *-data groups (material, inspection lot, control type, quality level, project, purchase order ,document, decision)*
CR.6	change in inspection *-transactions (change)*

QM Module – Required Changes & New Transactions	
CR.7	change in vendor inspection lots *-transactions (overview, overview of quantities, quantities)*
CR.8	new transaction for vendor inspection notification *-transactions (send e-mail)* *-data groups (vendor, material, inspection notifications)*

- *Step 4 - Calculating Size for Business Processes*

In this step, we calculated the size of the business processes as DTP for each category: "unchanged", "changed", and "new". The calculation table for the business process "Quality Inspection" of the 1st project's QM Module is presented in Table 33.

Table 33. Case Study 4 - A Sample Calculation for Quality Inspection

Process Step	***Transaction***	***Data Groups***	***#Data Groups***	u_i	c_i	n_i	***Size** unchanged (DTP)*	***Size** changed (DTP)*	***Size** new (DTP)*
Source Inspection Lot	Trigger	Inspection Lot, Material	2	0	1	0	0	2	0
Source Inspection Lot	Plan	Inspection Lot, Material	2	0	1	0	0	2	0
Source Inspection Lot	Overview	Inspection Lot, Material	2	1	0	0	2	0	0
Deadline Monitoring Lot	Trigger	Inspection Lot, Material	2	1	0	0	2	0	0
Deadline Monitoring Lot	Plan	Inspection Lot, Material	2	1	0	0	2	0	0
Deadline Monitoring Lot	Overview	Inspection Lot, Material	2	1	0	0	2	0	0
Inspection	Change	Inspection, Material	2	0	1	0	0	2	0
Inspection	Display	Inspection, Material	2	1	0	0	2	0	0
Inspection	Record	Inspection, Material	2	1	0	0	2	0	0
Maintain Control Card	Create	Quality Control Card, Material	2	1	0	0	2	0	0

Process Step	*Transaction*	*Data Groups*	*#Data Groups*	u_i	c_i	n_i	*Size unchanged (DTP)*	*Size changed (DTP)*	*Size new (DTP)*
Maintain Control Card	Change	Quality Control Card, Material	2	1	0	0	2	0	0
Maintain Control Card	Display	Quality Control Card, Material	2	1	0	0	2	0	0
Maintain Control Card	Print	Quality Control Card, Material	2	1	0	0	2	0	0
Usage Decision	Trigger	Usage Decision, Inspection, Material	3	0	1	0	0	3	0
Usage Decision	Plan	Usage Decision, Inspection, Material	3	0	1	0	0	3	0
Usage Decision	Overview	Usage Decision, Inspection, Material	3	1	0	0	3	0	0
Quality Inspection Lot	Maintain	Material, Inspection Lot, Control Type, Quality Level, Project, Purchase Order ,Document, Decision	8	0	0	1	0	0	8
Quality Inspection Lot	Summary	Material, Inspection Lot, Control Type, Quality Level, Project, Purchase Order ,Document, Decision	8	0	0	1	0	0	8
QM – Quality Inspection Total Size (DTP)							**23**	**12**	**16**

- *Step 5 - Calculating Total Size*

After calculating the size of each business process, we calculated the total size of each module of the projects. Size measurement results for both projects are shown in Table 34 and Table 35.

Table 34. Case Study 4 - Size Measurement Results of the 1st Project

1st Project	***Size (DTP)***			***Effort realized (person-hours)***		
Module	***Size unchanged***	***Size changed***	***Size new***	***Configuration***	***Programming***	***Total***
FI	2094	256	40	1958	444	2402
TRM	654	101	98	1131	42	1173

1st Project	*Size (DTP)*			*Effort realized (person-hours)*		
Module	***Size unchanged***	***Size changed***	***Size new***	***Configuration***	***Programming***	***Total***
CO	276	26	18	293	59	352
PS	156	6	182	343.5	50	393.5
MM	1111	103	84	1315	151	1466
QM	317	36	19	436	25	461
WM	188	15	18	288	7	295
SD	463	37	14	571	34	605
PM	284	108	123	780	251	1031
HCM	648	118	141	937	117	1054
					Total Effort realized (person-hours)	**9232.5**

Table 35. Case Study 4 - Size Measurement Results of the 2nd Project

2nd Project	***Size (DTP)***			***Effort realized (person-hours)***		
Module	***Size unchanged***	***Size changed***	***Size new***	***Configuration***	***Programming***	***Total***
CO	476	94	45	378	1	379
PS	128	15	7	165	4	169
FI	760	106	80	545	17	562
MM	487	52	74	562	16	578
WM	185	18	6	161	1	162
PP	510	33	32	397	12	409
QM	238	15	26	261	2	263
SD	324	50	37	371	22	393
HCM	499	124	25	388	0	388
					Total Effort realized (person-hours)	**3303**

6.2.4. Estimation Accuracy Evaluation

In order to evaluate the effort estimation accuracy of our size measurement method, we performed multiple linear regression analysis and leave one out cross validation for both projects.

- *Linear Regression Analysis*

We performed multiple linear regression analysis with the following variables:

Independent variables:

- $Size_{unchanged}$
- $Size_{changed}$
- $Size_{new}$

Dependent variable:

- $Effort_{realized}$

For the justification of the linear regression model for the purpose of prediction, we first assessed residual plots of the three independent variables as shown in Figure 30, Figure 31, Figure 32, Figure 33, Figure 34, and Figure 35 in Appendix C. Randomly dispersed points in a residual plot shows linear regression is appropriate for the data; otherwise a non-linear model is said to be more appropriate [47]. As it can be seen from the figures in Appendix C, no particular pattern seems to exist and the variables are randomly scattered above and below the Residual=0 line. Therefore, linear regression could be used for both case study projects.

Results of the multiple linear regression analysis for the projects are shown in Table 36.

Table 36. Case Study 4 - Regression Analysis Results

		1st Project	*2nd Project*
Multiple Linear Regression	*#Observations*	10	9
	Intercept	131.5	127.5
	$Size_{unchanged}$	0.53	0.23
	$Size_{changed}$	4.60	0.03
	$Size_{new}$	1.20	3.87
	R-squared	0.977	0.922
	Significance F	0.00002	0.00339

R-squared (coefficient of determination) is used to evaluate the strength of the relationship between the regression model and the dependent variable. Considering the defined criteria for the coefficient of determination in software engineering for planning purposes [45], having "R-squared" value greater than 0.90 for both projects shows us that there is a predictive relationship between size and effort. After assessing the coefficient of determination, we should determine if the result is significant; significance shows the probability of finding the correlation by chance [36]. With significance level of less than 0.5 for both projects, models are in an acceptable significance range.

- *Leave One Out Cross Validation*

In order to understand how accurately our size measurement method predicts effort; we performed LOOCV method for both projects. As explained in detail in Chapter 2, we preferred LOOCV since we had a limited number of measurement data, and we wanted to ensure consistency and repeatability for the estimation. In this method, a single observation from the data set is used as "test" data and the remaining n-1 observations are used as "training" data. The test data is changed in sequence in each iteration. Iterations are performed as many as the number of the total data set.

We run 10 tests for the first project and 9 tests for the second project, and calculated *MRE* value for each test. Equation to calculate MRE is presented in Figure 8.

The results of these tests are shown in Table 37 and Table 38 .

Table 37. Case Study 4 - LOOCV Results of the 1st Project

Iteration	***Module***	***Size** unchanged (DTP)*	***Size** changed (DTP)*	***Size** new (DTP)*	***Effort** predicted (person-hours)*	***Effort** realized (person-hours)*	***MRE***
1	FI	2094	256	40	2807	2402	0.17
2	TRM	654	101	98	1048	1173	0.11
3	CO	276	26	18	444	352	0.26
4	PS	156	6	182	659	393.5	0.68
5	MM	1111	103	84	1202	1466	0.18
6	QM	317	36	19	499	461	0.08
7	WM	188	15	18	334	295	0.13
8	SD	463	37	14	555	605	0.08
9	PM	284	108	123	749	1031	0.27
10	HCM	648	118	141	1250	1054	0.19

Table 38. Case Study 4 - LOOCV Results of the 2nd Project

Iteration	***Module***	***Size** unchanged (DTP)*	***Size** changed (DTP)*	***Size** new (DTP)*	***Effort** predicted (person-hours)*	***Effort** realized (person-hours)*	***MRE***
1	CO	476	94	45	431	379	0.14
2	PS	128	15	7	194	169	0.15
3	FI	760	106	80	688	562	0.22

Iteration	*Module*	***Size** unchanged* *(DTP)*	***Size** changed* *(DTP)*	***Size** new* *(DTP)*	***Effort** predicted* *(person-hours)*	***Effort** realized* *(person-hours)*	***MRE***
4	MM	487	52	74	467	578	0.19
5	WM	185	18	6	210	162	0.29
6	PP	510	33	32	210	409	0.49
7	QM	238	15	26	292	263	0.11
8	SD	324	50	37	339	393	0.14
9	HCM	499	124	25	239	388	0.38

In order to evaluate the accuracy of the effort estimations, we used MMRE, MdMRE and PRED(30), which are the most used estimation accuracy evaluation metrics in software engineering discipline [49]. Equations to calculate these metrics are presented in Figure 8.

Results of the estimation accuracy calculations are shown in Table 39.

Table 39. Case Study 4 - Estimation Accuracy Evaluation Results

Project	*#Observations*	*MMRE*	*MdMRE*	*PRED (30)*
1st Project	*10*	*0.21*	*0.17*	*0.90*
2nd Project	*9*	*0.24*	*0.19*	*0.78*

Considering the threshold value "0.25" of MMRE and MdMRE for an accurate estimation for software projects [50]; having MMRE and MdMRE less than 0.25 for both projects showed accurate estimations. For PRED (30), having a value more than or equal to 0.60 is defined as good estimation accuracy [51], and even more, having a value more than or equal to 0.70 is defined as a more reliable accuracy threshold [52] [53]. Estimations for both projects, with values greater than this threshold, appear to have met reliable accuracy criteria.

6.2.6. Discussions of the Results

With this multiple case study, we aimed to observe the validity of our size measurement method. The first objective of the case study was to evaluate the applicability of the size measurement method for EA projects. We measured the size of each module (19 modules in total) for the case projects. "Scope" documents of both projects were clear enough that we were able to extract the business scenarios and business processes of the projects. In the "Business Blueprint" documents of both projects, business process definitions were clear that they explained whether it would be used as is, or a change was required. New transactions needs were defined clearly in those documents. We could make the necessary categorization for transactions as defined in the method. We then calculated the total size of business processes and modules.

We discussed the potential threats to the validity of our study and took steps to minimize them in the case study design phase. We defined the main threat to internal validity as the quality and reliability of the project documents. In order to mitigate this threat, we selected projects where effort data was appropriately collected, and project documentation was clear and complete. We defined the other threat to internal validity as the interpretation of the project documents. To mitigate this threat, we contacted the project managers for complex and unclear issues in the documents. For the main threat to validity, the representation of the cases for EA projects, we chose the case study projects that included different modules and were implemented in different business domains. Lastly, for the reliability threat, we defined the method and validation procedures clearly in steps, and explained the method via a sample measurement scenario.

The most challenging and lengthy step in measuring size was to produce lists of transactions and data groups. The Business Process Repository that we used for this purpose is very comprehensive, it includes business scenarios for different industries and organizations. Even if we found the business process in the repository, it was difficult to determine which of the transactions in it were in the scope. We cross-checked the lists in the SAP modules to ensure we didn't make any incomplete or wrong choices. We had to spend 4-7 person-hours of effort for each module, depending on the module size, for creating and checking these lists.

The second objective of the case study was to evaluate estimation accuracy of the size measurement method for EA projects. We used both multiple linear regression analysis and LOOCV for this objective. Results showed that there was a predictive relationship between size and effort for both of the project. For both projects, we calculated MMRE, MdMRE and PRED (30) values. MMRE and MdMRE values were below the threshold values accepted in the software engineering discipline. With having PRED (30) values greater than 0.70, we observed that the quality estimation accuracy was achieved for both projects. With these estimation results, we concluded that the size measurement method is significantly better for EA projects.

CHAPTER 7

CONCLUSION

This thesis presents a novel size measurement method for Enterprise Applications and its application in the field. In this method, transactions are categorized as “unchanged”, “changed”, and “new” according to business requirements analysis and size is measured on the basis of a new size measurement unit, DTP (data transaction point). Section 7.1 summarizes the studies performed throughout the thesis. Section 7.2 presents the contributions accomplished by this study. Limitations of the study and planned future work are described in Section 7.1.

7.1.Summary of the Thesis Study

The main purpose of this thesis is to develop a size measurement method for EA projects. To achieve this purpose, the following research questions were answered within the scope of this thesis:

RQ1. Which size measurement methods have been developed or proposed for EA projects?

RQ1.1. What are the pros and cons of these methods?

RQ1.2. How are these size measurement methods validated, and what is the accuracy?

RQ2. What are the significant characteristics of EA projects that affect size?

RQ3. How can significant characteristics of EA projects be used to formulate a proper size measurement method?

RQ4. How can this size measurement method be used for effort estimations and what is the accuracy?

We implemented design science research methodology to accomplish research goals of the thesis. We explained this methodology in detail in the Chapter 2. This methodology was implemented in three main stages, namely “problem identification”, “solution design” and “evaluation”.

In the problem identification stage, we performed a Systematic Literature Review (SLR) in order to answer research questions RQ1, RQ1.1 and RQ1.2. In this SLR, our aim was to identify existing size measurement / effort estimation methods proposed for EA projects and reveal improvement areas of these methods. We analyzed deeply 41 primary studies in this study. The main implications of the study were the most common research approach applied in these studies was "development of estimation methods", "Function Points" based size measurement methods were the most common used methods, and the "Business Blueprint" document was the main source used in Function Point Analysis. We concluded that "size measurement / effort estimation" is a significant issue for EA projects and there was no consensus on how to measure the size or estimate the effort of such projects.

In the solution design stage, we conducted two different exploratory case studies in order to answer research questions RQ1.1, RQ2 and RQ3. In the first exploratory study, we aimed to analyze existing size measurement methods proposed for EA projects. We determined three main function point-based methods published in the literature, and applied them in an SAP Implementation Project including 7 modules. We evaluated the measurement process and estimation accuracy. The main implication of this study was that business processes are valuable resources for size measurement and effort estimation. We concluded that the critical parameters in size measurement of EA projects are "modification" and "reuse"; these parameters should be measured properly for precise size measurement.

In the second exploratory case study, we aimed to explore the usage of reuse levels for sizing EA projects. We developed an approach to calculate the reuse reflective size of EA projects. We used COSMIC function points as a base size unit, defined reuse levels and measured reuse reflective size using these reuse levels. The "Business Blueprint" document was the main source of this approach. We applied this approach in an SAP Implementation Project to assess the approach in real-life settings. In order to evaluate the effort estimation accuracy of the approach, we compared MRE values for COSMIC size where reuse was taken and not taken into account. Results showed that reuse reflected size measurement could lead to more accurate effort estimations. Although we had promising results, we had some concerns about using reuse levels and constant impact factors. Reuse level selection was critical in this approach and a wrong selection would result in a completely different size. We concluded that instead of selecting "level of reuse", we should develop a size measurement method that precisely measures the amount of "reuse" or "change".

Considering inferences that we obtained from all of these studies, we developed a size measurement method for EA projects. In this method, we proposed measuring "changes" where pre-built functionality is not sufficient for customer requirements. We counted "data groups" and measured "changes" applied to transactions. Claiming that a change in a transaction with one data group would not have the same effect on size as a change in a transaction with multiple data groups, we defined a novel size measurement unit, namely *Data Transaction Point (DTP)*, for EA projects. We measured size as DTP in three categories as "*unchanged* (no change required transactions)", "*changed* (customization required transactions)" and "*new* (new developed transactions)". We defined five main steps to apply this method as presented in Figure 18.

We proposed three different versions of this method. In the “Quick Size Measurement Method” version, we only counted the number of transactions in each category without considering the number of data groups in the transactions. The other two versions basically differentiated based on how we count the data groups for the "changed" category. In the “Size Measurement Method Version 1”, we counted all data groups in the changed transactions, whereas in the “Size Measurement Method Version 2”, we counted only the data groups affected by the change.

In the evaluation stage, we conducted two different case studies to mainly answer research questions RQ3 and RQ4. In the third case study of the thesis, we aimed to analyze different versions of the proposed size measurement method. In the last case study of the thesis, our aim was to validate the size measurement method for EA projects.

In the third case study, we applied three versions of the method in an SAP Implementation Project including 9 modules. We used simple and multiple linear regression analysis, and LOOCV method for assessing effort estimation accuracy. When both simple and multiple linear regression results were evaluated, we concluded that “Size Measurement Method Version 1” had the most predictive relationship for effort estimation. We then performed LOOCV method for three versions and calculated MMRE, MdMRE and PRED (30) metrics. With the results respectively, “0.21”, “0.19” and “0.75”, “Size Measurement Method Version 1” performed better for effort estimation than the other two versions. This result was not surprising to us, as we anticipated that if we counted only the data groups affected by the change, we might miss a part of the size. As defined in Version 1, by counting all data groups, we could measure the impact of the change on the entire transaction. Although the “Quick Size Measurement” version did not give good results compared to the others, we stated that it could still be used for early estimation.

For validation of the method, we conducted a multiple case study with two different SAP Implementation projects, including 10 and 9 modules for each. We used “Size Measurement Method Version 1” as the size measurement method. For assessing the effort estimation accuracy, we performed multiple linear regression analysis and LOOCV. When we analyzed the correlation results of both projects, we inferred that there is a predictive relationship between size and effort. Significance results showed that the linear regression models were within the acceptable range for both projects. We calculated MMRE, MdMRE and PRED (30) metrics for both projects. For the first project, results were “0.21”, “0.17”, “0.90” respectively, and for the second project, results were “0.24”, “0.19”, “0.78” respectively. When we compared the results with defined thresholds for these metrics, all values showed an acceptable and reliable effort estimation accuracy.

In the last two case studies, we experienced using the size measurement method in three different projects, including 28 modules, that were conducted in three different companies. We were able to apply the method as defined; we determined transactions, data groups, changes in the transactions, and new transactions needed for all modules. Effort estimation results were convincing for all projects. As a result of all these studies, we concluded that the size measurement method we developed is significantly better for EA projects.

7.2.Contributions

The main contribution of this study is the proposed size measurement method. This is a novel size measurement method developed by taking into account the unique characteristics of EA projects. As implementations presented in this thesis showed that effort estimations based on this size measurement method have significant accuracy. Obviously with better effort estimations, project schedule and budgets overruns could be decreased.

This size measurement method can be applied by a novel user having general knowledge about the implemented EA. By using this method, size measurement variances and subjectivity will be reduced. As the method is used many times in an organization, productivity values will be more reliable, and subsequent effort estimations will be more accurate.

The second contribution is the specific size measurement unit DTP for EA projects. This novel size measurement unit can be used as a base unit for these types of projects. Since size measurement is performed at business process level in the proposed method, size measurement unit DTP can be used for most of the project management processes such as planning, monitoring, and control.

Another contribution of this thesis is the useful yet simple definition and implementation of the concepts “reuse” and “change” in EA projects. With these definitions, we have revealed how EA projects could be evaluated.

7.3. Limitations and Future Work

The most important limitation of our study is the limited scope of the cases. We conducted case studies with SAP implementation projects. We use SAP projects for two main reasons. Firstly, SAP is one of the biggest EA vendors and includes many different types of Enterprise Applications, such as core ERP modules, HCM, TRM, CRM and SRM. In an SAP implementation project, we have the opportunity to try our method on many EA types. The second reason is our project management experience and network in the SAP field. In this way, we could obtain the required project documents and analyze them. We plan to conduct further case studies to evaluate the applicability and validity of the proposed size measurement method with other EA vendors.

The second major limitation of the case studies is the limited size of the datasets. There were 9-10 implemented modules in each one of the case projects. We were able to evaluate the size measurement method in 28 different EA modules in three different companies. Although each EA module is significantly large, increasing the number of measurements will strengthen the method’s validity.

Another limitation of our study is the adaptability of the model to the industry. Within the scope of this thesis, we did not have a chance to examine with industry experts whether the model would be accepted in the industry and how it could find an application area. Therefore, we plan to conduct further studies to get industry feedback on the model's adaptability. We also plan to conduct case studies where the method is applied by different measurers.

EA project tools as SAP Solution Manager are currently used also for Business Blueprint document generation. Since these project tools are capable of generating Business Blueprint documents, these tools could also be used for size measurement of EA projects. This will shorten total time required for DTP calculation.

Motivated by the case study results, this size measurement method can also be applied in other change-intensive project types such as software maintenance and upgrade projects. We plan to extend our studies to evaluate the method's applicability for these types of projects.

REFERENCES

[1] O. Volkoff, D. M. Strong, and M. B. Elmes, "Understanding enterprise systems-enabled integration," *Eur. J. Inf. Syst.*, vol. 14, no. 2, pp. 110–120, 2005.

[2] B. Ramdani, D. Chevers, and D. A. Williams, "SMEs' adoption of enterprise applications: A technology-organisation-environment model," *J. Small Bus. Enterp. Dev.*, 2013.

[3] T. H. Davenport, *Mission critical: realizing the promise of enterprise systems*. Harvard Business Press, 2000.

[4] M. L. Markus and C. Tanis, "The enterprise systems experience-from adoption to success," *Fram. domains IT Res. Glimpsing Futur. through past*, vol. 173, no. 2000, pp. 173–207, 2000.

[5] M. Supramaniam, A. Abdullah, and R. Ponnan, "Cost Analysis on ERP System Implementation amongst Malaysian SMEs," *Int. J. Trade, Econ. Financ.*, vol. 5, no. 1, pp. 72–76, 2014, doi: 10.7763/ijtef.2014.v5.343.

[6] M. Daneva and R. Wieringa, "Cost estimation for cross-organizational ERP projects: Research perspectives," *Softw. Qual. J.*, vol. 16, no. 3, pp. 459–481, 2008, doi: 10.1007/s11219-008-9045-8.

[7] W. B. Frakes and K. Kang, "Software reuse research: Status and future," *IEEE Trans. Softw. Eng.*, vol. 31, no. 7, pp. 529–536, 2005.

[8] I. Sommerville, M. Fowler, K. Beck, J. Brant, W. Opdyke, and D. Roberts, "Edition: Software Engineering," *Instructor*, 2019.

[9] M. Mijač, R. Picek, and D. Andročec, "Determinants of ERP Systems as a Large-Scale Reuse Approach," in *MATEC Web of Conferences*, 2019, vol. 292, p. 3007.

[10] L. Brehm, A. Heinzl, and M. L. Markus, "Tailoring ERP systems: a spectrum of choices and their implications," in *Proceedings of the 34th annual Hawaii international conference on system sciences*, 2001, pp. 9-pp.

[11] B. Light, "The maintenance implications of the customization of ERP software," *J. Softw. Maint. Evol. Res. Pract.*, vol. 13, no. 6, pp. 415–429, 2001.

[12] M. Keil and A. Tiwana, "Relative importance of evaluation criteria for enterprise systems: a conjoint study," *Inf. Syst. J.*, vol. 16, no. 3, pp. 237–262, 2006.

[13] F. Asher, "ASAP implementation roadmap," *Clear path SAP< www. Clear.*

com/ASAP_roadmap. pdf> accessed, vol. 15, p. 2015, 2009.

[14] E. Stensrud, “Alternative approaches to effort prediction of ERP projects,” *Inf. Softw. Technol.*, vol. 43, no. 7, pp. 413–423, 2001, doi: 10.1016/S0950-5849(01)00147-1.

[15] N. K. Omural and O. Demirors, “Effort estimation for ERP projects - A systematic review,” in *Proceedings - 43rd Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2017*, 2017, pp. 96–103, doi: 10.1109/SEAA.2017.68.

[16] B. Ozkan, O. Turetken, and O. Demirors, “Software functional size: For cost estimation and more,” in *European Conference on Software Process Improvement*, 2008, pp. 59–69.

[17] C. Gencel and O. Demirors, “Functional size measurement revisited,” *ACM Trans. Softw. Eng. Methodol.*, vol. 17, no. 3, pp. 1–36, 2008.

[18] A. Singh and J. Wesson, “Evaluation criteria for assessing the usability of ERP systems,” in *Proceedings of the 2009 annual research conference of the South African Institute of Computer Scientists and Information Technologists*, 2009, pp. 87–95.

[19] “The 2022 ERP Report,” *Panaroma Consulting Group*, 2022. https://www.panorama-consulting.com/resource-center/erp-report/.

[20] M. Daneva, “Understanding functional reuse of ERP requirements in the telecommunication sector: An empirical study,” in *Proceedings - 2014 Joint Conference of the International Workshop on Software Measurement, IWSM 2014 and the International Conference on Software Process and Product Measurement, Mensura 2014*, 2014, pp. 216–221, doi: 10.1109/IWSM.Mensura.2014.42.

[21] Technavio, “Enterprise Application Software Market by Deployment and Geography - Forecast and Analysis 2022-2026,” 2021. https://www.technavio.com/report/enterprise-application-software-market-industry-analysis.

[22] “Business Analytics And Enterprise Software Global Market Report 2022,” *The Business Research Company*, 2022. https://www.thebusinessresearchcompany.com/report/business-analytics-enterprise-software-global-market-report.

[23] W. Rosa, “Empirical effort and schedule estimation for enterprise resource planning projects,” in *Proceedings of the 2012 Joint Conf. of the 22nd Int. Workshop on Software Measurement and the 2012 7th Int. Conf. on Software Process and Product Measurement, IWSM-MENSURA 2012*, 2012, pp. 190–197, doi: 10.1109/IWSM-

MENSURA.2012.35.

[24] W. Rosa, T. Packard, A. Krupanand, J. W. Bilbro, and M. M. Hodal, "COTS integration and estimation for ERP," *J. Syst. Softw.*, vol. 86, no. 2, pp. 538–550, 2013, doi: 10.1016/j.jss.2012.09.030.

[25] M. Daneva, "Balancing uncertainty of context in ERP project estimation: an approach and a case study," *J. Softw. Maint. Evol. Res. Pract.*, vol. 22, no. 5, pp. 329–357, 2010.

[26] B. Boehm, "The COCOMO 2.0 software cost estimation model," *Am. Program.*, 1996.

[27] C. Z. Mooney, *Monte carlo simulation*, no. 116. Sage, 1997.

[28] I. P. Erasmus and M. Daneva, "ERP effort estimation based on expert judgments," in *Proceedings - Joint Conference of the 23rd International Workshop on Software Measurement and the 8th International Conference on Software Process and Product Measurement, IWSM-MENSURA 2013*, 2013, pp. 104–109, doi: 10.1109/IWSM-Mensura.2013.25.

[29] P. Erasmus and M. Daneva, "ERP services effort estimation strategies based on early requirements," *CEUR Workshop Proc.*, vol. 1342, no. 1, pp. 83–99, 2015.

[30] P. Erasmus and M. Daneva, "An experience report on ERP effort estimation driven by quality requirements," *CEUR Workshop Proc.*, vol. 1342, pp. 126–139, 2015.

[31] F. Vogelezang, "Using COSMIC-FFP for sizing, estimating and planning in an ERP environment," in *16th International Workshop on Software Measurement and DASMA Metrik Kongress*, 2006, pp. 327–342.

[32] F. M. Téllez, "Solving the size estimation problem in ERP project context : the eEPC- COSMIC approach," University of Twente, 2009.

[33] I. P. Erasmus, "The COSMIC EPC method An ERP functional size measurement method delivering time and cost estimates," University of Gothenburg, 2012.

[34] "ISO - ISO/IEC 24570:2018 - Software engineering — NESMA functional size measurement method — Definitions and counting guidelines for the application of function point analysis." https://www.iso.org/standard/72505.html (accessed Oct. 22, 2021).

[35] A. G. Asuero, A. Sayago, and A. G. González, "The correlation coefficient: An overview," *Crit. Rev. Anal. Chem.*, vol. 36, no. 1, pp. 41–59, 2006.

[36] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information

systems research," *MIS Q.*, pp. 75–105, 2004.

[37] P. Offermann, S. Blom, M. Schönherr, and U. Bub, "Artifact types in information systems design science–a literature review," in *International Conference on Design Science Research in Information Systems*, 2010, pp. 77–92.

[38] P. Offermann, O. Levina, M. Schönherr, and U. Bub, "Outline of a design science research process," in *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*, 2009, pp. 1–11.

[39] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele Univ.*, vol. 33, no. 2004, pp. 1–26, 2004.

[40] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, 2014, pp. 1–10.

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

[42] Robert K. Yin, *Case study research and applications: Design and methods*. Sage publications., 2017.

[43] C. Robson, *Real world research: A resource for social scientists and practitioner-researchers*. Wiley-Blackwell, 2002.

[44] C. De Santo, M. Totoro, and R. Moscatelli, *Introduction to statistics*. Learning Solutions., 2010.

[45] W. S. Humphrey, *A discipline for software engineering*. Pearson Education India, 1995.

[46] M. A. Ahmed, I. Ahmad, and J. S. AlGhamdi, "Probabilistic size proxy for software effort prediction: A framework," *Inf. Softw. Technol.*, vol. 55, no. 2, pp. 241–251, 2013.

[47] J. Miles, "Residual Plot," *Wiley StatsRef Stat. Ref. Online*, 2014.

[48] E. Kocaguneli and T. Menzies, "Software effort models should be assessed via leave-one-out validation," *J. Syst. Softw.*, vol. 86, no. 7, pp. 1879–1890, 2013.

[49] A. Idri, I. Abnane, and A. Abran, "Evaluating Pred (p) and standardized accuracy criteria in software development effort estimation," *J. Softw. Evol. Process*, vol. 30, no. 4, p. e1925, 2018.

[50] S. D. Conte, H. E. Dunsmore, and Y. E. Shen, *Software engineering metrics and models*. Benjamin-Cummings Publishing Co., Inc., 1986.

[51] S. G. MacDonell and A. R. Gray, “A comparison of modeling techniques for software development effort prediction,” 1998.

[52] G. Tate, J. Verner, and R. Veryard, “Software costing in practice,” *Econ. Inf. Syst. Softw.*, pp. 101–126, 1991.

[53] S. G. MacDonell, “Establishing relationships between specification size and software process effort in CASE environments,” *Inf. Softw. Technol.*, vol. 39, no. 1, pp. 35–45, 1997.

[54] A. J. Albrecht, “Measuring application development productivity,” 1979.

[55] “ISO - ISO/IEC 20926:2009 - Software and systems engineering — Software measurement — IFPUG functional size measurement method 2009.” https://www.iso.org/standard/51717.html (accessed Oct. 22, 2021).

[56] “ISO - ISO/IEC 19761:2011 - Software engineering — COSMIC: a functional size measurement method.” https://www.iso.org/standard/54849.html (accessed Oct. 22, 2021).

[57] “ISO/IEC 24570:2018 Software engineering — NESMA functional size measurement method,” *2018*. https://www.iso.org/standard/72505.html.

[58] “ISO - ISO/IEC 20968:2002 - Software engineering — Mk II Function Point Analysis — Counting Practices Manual.” https://www.iso.org/standard/35603.html (accessed Oct. 22, 2021).

[59] C. Ebert and H. Soubra, “Functional size estimation technologies for software maintenance,” *IEEE Softw.*, vol. 31, no. 6, pp. 24–29, 2014.

[60] G. Srivastava, Y. More, and J. Sam, “Effort Estimation Model for an Enterprise Software Upgrade,” in *2020 International Conference for Emerging Technology (INCET)*, 2020, pp. 1–6.

[61] S. Jalali and C. Wohlin, “Systematic literature studies: database searches vs. backward snowballing,” in *Proceedings of the 2012 ACM-IEEE international symposium on empirical software engineering and measurement*, 2012, pp. 29–38.

[62] S. Koch and J. Mitlöhner, “Effort estimation for enterprise resource planning implementation projects using social choice - a comparative study,” *Enterp. Inf. Syst.*, vol. 4, no. 3, pp. 265–281, 2010, doi: 10.1080/17517575.2010.496494.

[63] M. Daneva, “Preliminary results in a multi-site empirical study on cross-

organizational ERP size and effort estimation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4895 LNCS, pp. 60–71, 2008, doi: 10.1007/978-3-540-85553-8_5.

[64] F. Vogelezang *et al.*, "Estimating packaged software implementations: The first part of a framework," in *Proceedings of the 2012 Joint Conf. of the 22nd Int. Workshop on Software Measurement and the 2012 7th Int. Conf. on Software Process and Product Measurement, IWSM-MENSURA 2012*, 2012, pp. 1–7, doi: 10.1109/IWSM-MENSURA.2012.10.

[65] T. Hansen, "Multidimensional effort prediction for ERP system implementation," in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 2006, pp. 1402–1408.

[66] M. Daneva, "Uncertain context factors in ERP project estimation are an asset: Insights from a semi-replication case study in a financial services firm," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 21, no. 3, pp. 389–411, 2011, doi: 10.1142/S0218194011005335.

[67] M. Daneva, S. Wettflower, and S. De Boer, "Uncertainty in ERP effort estimation: A challenge or an asset?," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5338 LNCS, no. i, 2008, pp. 208–222.

[68] M. Daneva, "Managing uncertainty in erp project estimation practice: An industrial case study," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5089 LNCS, pp. 90–104, 2008, doi: 10.1007/978-3-540-69566-0_10.

[69] M. Daneva, "Integrating portfolio management and simulation concepts in the ERP project estimation practice," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5025 LNCS, no. iv, pp. 147–152, 2008, doi: 10.1007/978-3-540-69062-7_15.

[70] M. Daneva, "Integrating reuse measurement practices into the ERP requirements engineering process," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4034 LNCS, pp. 112–126, 2006, doi: 10.1007/11767718_12.

[71] C. Kuijpers, "Automated FPA (eFPA) in SAP environment - Visions and experiences of automated function point analysis," *Proc. - 2014 Jt. Conf. Int. Work. Softw. Meas. IWSM 2014 Int. Conf. Softw. Process Prod. Meas. Mensura 2014*, pp. 72–78, 2014, doi: 10.1109/IWSM.Mensura.2014.35.

[72] N. K. Ömüral and O. Demirörs, "Effort estimation methods for ERP projects based on function points: A case study," in *ACM International Conference Proceeding*

Series, 2017, vol. Part F1319, pp. 199–206, doi: 10.1145/3143434.3143464.

[73] O. Demirörs and N. Küçükateş Ömüral, "Exploring reuse levels in ERP projects in search of an effort estimation approach," in *Proceedings - 44th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2018*, 2018, pp. 191–197, doi: 10.1109/SEAA.2018.00038.

[74] Ö. Özcan Top, "Functional similarity impact on the relation between functional size and software development effort." Middle East Technical University, 2008.

[75] O. O. Top, O. Demirors, and O. Turetken, "Making functional similarity count for more reliable effort prediction models," in *2009 24th International Symposium on Computer and Information Sciences*, 2009, pp. 504–512.

[76] E. Monk and B. Wagner, *Concepts in enterprise resource planning*. Cengage Learning, 2012.

[77] T. Orosz, "Analysis of SAP Development tools and methods," in *2011 15th IEEE International Conference on Intelligent Engineering Systems*, 2011, pp. 439–443.

[78] T. E. Hastings and A. S. M. Sajeev, "A vector-based approach to software size measurement and effort estimation," *IEEE Trans. Softw. Eng.*, vol. 27, no. 4, pp. 337–350, 2001.

[79] S. J. Raudys and A. K. Jain, "Small sample size effects in statistical pattern recognition: Recommendations for practitioners," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 3, pp. 252–264, 1991.

[80] R. Akkiraju and H. Van Geel, "Estimating the cost of developing customizations to packaged application software using service oriented architecture," in *ICWS 2010 - 2010 IEEE 8th International Conference on Web Services*, 2010, pp. 433–440, doi: 10.1109/ICWS.2010.94.

[81] M. T. Kotb, M. Haddara, and Y. T. Kotb, "Back-propagation artificial neural network for ERP adoption cost estimation," in *International Conference on Enterprise Information Systems*, 2011, pp. 180–187.

[82] T. Chandrakumar and S. Parthasarathy, "An approach to estimate the size of ERP package using package points," *Comput. Stand. Interfaces*, vol. 47, pp. 100–107, 2016, doi: 10.1016/j.csi.2015.10.003.

[83] G. Janssens, R. Kusters, and F. Heemstra, "Clustering ERP implementation project activities: a foundation for project size definition," *Technol. Collab. Bus. Process. Manag. Enterp. Inf. Syst.*, vol. 23, 2007.

[84] M. Haddara, "ERP adoption cost factors in SMEs," in *Proceedings of the European,*

Mediterranean and Middle Eastern Conference on Information Systems - Informing Responsible Management: Sustainability in Emerging Economies, EMCIS 2011, 2011, vol. 2011, pp. 130–141.

[85] C. Equey, R. J. Kusters, S. Varone, and N. Montandon, "Empirical study of ERP systems implementation costs in swiss smes," *ICEIS 2008 - Proc. 10th Int. Conf. Enterp. Inf. Syst.*, vol. DISI, pp. 143–148, 2008, doi: 10.5220/0001683601430148.

[86] M. Haddara and A. Elragal, "ERP adoption cost factors identification and classification: A study in SMEs," *Int. J. Inf. Syst. Proj. Manag.*, vol. 1, no. 2, pp. 5–21, 2013, doi: 10.12821/ijispm010201.

[87] R. J. Kusters, F. J. Heemstra, and A. Jonker, "Determining the costs of ERP implementation," *ICEIS 2007 - 9th Int. Conf. Enterp. Inf. Syst. Proc.*, vol. DISI, pp. 102–110, 2007, doi: 10.5220/0002363001020110.

[88] J. Lee, S. K. Chen, V. Shivpuriya, and R. Mohan, "Service cost estimation for packaged business application-based business transformation," *Proc. 2008 IEEE Int. Conf. Serv. Oper. Logist. Informatics, IEEE/SOLI 2008*, vol. 1, pp. 890–895, 2008, doi: 10.1109/SOLI.2008.4686525.

[89] H. Barki and A. Pinsonneault, "Explaining ERP Implementation Effort & Benefits With Organizational Integration," *Cah. du GReSI*, vol. 2, no. 02, p. 1, 2002, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.4.3572&rep=rep1&type=pdf.

[90] S. Koch, "ERP implementation effort estimation using data envelopment analysis," *Technol. Bus. Inf. Syst.*, pp. 121–132, 2007, doi: 10.1007/1-4020-5634-6_11.

[91] A. Momoh, "A framework for complexity cost modelling of ERP implementation," Cranfield University, 2015.

[92] J.-S. Chou, M.-Y. Cheng, Y.-W. Wu, and C.-C. Wu, "Forecasting enterprise resource planning software effort using evolutionary support vector machine inference model," *Int. J. Proj. Manag.*, vol. 30, no. 8, pp. 967–977, 2012, doi: 10.1016/j.ijproman.2012.02.003.

[93] P. Erasmus and M. Daneva, "An empirical study on memory bias situations and correction strategies in ERP effort estimation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9459, pp. 227–242, 2015, doi: 10.1007/978-3-319-26844-6_17.

[94] B. Zrimsek, P. Phelan, F. Karamouzis, and N. Frey, "Estimating the Time and Cost of ERP and ERP II Projects: A 10-Step Process," *Gartner*, no. April, p. 18, 2001, [Online]. Available: http://www.gartner.com/id=341315.

[95] M. Plaza and K. Rohlf, “Learning and performance in ERP implementation projects: A learning-curve model for analyzing and managing consulting costs,” *Int. J. Prod. Econ.*, vol. 115, no. 1, pp. 72–85, 2008, doi: 10.1016/j.ijpe.2008.05.005.

[96] T. van Hest, “Factors that determine and control the Total Cost of Ownership of an ERP solution-A maturity model that increases reliable ERP cost estimation,” p. 59, 2013.

[97] C. Francalanci, “Predicting the implementation effort of ERP projects: Empirical evidence on SAP/R3,” *J. Inf. Technol.*, vol. 16, no. 1, pp. 33–48, 2001, doi: 10.1080/02683960010035943.

APPENDICES

APPENDIX A – SYSTEMATIC LITERATURE REVIEW

Data Source

Table 40. SLR Data Source

Electronic Database
Scholar.Google
Proquest
ScienceDirect
INFORMS PubsOnline
IEEE Xplore Digital Library

Table 41. SLR Search Strings

Search Strings
ERP size measurement
ERP effort estimation
ERP work estimation
ERP time estimation
ERP cost estimation
ERP effort prediction
ERP work prediction
ERP time prediction
ERP cost prediction
RICE Objects
Business Blueprint
SAP size measurement
SAP effort estimation
SAP work estimation
SAP time estimation
SAP cost estimation
SAP effort prediction
SAP work prediction
SAP time prediction
SAP cost prediction

Research Approaches

Theory: Theoretical evaluation of effort estimation methods' features. Studies applying theories are counted in this classification if only they heavily rely on non-empirical research methods for effort estimation method suggestion or evaluation.

Survey: Survey-based studies as interviews, questionnaires to understand industry practice.

Experiment: Experiment-based studies for testing different hypothesis by trial.

Case study: Case-based studies to analyze deeply estimation processes of one, or a very small number, ERP projects.

Development of estimation method: Studies where size/effort estimation methods are developed. These methods could be a combination of existing methods. Studies defining how to apply an existing estimation method for ERP projects are also counted in this classification.

History-based evaluation: Studies using historical project data to evaluate estimation methods or figure out relationships of other factors.

Own experience/lessons learned: Studies heavily rely on one's own experience where this experience is not supported with a scientific documentation as a case study, an observation or an experiment.

Real-life evaluation: Studies evaluating estimation methods in real industry estimation situations.

Review: Studies with main purpose of reviewing other estimation papers/methods for ERP projects.

Simulation: Simulation-based studies. Studies using simulation for estimation method evaluation are counted in this classification.

Other: Studies with other research approaches.

Distribution of Papers Based on Estimation Topics

Based on the size measurement / effort estimation methods they applied, all primary studies are listed in Table 42.

Table 42. Papers Based on Estimation Topics

Estimation Topic	**Papers**
Analogy	[62]
Artifact-Centric Approach	[80]
Artificial Neural Network	[81]
Automated FPA	[5]
COCOMO II	[25][82][66][67][68][69]
Complexity Cost Modeling	[83]
COSMIC EPC	[33]
Cost Drivers	[64][84][85][86][87][88][89]
Cross-Organizational Estimation	[6][63]
Data Envelopment Analysis	[90]
eEPC-COSMIC	[91]
Evolutionary Support Vector Machine Inference Model	[92]
Expert Judgments	[28][29][30][93][94]
Function Points	[64][29][30]
Function Points (reuse measurement)	[20][70]
Learning-Curve Model	[95]
Linguistic Analysis	[80]
Maturity Model (Cost Factors)	[96]
Monte Carlo Simulation	[25][66][67][68][69]
Non-parametric Models	[14]
Organizational Integration	[32]
Package Points	[82]
Parametric Models	[14]
Portfolio Management	[25][66][67][68]

Estimation Topic	Papers
Refined Approximate COSMIC-FFP	[31]
Regression Analysis	[62][97]
RICE Objects	[23]
Similarity-based estimation (key-ratios)	[65]
Size Drivers	[64][83]
Social Choice	[62]
WBS Model	[80]

Characteristics of Estimation Papers

Distribution of the inputs used in the estimation studies is shown in Table 43.

Table 43. Distribution of the Inputs Used in the Estimation Papers

Input	# Papers
Number of Business Objects	1
COSMIC Function Points	3
IFPUG Function Points	1
LOC (derived from IFPUG FP)	1
Package Points	1
Project Characteristics	3
Project Scope	1
Work Break-Down Structure	1

Prediction models used in the studies are listed in Table 44.

Table 44. Prediction Models Used in the Estimation Papers

Prediction Model	Paper
Support Vector Machine (learning&curve fitting) & Fast Messy Genetic Algorithm (optimizing prediction errors)	[92]
COCOMO II (accounting ERP adopter's specific cost drivers) Monte Carlo Simulation (approaching degrees of uncertainty of the cost drivers)	[25]

Prediction Model	**Paper**
Portfolio Management (quantifing the chance for success with proposed interdependent deadlines)	
Artifact-centric & Linguistic analysis approaches (deriving the number of business objects) & WBS model	[80]
Expert judgment based prediction	[28]
Social Choice (ranking projects per attribute by voters)	[62]
Data Envelopment Analysis	[90]
Effort formula using Cfu, reuse&modification levels, conversion factor and template time	[33]
Effort formula using Cfu,#production lines and conversion factor	[31]
COCOMO II	[82]
Effort formula using project scope	[88]

Papers validated estimation method using "history-based validation approach"; "data sets" and "descriptive statistics" are listed in Table 45 as follows.

Table 45. Descriptive Statistics of the Estimation Papers

Paper	**Datasets**	**Descriptive Statistics**
[92]	182 ERP Projects	MMRE(training)-26,8% MMRE(testing)- 27,3%
[62]	39 ERP Projects	MMRE(Copeland)=5.43% MMRE(Borda)= 9,58%
[90]	35 ERP Projects	MMRE= 54,81%
[33]	9 SAP Projects	MMRE =8%
[82]	14 ERP Projects	R (coefficient of correlation)=0,94

APPENDIX B – SAMPLE SIZE MEASUREMENT

Business Process Repository

Table 46. BPR Master Data List for SAP MM Module

Organizational Area	**Object Type**	**Name**	**Package**	**Module**
Procurement	Master Data	Buyer	BP_LIB_R3 MM	MM
Procurement	Master Data	Conditions (Procurement)	BP_LIB_R3 MM	MM
Procurement	Master Data	Contract (Purchasing)	BP_LIB_R3 MM	MM
Procurement	Master Data	Delivery Address	BP_LIB_R3 MM	MM
Procurement	Master Data	Manufacturer Part Number	BP_LIB_R3 MM	MM
Procurement	Master Data	Purchasing Info Record	BP_LIB_R3 MM	MM
Procurement	Master Data	Quota Arrangement	BP_LIB_R3 MM	MM
Procurement	Master Data	Release Strategy with Classification	BP_LIB_R3 MM	MM
Procurement	Master Data	Service Condition	BP_LIB_R3 MM	MM
Procurement	Master Data	Settlement Accounting for Conditions Requiring Subsequent Settlement	BP_LIB_R3 MM	MM
Procurement	Master Data	Source List	BP_LIB_R3 MM	MM
Procurement	Master Data	Standard Service Catalog	BP_LIB_R3 MM	MM
Procurement	Master Data	Sustainability Information Record	BP_LIB_R3 MM	MM
Procurement	Master Data	Vendor Evaluation	BP_LIB_R3 MM	MM
Procurement	Master Data	Vendor Master Record	BP_LIB_R3	MM
Procurement	Master Data	Vendor Rebate Arrangements	BP_LIB_R3 MM	MM
Procurement	Master Data	Vendor Sustainability Record	BP_LIB_R3 MM	MM

SAP Sample Views

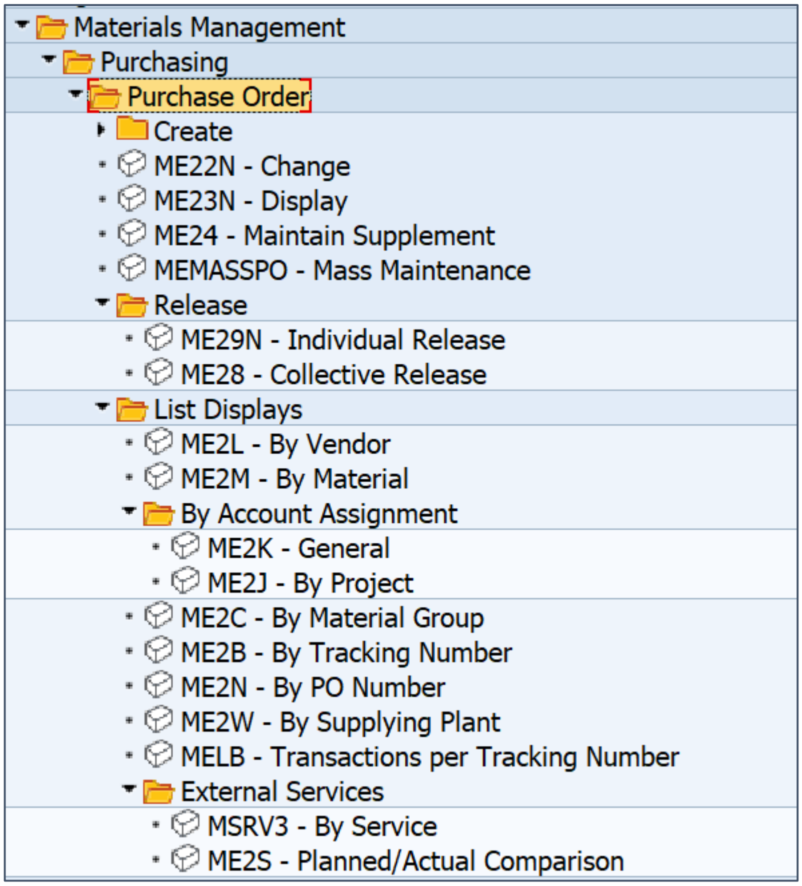


Figure 28. SAP Transactions for Purchase Order Process

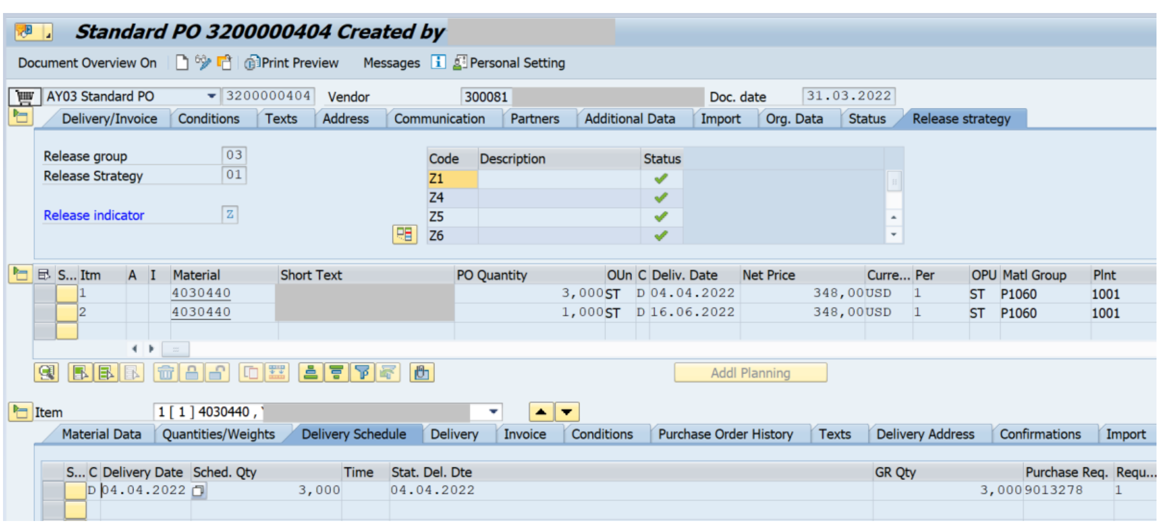


Figure 29. SAP Sample View for Purchase Order Display

APPENDIX C – MULTIPLE REGRESSION ANALYSIS

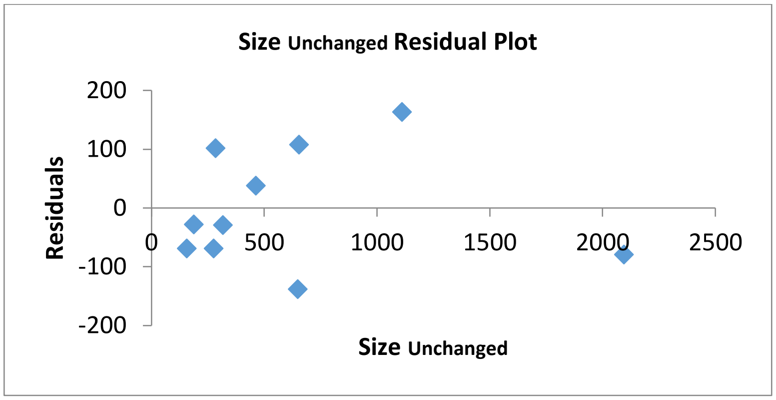


Figure 30. Case Study 4 - Residual Plot of Size $_{unchanged}$ for the 1st Project

Figure 31. Case Study 4 - Residual Plot of Size $_{changed}$ for the 1st Project

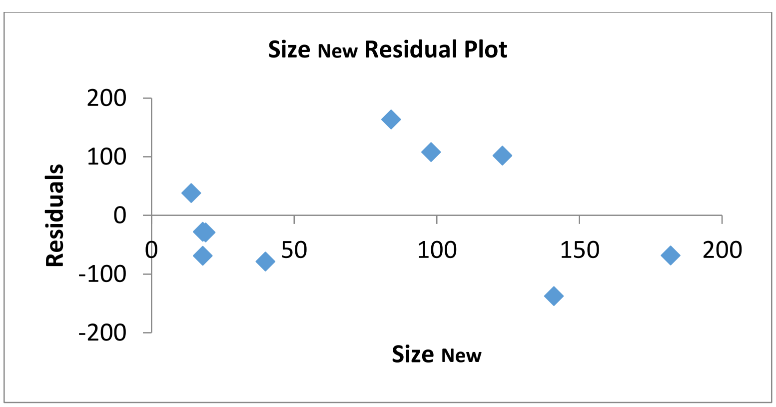


Figure 32. Case Study 4 - Residual Plot of Size $_{new}$ for the 1st Project

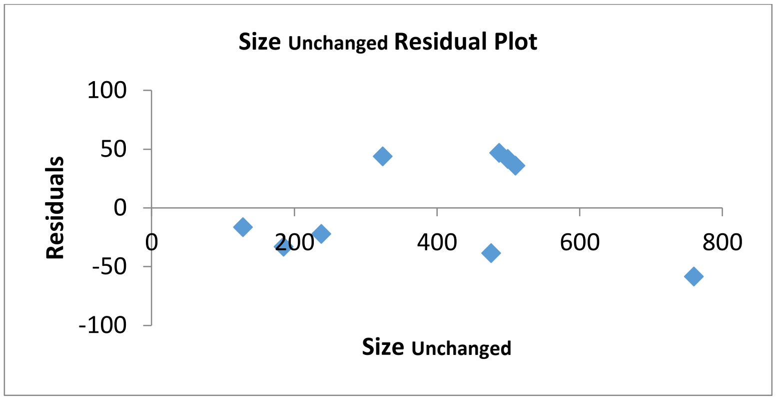


Figure 33. Case Study 4 - Residual Plot of Size $_{unchanged}$ for the 2nd Project

Figure 34. Case Study 4 - Residual Plot of Size $_{changed}$ for the 2nd Project

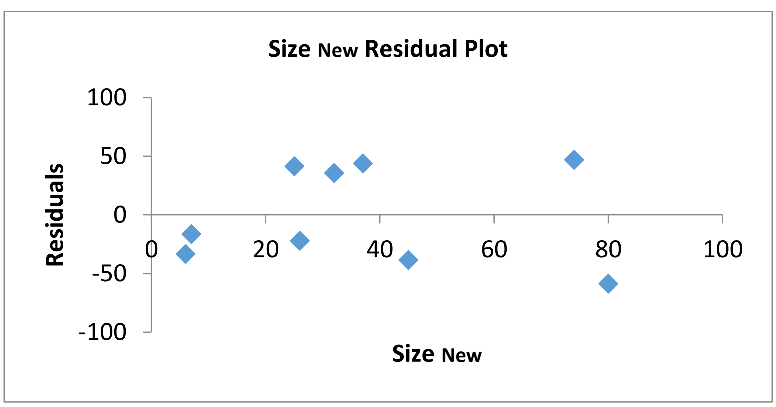


Figure 35. Case Study 4 - Residual Plot of Size $_{new}$ for the 2nd Project

CURRICULUM VITAE

PERSONAL INFORMATION

Neslihan Küçükateş Ömüral

kucukates@gmail.com

https://www.linkedin.com/in/neslihan-küçükateş-ömüral-a6305b25/

https://www.researchgate.net/profile/Neslihan-Kuecuekates-Oemueral

EDUCATION

Degree	Institution	Year of Graduation
PhD	METU, Information Systems	2022
MS	METU, Information Systems	2010
BS	METU, Industrial Engineering	2006
High School	Halide Edip Adıvar High School	2001

WORK EXPERIENCE

Year	Place	Enrollment
2013-Present	Lentatek (Zorlu Holding), Ankara, Turkey	Information Systems Manager
2012-Present	AYESAŞ (Zorlu Holding), Ankara, Turkey	Information Systems Manager
2022-2022	Atılım University, Ankara, Turkey	Part Time Faculty Member
2009-2011	Borçelik, Bursa, Turkey	SAP System Analyst and Programmer
2008-2009	Bosch Rexroth, Bursa, Turkey	SAP Specialist
2006-2008	Yazaki Automative, Bursa, Turkey	Production Planning Engineer & SAP Key User

PROJECTS

Year	Place	Project
2021	AYESAŞ & Lentatek	SAP Production Tracking Project
2021	AYESAŞ & Lentatek	SAP E-Payment Project
2020	AYESAŞ	ESSEGI Warehouse Automation Project
2020	AYESAŞ & Lentatek	SAP E-Delivery Project
2019	Lentatek	SAP SD & PP & MM Modules Implementation Project
2018	AYESAŞ & Lentatek	SAP E-Ledger Project
2017	AYESAŞ & Lentatek	SAP E-Invoice Project
2017	AYESAŞ & Lentatek	SAP MT940 Electronic Bank Statement Project
2017	AYESAŞ	ISO 27001 Certification Project
2013	AYESAŞ & Lentatek	SAP TRM Implementation Project
2010	Borçelik	SAP Supplier Portal Project
2010	Borçelik	SAP Authorization Management Project
2009	Borçelik	SAP Warehouse Management Project
2008	Bosch Rexroth	SAP E-Kanban Project
2008	Bosch Rexroth	SAP Spare Parts Management Project
2008	Bosch Rexroth	SAP Warehouse Management Project

PUBLICATIONS

1	Ömüral, N. K., & Demirörs, O. (2022). Measuring size of Enterprise Applications: Using data transaction points in the field. *Information and Software Technology.* (submitted)
2	Ömüral, N. K., & Demirörs, O. (2022, September). A Size Measurement Method for Enterprise Applications. In Proceedings of the 31st International Workshop on Software Measurement and 16th International Conference on Software Process and Product Measurement (submitted)
3	Ömüral, N. K., & Demirörs, O. (2019, September). ERP Projelerinde Efor Kestirimi. Ulusal Yazılım Mühendisliği Sempozyumu (UYMS)
4	Demirors, O., & Omural, N. K. (2018, August). Exploring reuse levels in ERP projects in search of an effort estimation approach. In *2018 44th Euromicro*

	Conference on Software Engineering and Advanced Applications (SEAA) (pp. 191-197). IEEE Computer Society.
5	Ömüral, N. K., & Demirörs, O. (2017, October). Effort estimation methods for ERP projects based on function points: a case study. In Proceedings of the 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement (pp. 199-206).
6	Ömüral, N. K., & Demirörs, O. (2017, August). Effort Estimation for ERP Projects—A Systematic Review. In *2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)* (pp. 96-103). IEEE.

FOREIGN LANGUAGES

Native Turkish, Advanced English, Intermediate German …